

# **CHƯƠNG 5**

# **MÔ HÌNH HÓA HỆ THỐNG**

# NỘI DUNG



Luồng công việc



Phân tích hệ thống



Thiết kế usecase



# 1. Luồng công việc



Phân tích hệ  
thống

Đánh giá hiện  
trạng tổ chức

Xác định  
thuật ngữ

Xác định ràng  
buộc

Xác định tác nhân và  
use case hệ thống

Lập mô hình use  
case hệ thống

Thiết kế use  
case

Đặc tả use  
case

Xác định thực  
thể

Hiện thực  
hoá use case

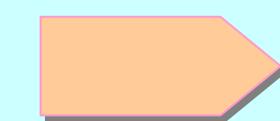
Lập mô hình lớp đối  
tượng



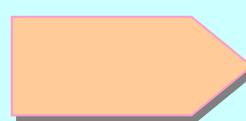
## 2. Phân tích hệ thống



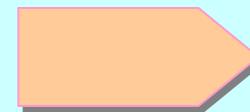
Phân tích hệ  
thống



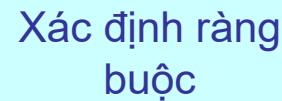
Đánh giá hiện  
trạng tổ chức



Xác định  
thuật ngữ



Lập mô hình use  
case hệ thống



Xác định ràng  
buộc



Xác định tác nhân và  
use case hệ thống



## 2.1 Đánh giá hiện trạng tổ chức



- Sơ đồ tổ chức
- Xác định đối tượng liên quan và khách hàng
- Mô tả nhu cầu của từng đối tượng liên quan
- Giới hạn hệ thống phát triển
- Xác định và mô tả vấn đề có hệ thống

## 2.1 Đánh giá hiện trạng tổ chức



### ■ Mục đích:

- ◆ Đánh giá và nắm bắt thông tin về tổ chức nghiệp vụ
- ◆ Xác định các đối tượng tham gia (stakeholder) và khách hàng của hệ thống.
- ◆ Định nghĩa phạm vi của việc mô hình hóa nghiệp vụ.
- ◆ Tán thành những tiềm năng cải tiến và các mục tiêu mới của tổ chức.
- ◆ Mô tả những mục tiêu chính của tổ chức.



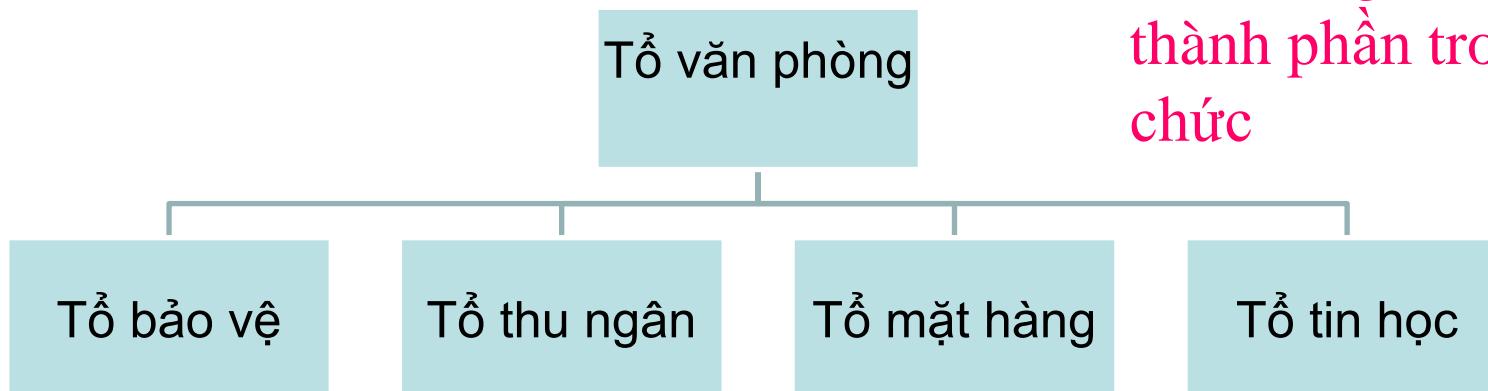
## 2.1 Đánh giá hiện trạng tổ chức



### ■ Nắm bắt thông tin về tổ chức:

- ◆ Cơ cấu tổ chức, phân cấp và các vai trò trong hệ thống.
- ◆ Mô tả ngắn gọn các thành phần và mối quan hệ này thông qua sơ đồ tổ chức

Cần mô tả ngắn gọn bằng văn bản vai trò và hoạt động của từng thành phần trong tổ chức



Sơ đồ tổ chức của siêu thị



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Nắm bắt thông tin về tổ chức

- ◆ ví dụ: mô tả thông tin hoạt động

**Tổ văn phòng:** Gồm 1 Giám đốc và 2 phó Giám đốc có nhiệm vụ điều phối toàn bộ hoạt động của siêu thị. Tổ phải nắm được tình hình mua bán, doanh thu của siêu thị để báo cáo lại cho ban giám đốc. Việc báo cáo được thực hiện hàng tháng, hàng quý hoặc cũng có khi báo cáo đột xuất theo yêu cầu



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Xác định các đối tượng liên quan và khách hàng

- ◆ **Đối tượng liên quan (stakeholder)**: là những cá nhân chịu ảnh hưởng trực tiếp từ các tác động của hệ thống.
- ◆ **Khách hàng**: người dùng hệ thống, có thể là các stakeholder.



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Xác định các đối tượng liên quan và khách hàng

Tên	Đại diện	Vai trò
Người quản lý	Giám đốc, người quản lý siêu thị	Theo dõi tiến trình phát triển của dự án và theo dõi tình hình hoạt động của siêu thị.
Nhân viên bán hàng	Người nhập các thông tin trong hệ thống.	Chịu trách nhiệm trong khâu bán hàng ở siêu thị, duy trì hoạt động của siêu thị.



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Nắm bắt nhu cầu các đối tượng liên quan

Tên	Mô tả
Người quản lý	Đáp ứng các nhu cầu quản lý siêu thị như hàng hóa, khách hàng, doanh số.
Nhân viên bán hàng	Đảm bảo rằng hệ thống sẽ đáp ứng các nhu cầu của công việc bán hàng.
Khách hàng	Đáp ứng nhu cầu tra cứu thông tin về hàng hóa có trong siêu thị.



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Mô tả nhu cầu của từng đối tượng liên quan

Tên	Độ ưu tiên	Nhu cầu	Giải pháp hiện hành	Giải pháp đề xuất
Người quản lý	Cao	Xem các báo cáo thống kê theo các yêu cầu khác nhau	Báo cáo thống kê doanh thu	Hiển thị báo cáo theo nhiều tiêu chí khác nhau, thông tin bố trí dễ nhìn và đơn giản nhưng đầy đủ.

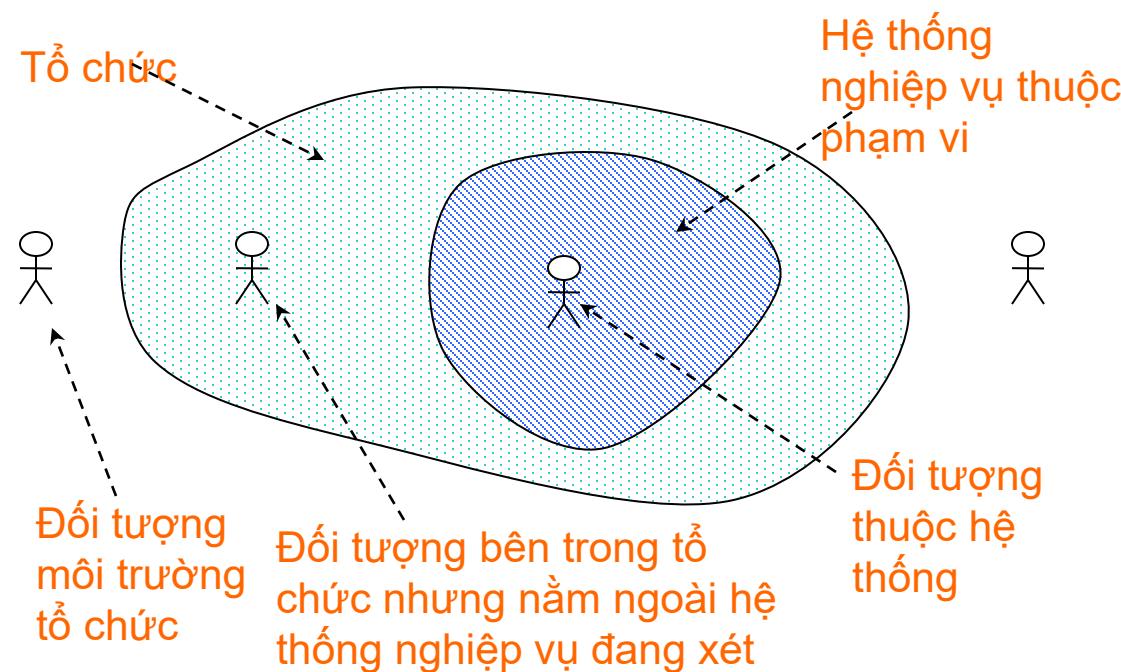


## 2.1 Đánh giá hiện trạng tổ chức



### ■ Giới hạn hệ thống phát triển

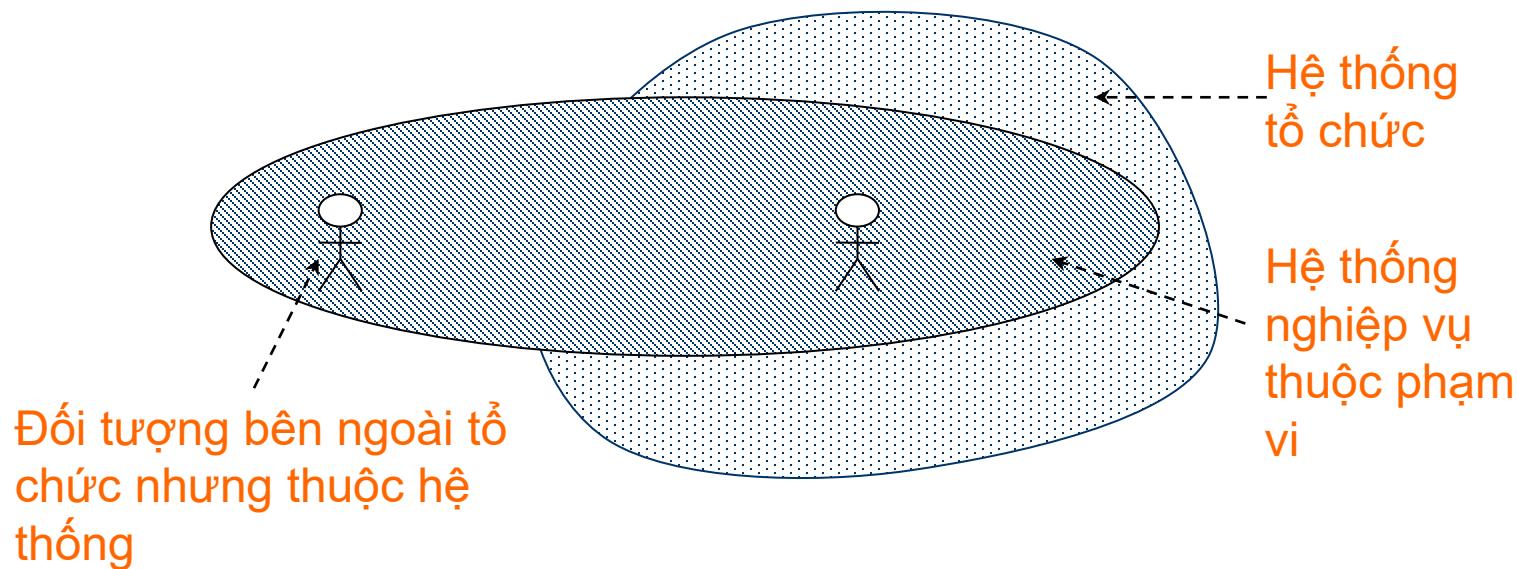
- ◆ Xác định ranh giới phát triển hệ thống bằng cách:
  - Chỉ ra những thực thể nằm ngoài hệ thống
  - Chỉ ra những thực thể bên trong tổ chức nhưng nằm ngoài hệ thống



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Giới hạn hệ thống phát triển(tt)



→ Các hệ thống thương mại điện tử e-Business, e-Commerce



## 2.1 Đánh giá hiện trạng tổ chức



### ■ Xác định và mô tả vấn đề

- ◆ Mẫu trình bày

Vấn đề	mô tả vấn đề
Đối tượng chịu tác động	các đối tượng liên quan bị ảnh hưởng bởi vấn đề
Ảnh hưởng của vấn đề	tác động ảnh hưởng của vấn đề
Một giải pháp thành công	liệt kê một vài lợi ích của một giải pháp thành công



## 2.1 Đánh giá hiện trạng tổ chức



### Trình bày vấn đề có hệ thống

#### ■ Ví dụ:

Vấn đề	Cơ sở dữ liệu của các khách hàng thân thiết được lưu trữ ở nhiều nơi và không có sự đồng bộ.
Đối tượng chịu tác động	Khách hàng, người quản lý
Ảnh hưởng của vấn đề	Dịch vụ khách hàng thân thiết chỉ thiết lập được ở từng siêu thị. Điều này là bất hợp lý, làm rắc rối trong việc nâng cao dịch vụ khách hàng, làm giảm khả năng cạnh tranh của siêu thị.
Một giải pháp thành công	Nhân viên có thể sử dụng chung một tài khoản (account) cấp cho mỗi khách hàng được dùng ở tất cả siêu thị. Nâng cao khả năng chăm sóc khách hàng của siêu thị tốt hơn từ đó thu hút được khách hàng nhiều hơn, tăng doanh thu của siêu thị. Giúp người quản lý có thể làm tốt công tác quản lý khách hàng, theo dõi tình hình phục vụ khách hàng một cách dễ dàng.



## 2.2 Xác định các thuật ngữ nghiệp vụ

Thuật ngữ	Diễn giải
Người quản lý	Người quản lý siêu thị và cũng là người quản trị hệ thống. Nguoiquanly được gọi chung cho những người được cấp quyền là "Quản lý", có thể bao gồm giám đốc, phó giám đốc, kế toán, nhân viên tin học, ...
Nhân viên bán hàng	Là nhân viên làm việc trong siêu thị. Nhân viên bán hàng, đứng ở quầy thu tiền và tính tiền cho khách hàng. Thông qua các mã vạch quản lý trên từng mặt hàng được nhân viên bán hàng nhập vào hệ thống thông qua một đầu đọc mã vạch.
Khách hàng thân thiết	Khách hàng thân thiết của siêu thị hay khách hàng đăng ký tham gia chương trình khách hàng thân thiết của siêu thị.



## 2.3 Xây dựng mô hình Use Case hệ thống

### ■ Mục đích Use Case Diagram:

- ◆ Biểu diễn sơ đồ chức năng của hệ thống
- ◆ Chỉ ra sự tương tác giữa các tác nhân và hệ thống thông qua các use case
- ◆ Một sơ đồ use case là một tập hợp các tác nhân, các use case và các mối quan hệ giữa chúng.
- ◆ Các use case trong sơ đồ use case có thể được phân rã theo nhiều mức khác nhau



## 2.3 Xây dựng mô hình Use Case hệ thống

- Xác định tác nhân hệ thống (Actor)
- Xác định Use case hệ thống (Use case)
- Vẽ sơ đồ Use case hệ thống
- Tinh chế sơ đồ Use case hệ thống



## 2.3 Xây dựng mô hình Use Case hệ thống

### ■ Bộ ký hiệu – Use-case Diagram

Kí hiệu	Diễn giải
	Use-case biểu diễn một chức năng của hệ thống
	Biên của hệ thống, dùng để tách biệt phần bên trong và bên ngoài hệ thống.
	Liên kết giữa actor và use-case, thể hiện tương tác của actor lên use-case
	Mối quan hệ giữa các use-case: - Extend : quan hệ tùy chọn - Include : quan hệ bắt buộc
	Actor là đối tượng bên ngoài hệ thống, tương tác trực tiếp với use-case
	Quan hệ tổng quát hóa của use-case

## 2.3.1 Xác định tác nhân hệ thống



- Tác nhân là người sử dụng hệ thống (cả người quản trị hệ thống) hoặc các phần mềm tin học bên ngoài nhưng có tương tác với hệ thống đang xét.
- Phương pháp xác định tác nhân hệ thống:
  - ◆ Sử dụng câu hỏi gợi ý

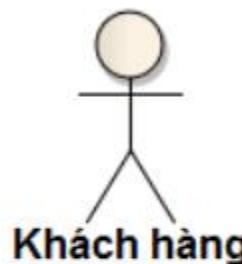
## 2.3.1 Xác định tác nhân hệ thống



### Một số câu hỏi gợi ý xác định tác nhân:

◆ **Tác nhân chính:** là tác nhân sử dụng những chức năng cơ bản của hệ thống.

- Ai đang sử dụng hệ thống?
- Ai được tác động bởi hệ thống?
- Nhóm đối tượng nào cần hệ thống trợ giúp để làm công việc?



*Trong hệ thống ATM*



*Trong hệ thống thư viện*



## 2.3.1 Xác định tác nhân hệ thống



### ■ Một số câu hỏi gợi ý xác định tác nhân:

#### ◆ Tác nhân hỗ trợ:

- Nhóm đối tượng nào hệ thống cần để thực hiện hoạt động



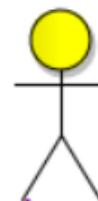
Nhân viên vận hành

*Trong hệ thống ATM*



Quản trị hệ thống

*Trong thư viện*



Hệ thống thanh toán

Bán hàng

- Những hệ thống tin học bên ngoài nào sử dụng hệ thống

## 2.3.2 Xác định Use-Case hệ thống



- Bước đầu tiên của phân tích yêu cầu là xác định các use case của hệ thống
- Một use case là một tương tác giữa hệ thống và môi trường
- Tập hợp các use case là mô tả toàn bộ hệ thống cần xây dựng



## 2.3.2 Xác định Use-Case hệ thống



- Ví dụ: phát triển phần mềm soạn thảo văn bản
- Các *use case* có thể:
  - ◆ Nhập văn bản mới
  - ◆ Sửa văn bản đã tồn tại
  - ◆ Tạo mục lục
  - ◆ Chép đoạn văn bản
  - ◆ ...



## 2.3.2 Xác định Use-Case hệ thống



- Một **use case** tương ứng với một **chức năng** của hệ thống dưới góc nhìn của người sử dụng
- Một use case có thể lớn hoặc nhỏ
- Một **use case** chỉ ra làm thế nào một **mục tiêu** của người sử dụng được thỏa mãn bởi hệ thống
- Cần phân biệt các mục tiêu của người sử dụng và các tương tác của họ với hệ thống
  - ◆ **Mục tiêu:** cái mà người sử dụng mong đợi
  - ◆ **Tương tác:** kỹ thuật cho phép đáp ứng mục tiêu



## 2.3.2 Xác định Use-Case hệ thống



- Ví dụ: phần mềm soạn thảo văn bản
  - ◆ Mục tiêu: có được một văn bản trình bày đẹp
  - ◆ Tương tác: chọn định dạng trang, chọn font chữ, định nghĩa các kiểu tiêu đề (heading), ...
- Thực tế, chúng ta xác định các *mục tiêu trước*, sau đó chọn tập hợp các tương tác đáp ứng các mục tiêu đó



## 2.3.2 Xác định Use-Case hệ thống



- Ví dụ: cần xây dựng một hệ thống ATM cho phép rút tiền
- Có thể có vài tương tác chung trong một kịch bản sau:
  - ◆ Đưa thẻ vào
  - ◆ Nhập mã PIN
  - ◆ Chọn số tiền rút
  - ◆ Khẳng định số tiền rút
  - ◆ Lấy thẻ ra
  - ◆ Lấy tiền
  - ◆ Lấy phiếu rút tiền
- Các tương tác trên có là các use case không?

## 2.3.2 Xác định Use-Case hệ thống



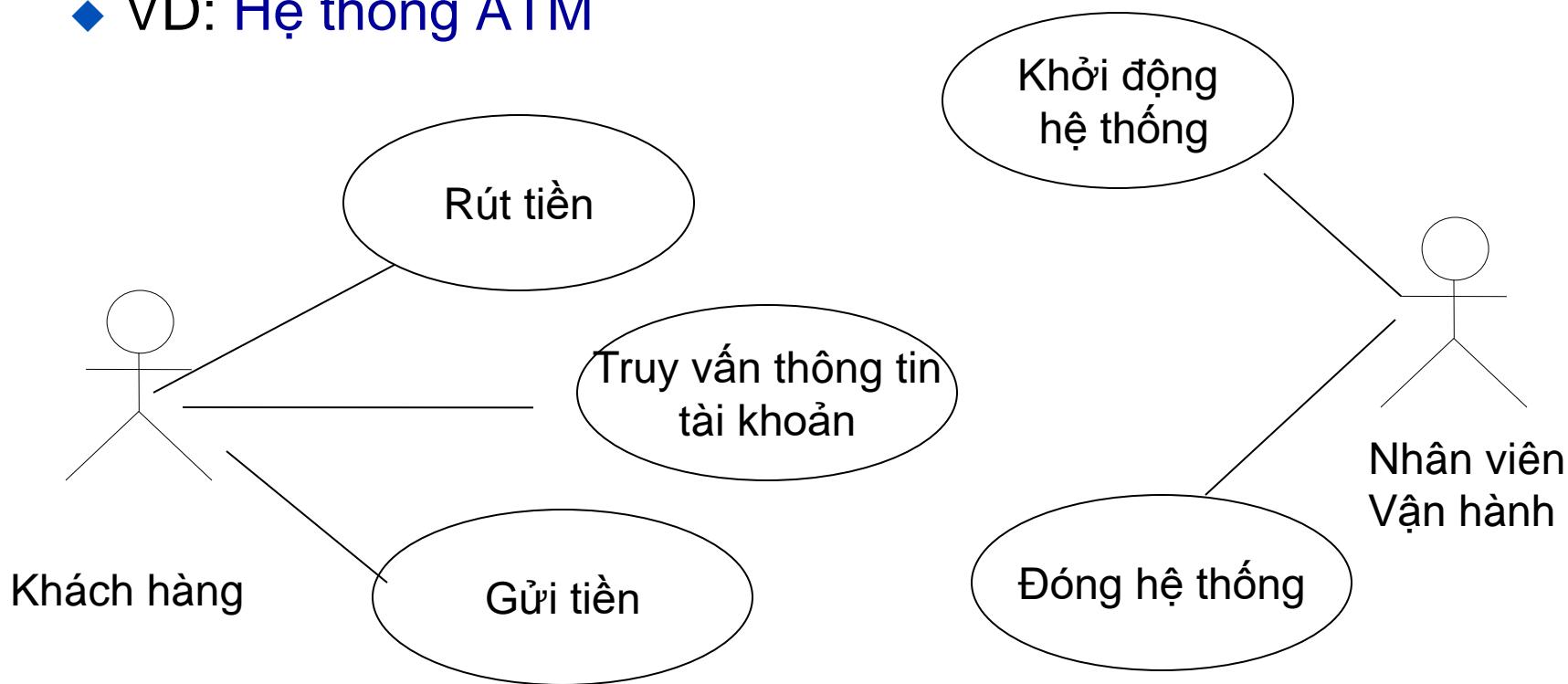
- Phương pháp xác định Use-Case:
  - ◆ Xác định Use-Case thông qua các tác nhân

## 2.3.2 Xác định Use-Case hệ thống



### ■ Xác định Use-Case thông qua các **tác nhân**

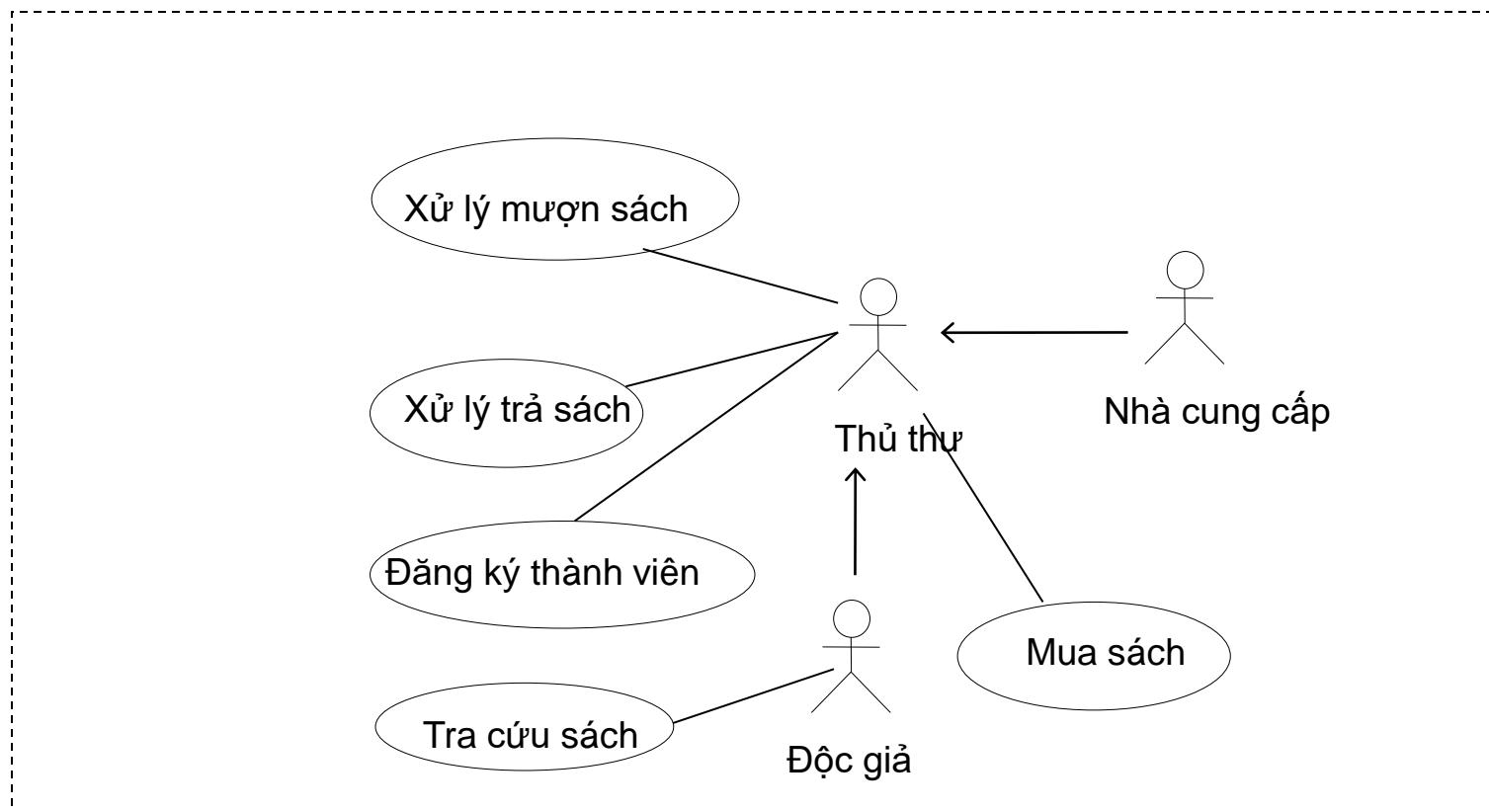
- ◆ Tìm các nhiệm vụ và **chức năng** mà tác nhân sẽ thi hành hoặc hệ thống cần tác nhân để thi hành và mô hình hóa nó như là **Use-Case**
- ◆ VD: Hệ thống ATM



## 2.3.2 Xác định Use-Case hệ thống



- Xác định Use-Case thông qua các **tác nhân**
  - ◆ VD: Hệ thống Thư viện



## 2.3.2 Xác định Use-Case hệ thống



### ■ Xác định Use-Case thông qua các **tác nhân**

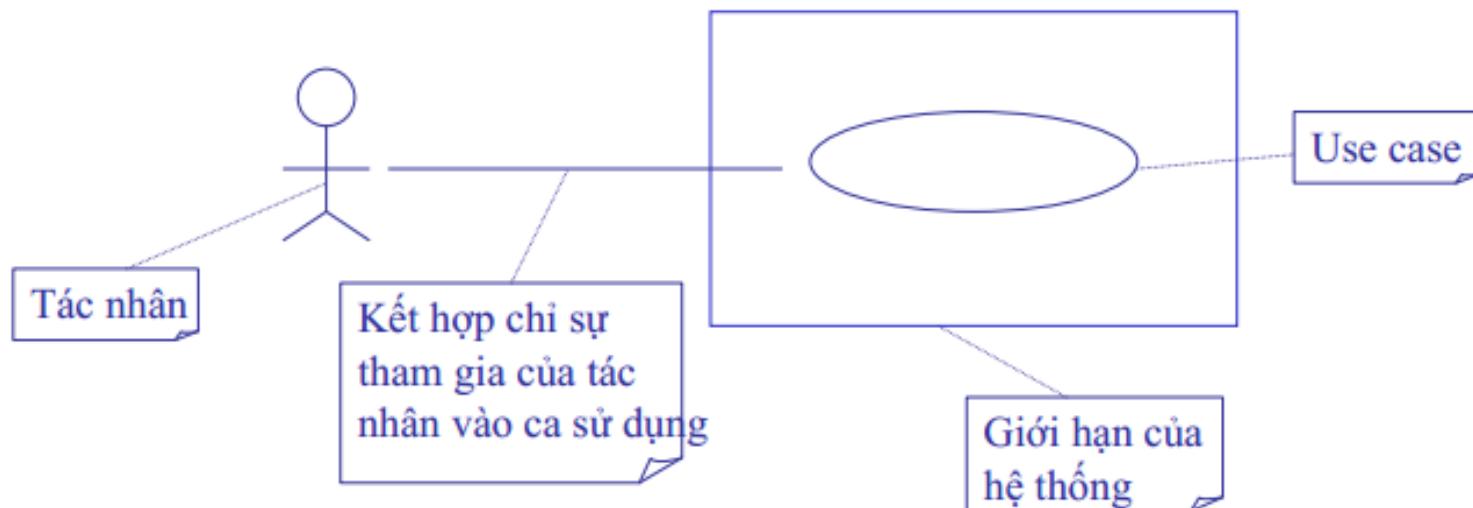
- ◆ Đặt tên Use-Case: tên Use-case nên đặt nhằm phản ánh một mô tả tổng quan về chức năng của use-case
- ◆ Tên Use-case = **động từ** (do) + **danh từ** (what)



## 2.3.3 Sơ đồ Use-Case hệ thống



- Sơ đồ Use-Case mô tả quan hệ giữa các tác nhân và use case của một hệ thống



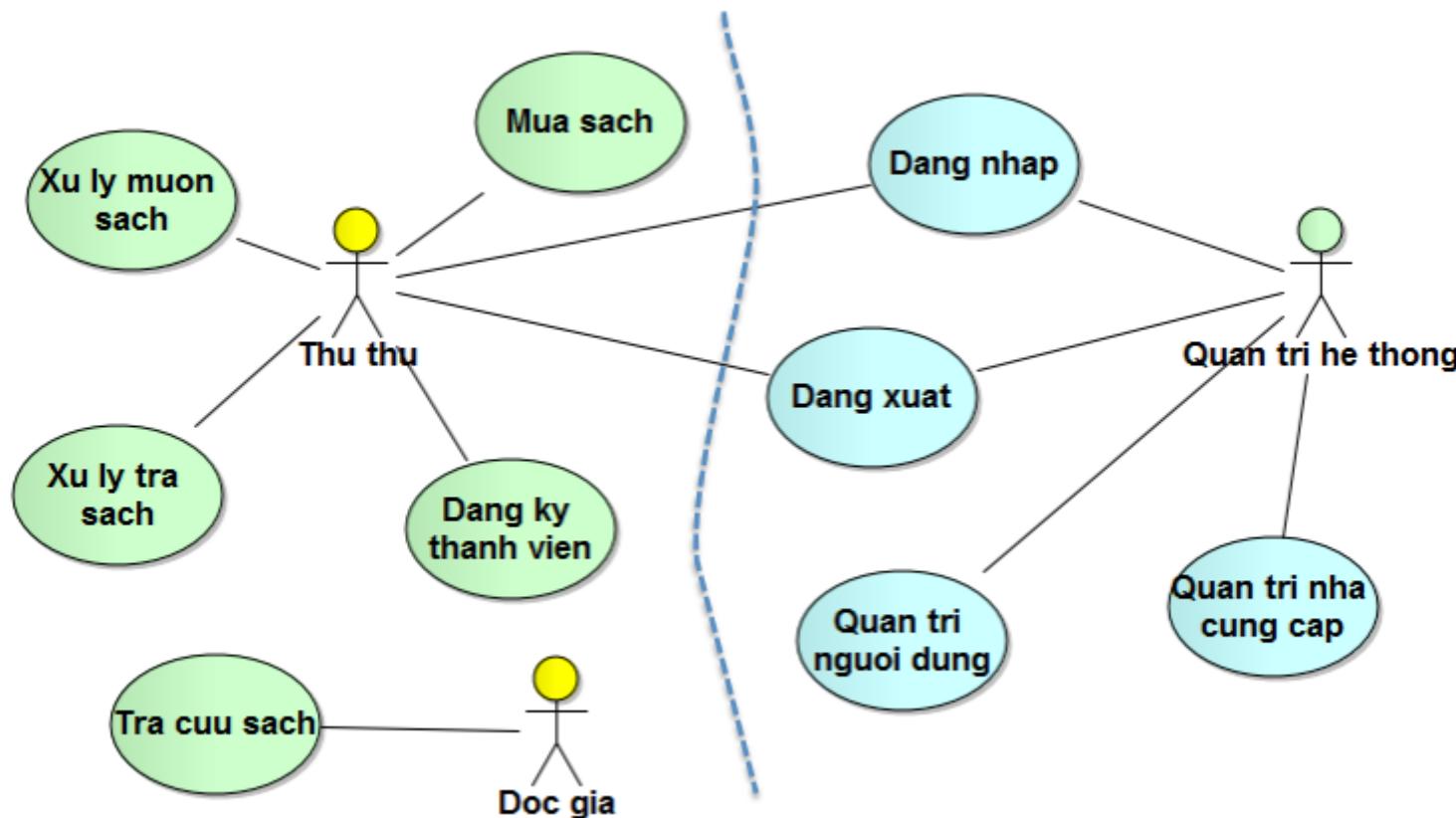
## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

- Bổ sung các Use-Case mô tả chức năng đặc trưng phần mềm:
  - ◆ Các Use-Case quản trị hệ thống:
    - Quản trị người dùng
  - ◆ Các Use-Case quản trị dữ liệu:
    - Quản trị danh mục dữ liệu
      - VD: quản lý danh mục sách, danh mục nhà cung cấp, ...
      - Quản trị an toàn (backup / restore)
  - ◆ ....



## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

- Bổ sung các Use-Case mô tả chức năng đặc trưng phần mềm:
  - ◆ Ví dụ: Trong Hệ thống Thư viện

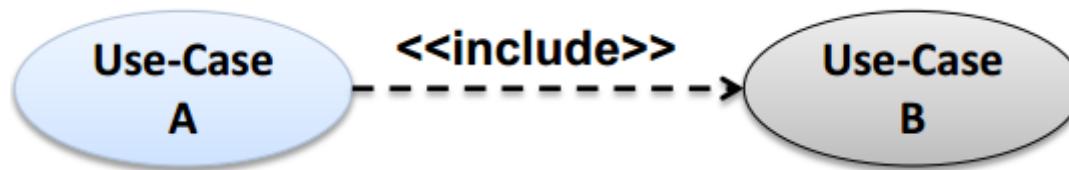


## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

- Phát triển các mối liên kết <<include>> và <extend>>:

- ♦ **Include**: Use-Case A <<include>> Use-Case B:

- Trong quá trình thực thi Use-case A sẽ luôn cần thực hiện Use-case B



- ♦ **Extend**: Use-Case A <<extend>> Use-Case B:

- Trong quá trình thực thi Use-case A, trong một số trường hợp sẽ thực hiện Use-case B



## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

- Phát triển các mối liên kết <<include>>:

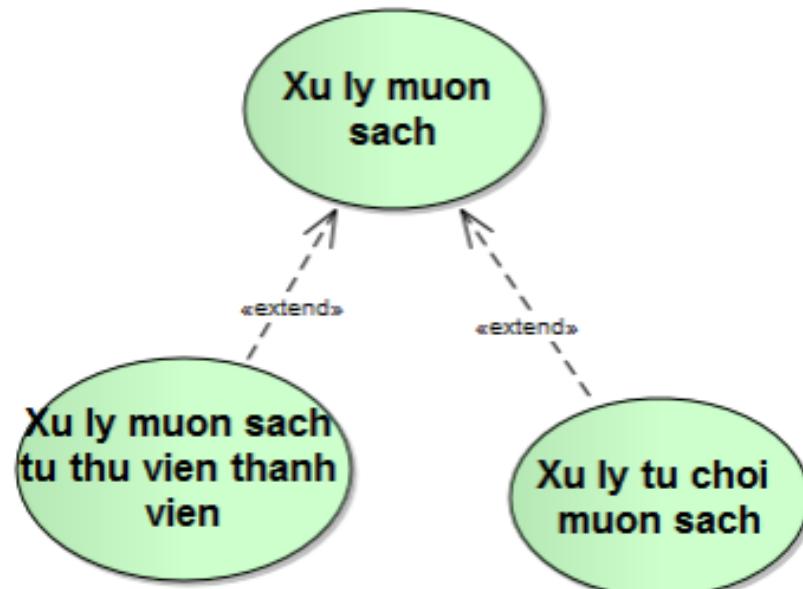
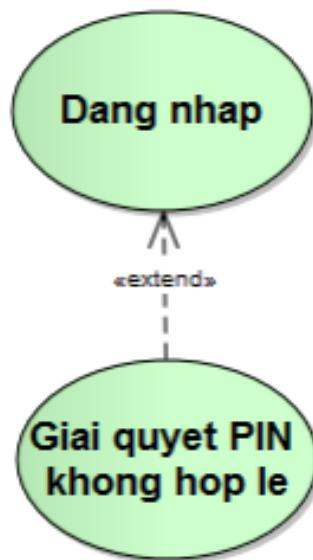


*Hệ thống ATM*

## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

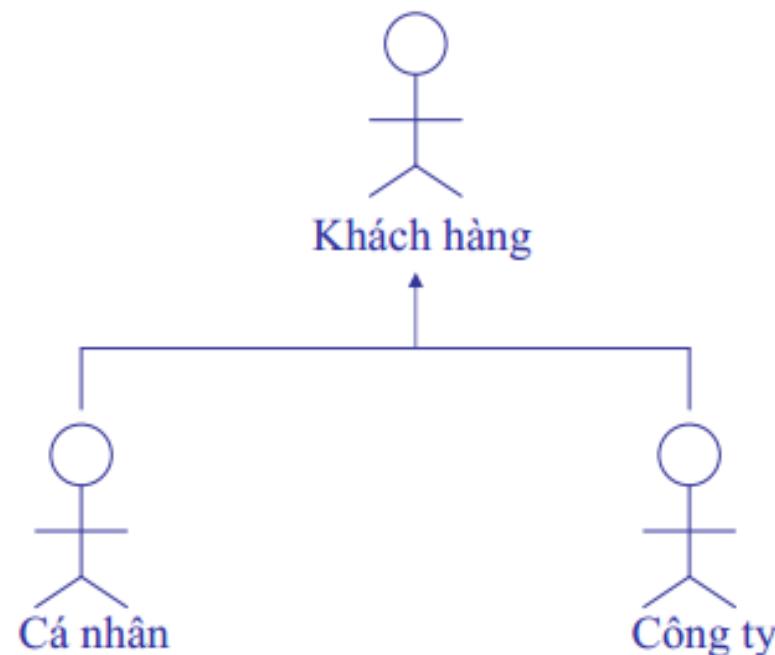
### ■ Phát triển các mối liên kết *<<extend>>*:

- ◆ Tách những dòng xử lý đặc biệt → Use-case mở rộng



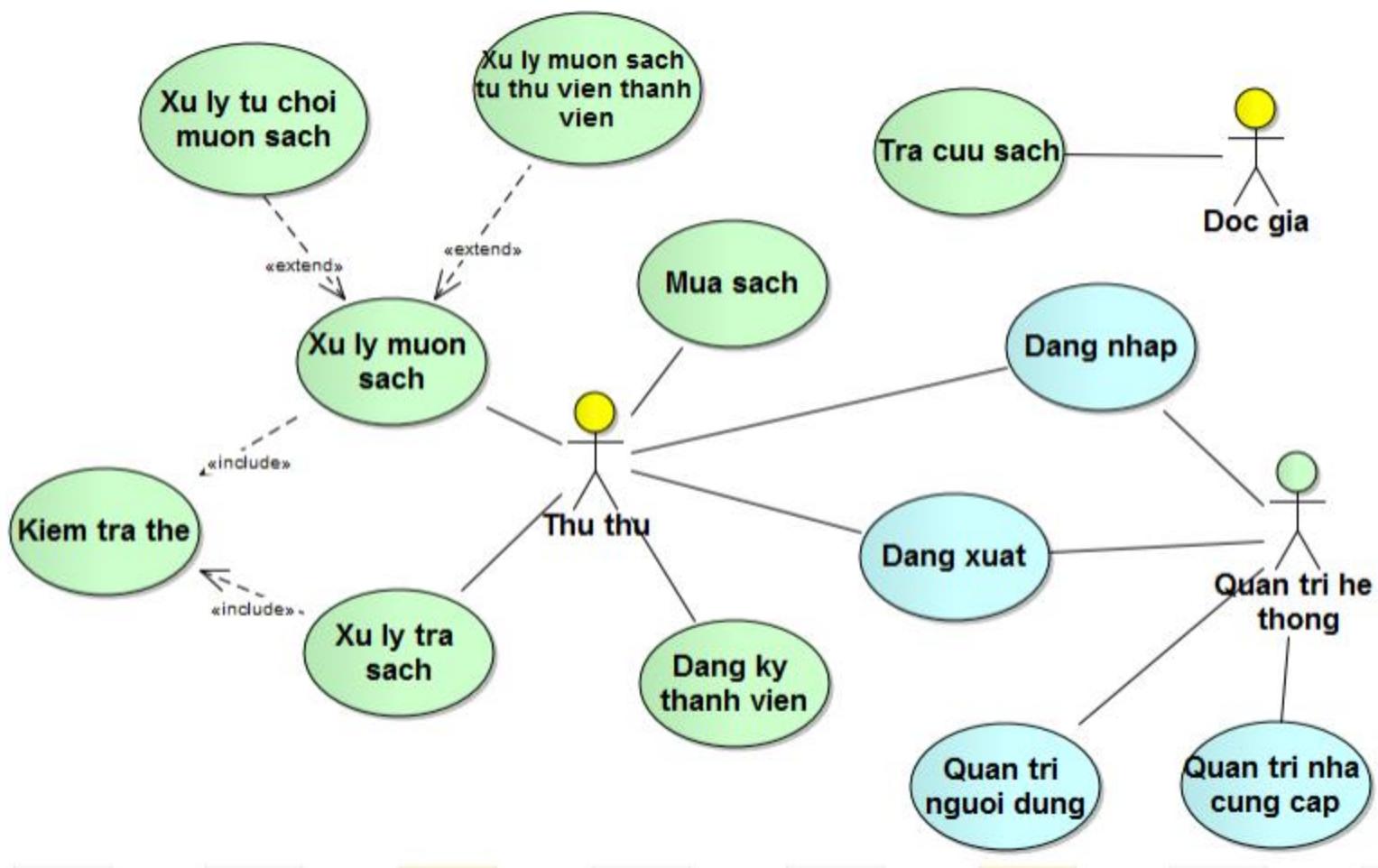
## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

- Các tác nhân có mối quan hệ thừa kế



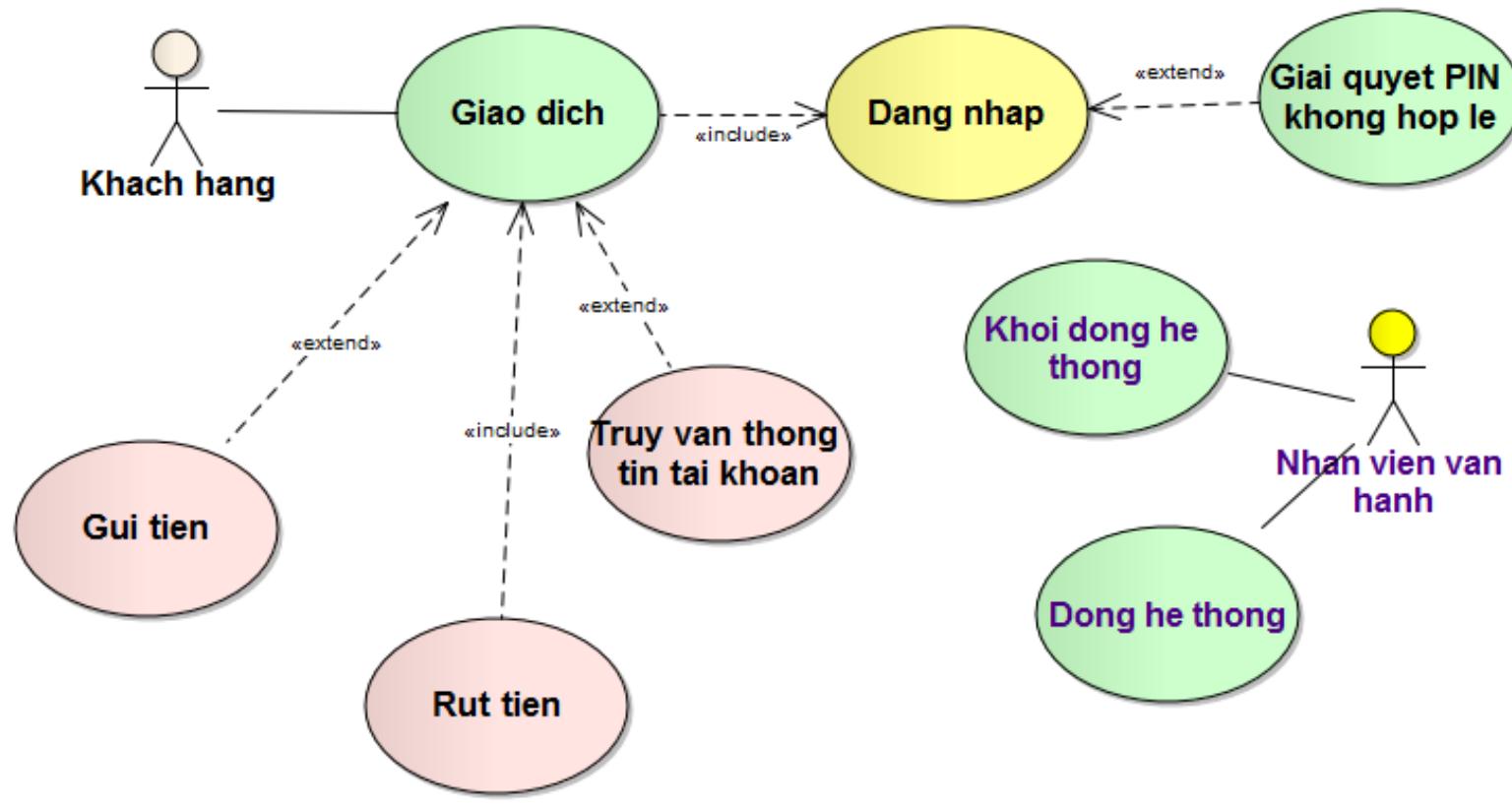
## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

### Kết quả sơ đồ Use-Case hệ thống Thư viện



## 2.3.4 Tinh chế và nâng cấp sơ đồ Use-Case

### Kết quả sơ đồ Use-Case hệ thống ATM



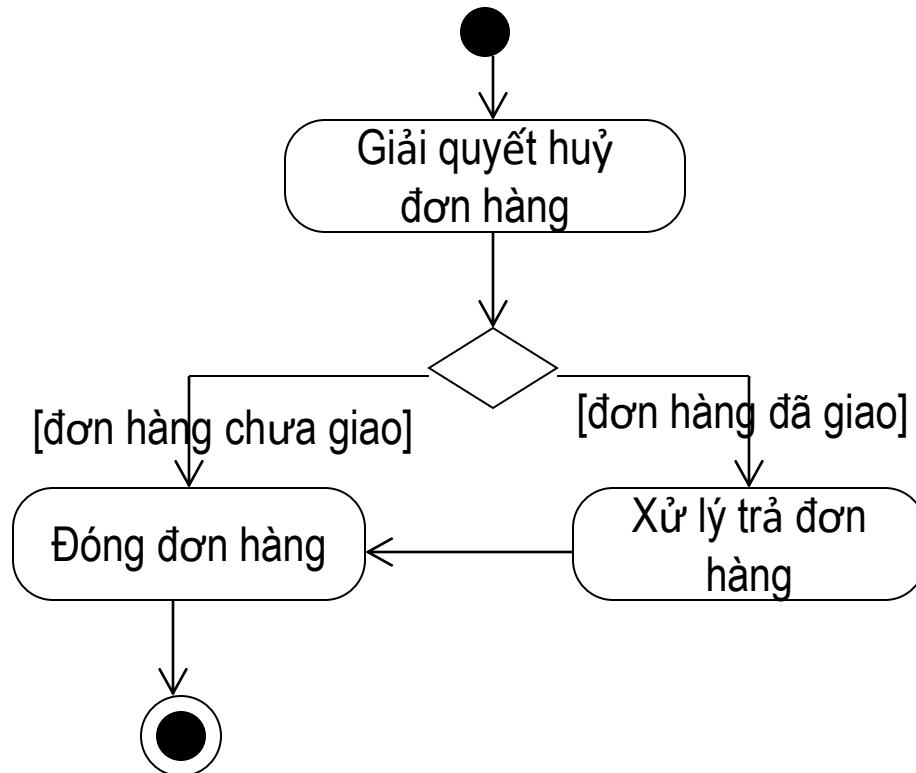
## 2.4 Ràng buộc nghiệp vụ (Business rule)

- Là những quy tắc, quy định xuất hiện trong các quy trình nghiệp vụ
- Có 4 loại ràng buộc:
  - ◆ Nguyên tắc kích hoạt và phản ứng
  - ◆ Nguyên tắc ràng buộc thao tác
  - ◆ Nguyên tắc ràng buộc cấu trúc
  - ◆ Nguyên tắc diễn dịch:
    - Nguyên tắc suy luận
    - Nguyên tắc tính toán

## 2.4 Ràng buộc nghiệp vụ (Business rule)

### ■ Nguyên tắc kích hoạt và phản ứng:

- ◆ Là những ràng buộc hay điều kiện xác định khi nào một hành động xảy ra
- ◆ Ví dụ: ràng buộc trong hoạt động huỷ một đơn hàng



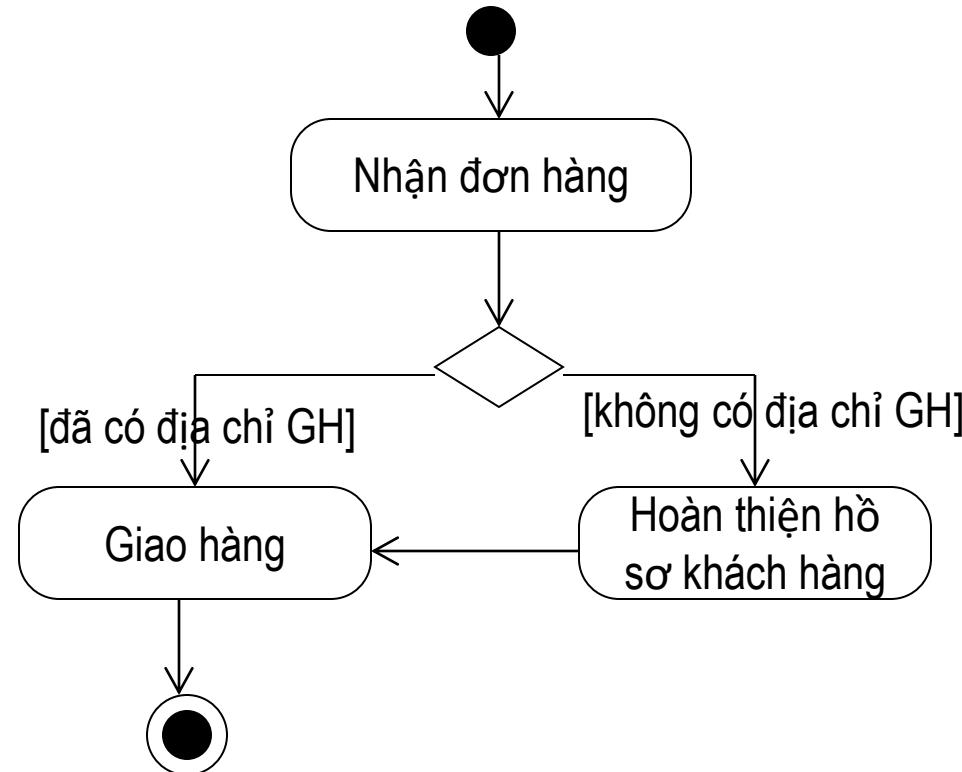
WHEN một đơn đặt hàng bị hủy bỏ  
IF hàng chưa được vận chuyển  
THEN kết thúc đơn đặt hàng



## 2.4 Ràng buộc nghiệp vụ (Business rule)

### ■ Nguyên tắc ràng buộc thao tác

- ◆ Đây là những điều kiện phải thỏa trước và sau thao tác để bảo đảm thao tác đó hoạt động đúng
- ◆ Ví dụ: Trong một tổ chức quản lý đặt hàng, nguyên tắc sau đây có thể xảy ra



Vận chuyển Hàng hóa đến chỗ Khách hàng  
ONLY IF Khách hàng có địa chỉ



## 2.4 Ràng buộc nghiệp vụ (Business rule)

### ■ Nguyên tắc ràng buộc cấu trúc

- ◆ Xác định các quy định và điều kiện về các lớp, đối tượng, và các mối quan hệ giữa chúng
- ◆ Ví dụ:



**IT MUST ALWAYS HOLD THAT**

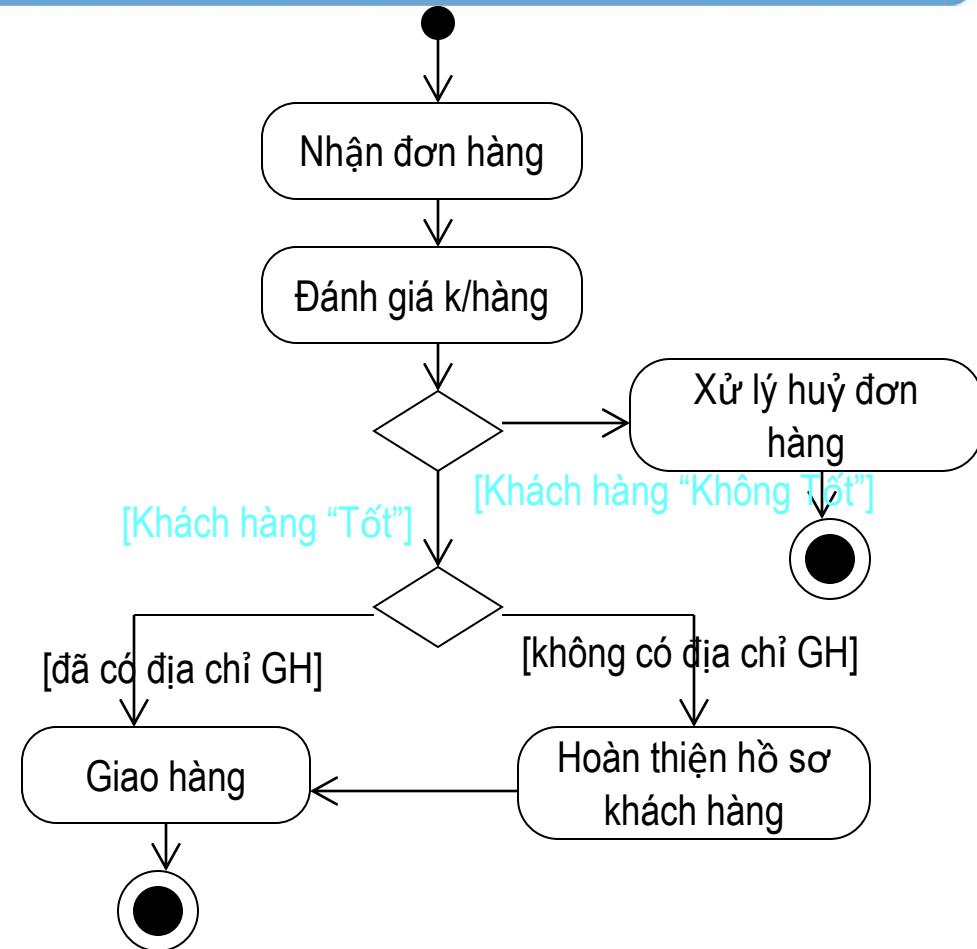
Một đơn đặt hàng tham chiếu đến tối thiểu một sản phẩm



## 2.4 Ràng buộc nghiệp vụ (Business rule)

### ■ Nguyên tắc suy luận:

- ◆ Xác định rằng nếu một số sự kiện nhất định là đúng, một kết luận có thể được suy ra
- ◆ Ví dụ: Thiết lập nguyên tắc sau đây để xác định trạng thái của một khách hàng



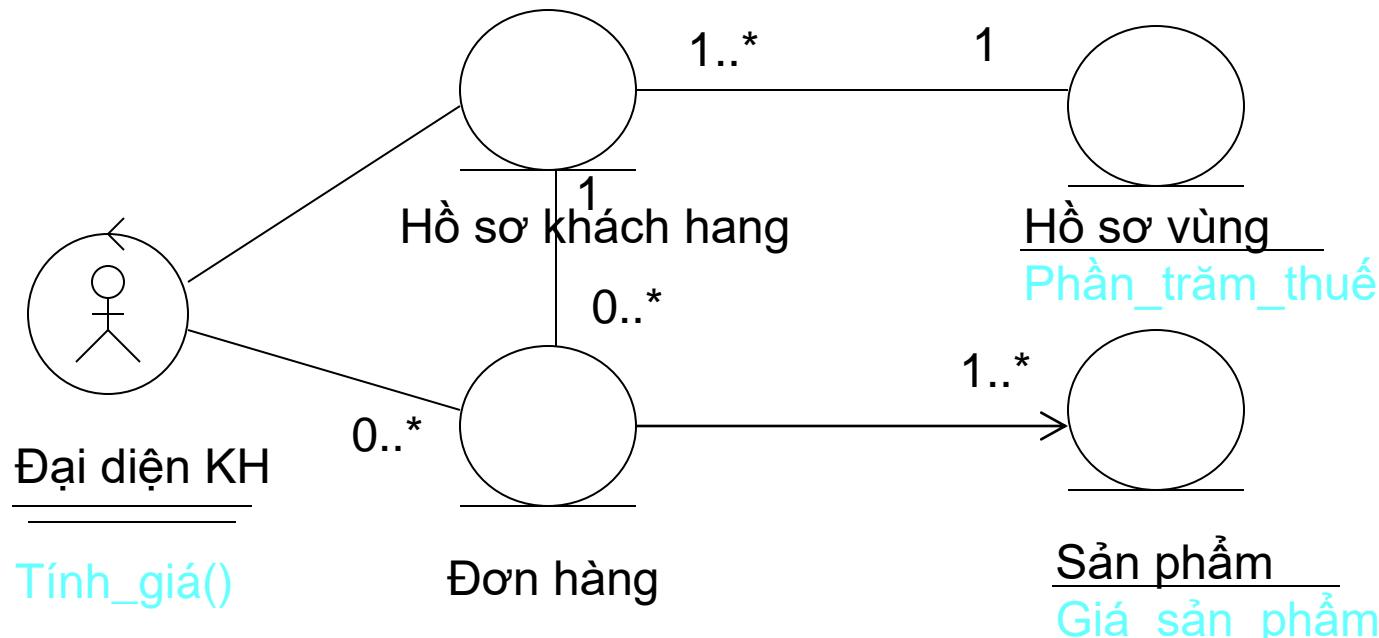
Một Khách hàng là một Khách hàng “**Tốt**” **IF AND ONLY IF**  
Những hóa đơn chưa thanh toán gửi đến Khách hàng đều ít hơn 30 ngày



## 2.4 Ràng buộc nghiệp vụ (Business rule)

### ■ Nguyên tắc tính toán:

- ◆ Tạo ra kết quả bằng cách xử lý các thuật toán.



Giá một sản phẩm được tính toán như sau:  
giá sản phẩm \* (1 + phần trăm thuế/ 100)





## Đánh giá kết quả

- Tất cả các chức năng cần thiết đã được xác định chưa?
- Có xác định được Use-Case dư thừa nào không?
- Hành vi của mỗi Use-Case có theo đúng thứ tự không?
- Luồng công việc của mỗi Use-Case có hoàn chỉnh không?
- Tìm thấy được tất cả các Use-Case ?

# Kết quả quá trình phân tích hệ thống

## ■ Hiện trạng tổ chức

- ◆ Nắm bắt thông tin về tổ chức
- ◆ Các đối tượng có liên quan và khách hàng
- ◆ Giới hạn hệ thống phát triển
- ◆ Các vấn đề của hệ thống

## ■ Bảng mô tả các thuật ngữ nghiệp vụ

## ■ Mô hình Use-Case hệ thống

- ◆ Xác định các tác nhân và Use-Case hệ thống
- ◆ Sơ đồ Use-Case hệ thống

## ■ Mô tả các ràng buộc



# Bài tập 1



Một thư viện có nhu cầu quản lý thông tin mượn sách của độc giả. Quy trình đã mô tả như sau:

Khi độc giả có nhu cầu mượn sách, thủ thư tiến hành kiểm tra thẻ thư viện, nếu thẻ không hợp lệ (hết hạn) hoặc chưa có thẻ thư viện, thủ thư hướng dẫn độc giả làm thẻ thư viện.

Quy trình đăng ký thẻ như sau: trước tiên độc giả điền đầy đủ thông tin về họ tên, địa chỉ, CMND, ... vào mẫu đăng ký và gửi thông tin cho thủ thư, đóng tiền và thủ thư hẹn 30 ngày độc giả nhận thẻ. Tiếp theo thủ thư nhận thông tin đăng ký tiến hành lưu trong hồ sơ thông tin độc giả, chuyển thông tin độc giả đến bộ phận kỹ thuật làm thẻ, sau khi hoàn thành thủ thư tiến hành cập nhật tình trạng hồ sơ đăng ký của độc giả. Kết thúc quá trình đăng ký thẻ thư viện. Ngược lại, nếu độc giả đã có thẻ thư viện hợp lệ, thủ thư kiểm tra độc giả vi phạm quy định 1 (mượn sách quá hạn và không trả sách) hay không. Nếu vi phạm, thủ thư từ chối mượn sách, ngược lại tiến hành tìm sách và cập nhật tình trạng mượn sách trong hồ sơ giao sách cho độc giả, kết thúc quá trình mượn sách.

*“Quy định 1: Độc giả mượn sách phải trả sách trong vòng 7 ngày.”*



## Bài tập 2



Quản lý đào tạo nhân viên: Một công ty muốn mô tả bằng UML việc đào tạo nhân viên để tin học hóa một số công việc.

Việc đào tạo được bắt đầu khi người quản lý đào tạo nhận được yêu cầu đào tạo của một nhân viên. Nhân viên này có thể xem danh mục các chuyên đề đào tạo của các đơn vị đào tạo ký kết với công ty.

Yêu cầu của nhân viên được xem xét bởi người quản lý đào tạo và người quản lý sẽ trả lời là chấp nhận hay từ chối đề nghị đó. Trong trường hợp chấp nhận, người quản lý sẽ xác định chuyên đề phù hợp trong danh mục các chuyên đề, sau đó gửi cho nhân viên nội dung của chuyên đề và danh sách các khóa đào tạo. Nhân viên sẽ chọn khóa đào tạo và người quản lý sẽ đăng ký khóa học với đơn vị đào tạo cho nhân viên. Trong trường hợp muốn hủy bỏ đăng ký khóa đào tạo, nhân viên phải thông báo sớm cho người quản lý biết để người quản lý thực hiện hủy bỏ. Cuối khóa đào tạo, nhân viên chuyển phiếu đánh giá kết quả học về cho công ty. Người quản lý sẽ kiểm tra hóa đơn thanh toán tiền của đơn vị đào tạo.

Xây dựng biểu đồ ca sử dụng.

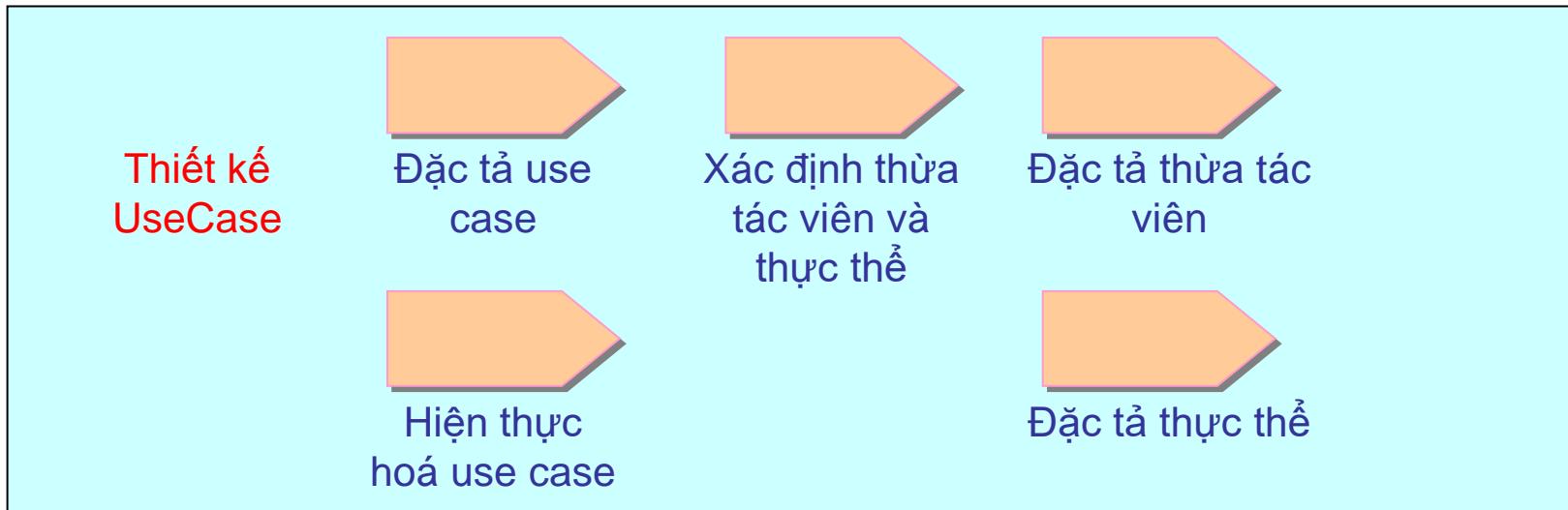


# Hướng dẫn thực hành



- Sử dụng phần mềm Enterprise Architect
- Vẽ mô hình Use-Case hệ thống cho quản lý mượn sách thư viện

### 3. Thiết kế Use Case



### 3. Thiết kế Use Case



- Đặc tả Use-case hệ thống
- Xác định thừa tác viên và thực thể
  - ◆ Thừa tác viên (worker)
  - ◆ Thực thể nghiệp vụ (entity)
- Hiện thực hóa các Use-case hệ thống bằng
  - ◆ Sơ đồ hoạt động (Activity Diagram)
  - ◆ Sơ đồ tuần tự (Sequence Diagram)
  - ◆ Sơ đồ cộng tác (Collaboration Diagram & Communication)



### 3.1 Đặc tả Use-Case hệ thống



- Xác định tương tác giữa tác nhân và Use-case hệ thống
- Mô tả luồng công việc chính
- Những luồng công việc bất thường và tùy chọn
  - ◆ Những luồng sự kiện con tham gia phần lớn vào luồng sự kiện chính
  - ◆ Những luồng sự kiện bất thường giúp cho luồng sự kiện chính rõ ràng hơn.
  - ◆ Những luồng sự kiện con xảy ra ở những khoảng thời gian khác nhau trong cùng một luồng công việc và chúng có thể được thực thi

# 3.1 Đặc tả Use-case hệ thống

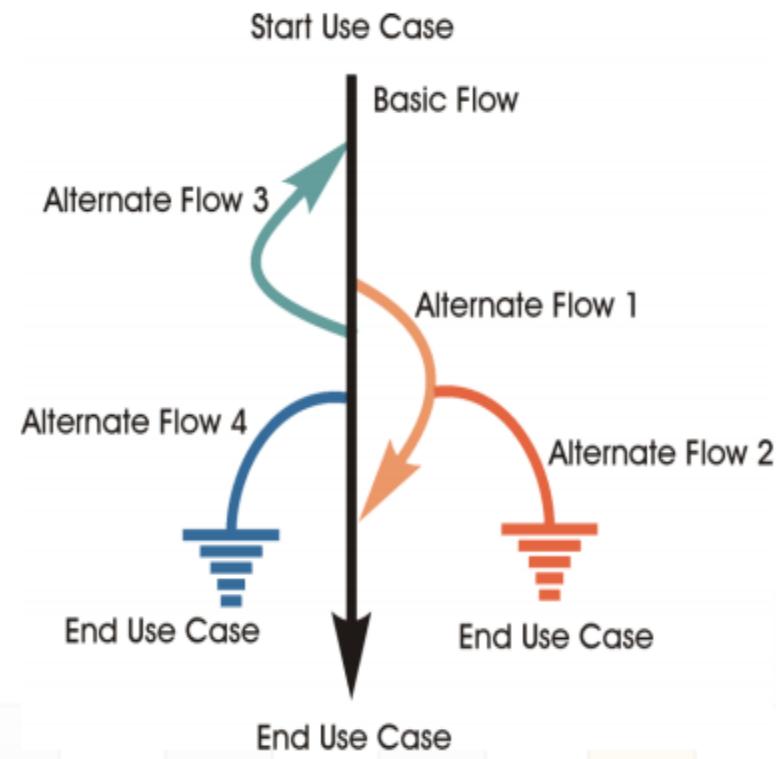


## ■ Mẫu đặc tả

*Giới thiệu về Use case:* trình bày mục đích của Use case, các điều kiện để Use case được kích hoạt

*Các dòng cơ bản (basic flow):* Bao gồm các hoạt động chính và thứ tự mô tả nội dung chính của Use case

*Các dòng thay thế (alternative flow):* mô tả các nhánh hoạt động bất thường để xử lý ngoại lệ của các dòng chính.



### 3.1 Đặc tả Use-case hệ thống



#### ■ Mẫu mô tả Use-case:

**Tóm tắt:** Tóm tắt nội dung hoạt động của use case

**Tác nhân liên quan:** Các tác nhân liên quan

**Tiền điều kiện (pre-condition):** mô tả trạng thái của hệ thống phải đạt được để use case có thể bắt đầu

**Các Use case liên quan:** Các use case liên quan với use case này.

**Dòng sự kiện chính (basic flow):** Các sự kiện và hoạt động xảy ra bình thường của use case

**Dòng sự kiện phụ (alternative flows):** Các sự kiện và những hoạt động bất thường của use case ngoài những hoạt động chính

**Hậu điều kiện (post-condition):** liệt kê các trạng thái có thể của hệ thống tại cuối use case. Hệ thống phải thuộc một trong những trạng thái đó khi use case kết thúc.



### 3.1 Đặc tả Use-case hệ thống



#### ■ VD1 Mô tả Use-case Giao dịch:

**Tóm tắt:** use case bắt đầu khi có khách hàng đưa thẻ vào máy ATM. Hệ thống sẽ cho phép khách hàng thực hiện các giao dịch.

##### **Dòng sự kiện chính:**

1. Xác nhận thẻ và kiểm tra đăng nhập của khách hàng (thực hiện bởi Use case đăng nhập)
2. Hiển thị giao diện thực đơn yêu cầu khách hàng chọn dịch vụ thực hiện.
3. Khách hàng chọn dịch vụ thực hiện.
4. Thi hành dịch vụ tương ứng khi khách hàng chọn (rút tiền, gửi tiền, xem thông tin tài khoản: được mô tả trong từng use case)
5. Đóng giao diện thực đơn và thông báo khách hàng rút thẻ.
6. Khách hàng rút thẻ và hệ thống trở về trạng thái sẵn sàng.



### 3.1 Đặc tả Use-case hệ thống



#### ■ VD1 Mô tả Use-case Giao dịch (tt):

##### *Dòng sự kiện phụ:*

**A1**- Xử lý thẻ không hợp lệ (Xảy ra khi thẻ không thuộc hệ thống ngân hàng)

1. Hệ thống thu hồi thẻ ATM của khách hàng
2. Thông báo thu hồi thẻ

**A2** - Đăng nhập không thành công (Xảy ra khi khách hàng nhập sai PIN): Hệ thống trả thẻ cho khách hàng.

**A3** - Xử lý đăng nhập quá 3 lần (Xảy ra khi khách hàng nhập sai PIN quá 3 lần): Hệ thống thu hồi thẻ ATM của khách hàng

**A4**- Hủy giao dịch (Xảy ra khi khách hàng chọn hủy giao dịch trong lúc đang thực hiện giao dịch): Hệ thống hủy giao dịch và trả thẻ cho khách hàng

**Tiền điều kiện:** ATM phải kết nối với mạng ngân hàng.

##### *Hậu điều kiện:*

- Khách hàng thực hiện giao dịch thành công. Nếu thực hiện giao dịch không thành công thì phải có thông báo lỗi tương ứng.



### 3.1 Đặc tả Use-case hệ thống



#### ■ VD2 Mô tả Use-case Rút tiền:

##### Tóm tắt:

Use case bắt đầu khi khách hàng chọn thực hiện rút tiền từ thực đơn chính. Hệ thống sẽ xử lý yêu cầu rút tiền của khách hàng

##### Dòng sự kiện chính:

1. Yêu cầu xác định tài khoản
2. Yêu cầu nhập số tiền cần rút
3. Khách hàng nhập số tiền
4. Kiểm tra số tiền nhập với số dư tài khoản
5. Chuyển tiền ra cho khách hàng
6. In hóa đơn rút tiền cho khách hàng



### 3.1 Đặc tả Use-case hệ thống



#### ■ VD2 Mô tả Use-case Rút tiền (tt):

##### *Dòng sự kiện phụ:*

**A1 – Tiền rút vượt quá số dư**

Nếu số tiền vượt qua số dư thì thông báo cho khách hàng điều chỉnh hoặc hủy bỏ giao dịch.

**A2 – Không in được hóa đơn (do hết giấy in hóa đơn)**

Hệ thống thông báo cho khách hàng không in được hóa đơn rút tiền.

**A3 – Không đủ tiền trong máy**

Hệ thống báo lỗi không thể rút tiền được.

**A4 – Khách hàng nhập số tiền không đúng quy định (x50k)**

Hệ thống yêu cầu khách hàng nhập lại số tiền

##### *Tiền điều kiện:*

Khách hàng phải đăng nhập

##### *Hậu điều kiện:*

Khách hàng rút tiền thành công, tài khoản được cập nhật hoặc hệ thống thông báo lỗi nếu không rút tiền được.



## 3.2 Xác định thừa tác viên và thực thể



### ■ Thừa tác viên (worker)

- ◆ Một thừa tác viên biểu diễn sự trừu tượng của một người hoạt động trong nghiệp vụ.
- ◆ Một thừa tác viên có thể tương tác với các thừa tác viên khác và thao tác với các thực thể để hiện thực hóa hoạt động của một Use-Case
- ◆ Thừa tác viên hoạt động khi luồng công việc của một thể hiện Use-case được bắt đầu.
- ◆ Có 2 loại thừa tác viên:
  - Thừa tác viên thực hiện các công việc bên trong hệ thống
  - Thừa tác viên tương tác với các tác nhân bên ngoài hệ thống

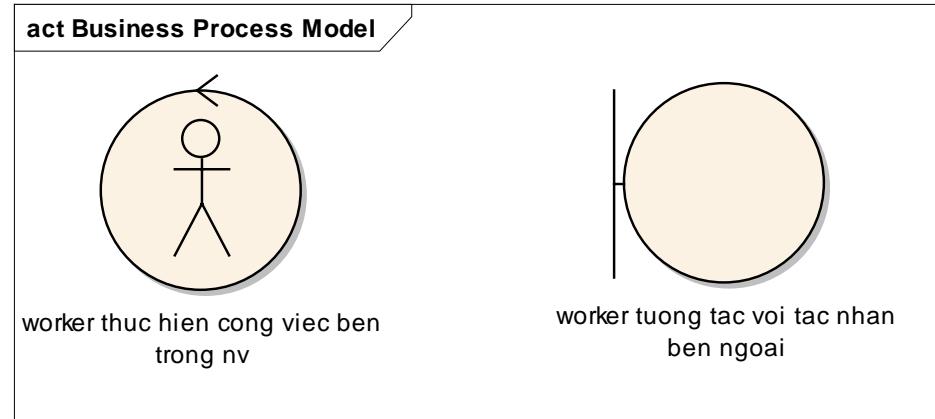


## 3.2 Xác định thừa tác viên và thực thể

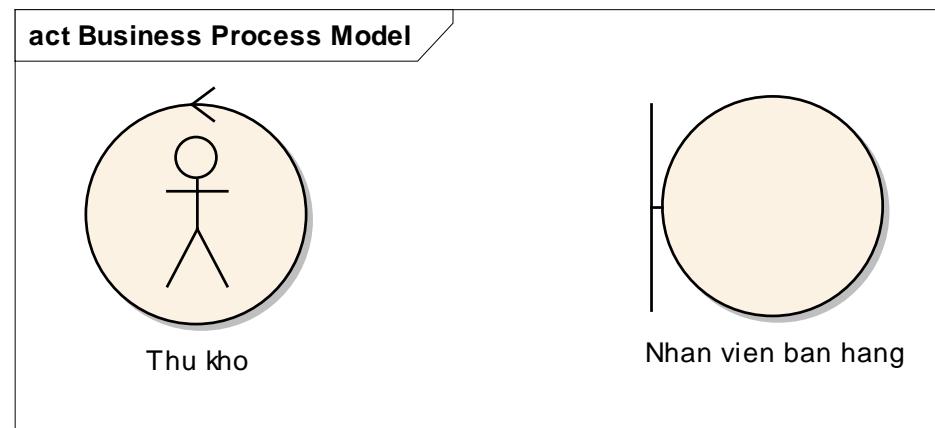


### ■ Thừa tác viên (worker)

- ◆ ký hiệu:



- ◆ Ví dụ:



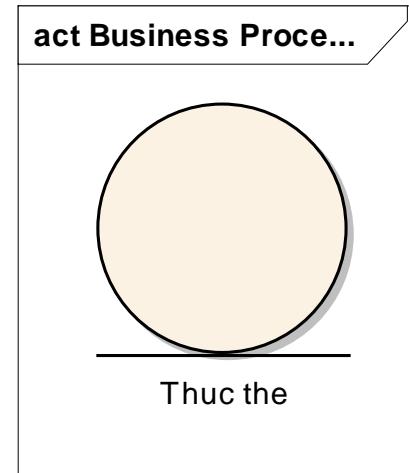
## 3.2 Xác định thừa tác viên và thực thể



### ◆ Thực thể (entity):

- ◆ Biểu diễn những sự vật được thừa tác viên sử dụng để thực hiện nghiệp vụ.

### ◆ Ký hiệu:



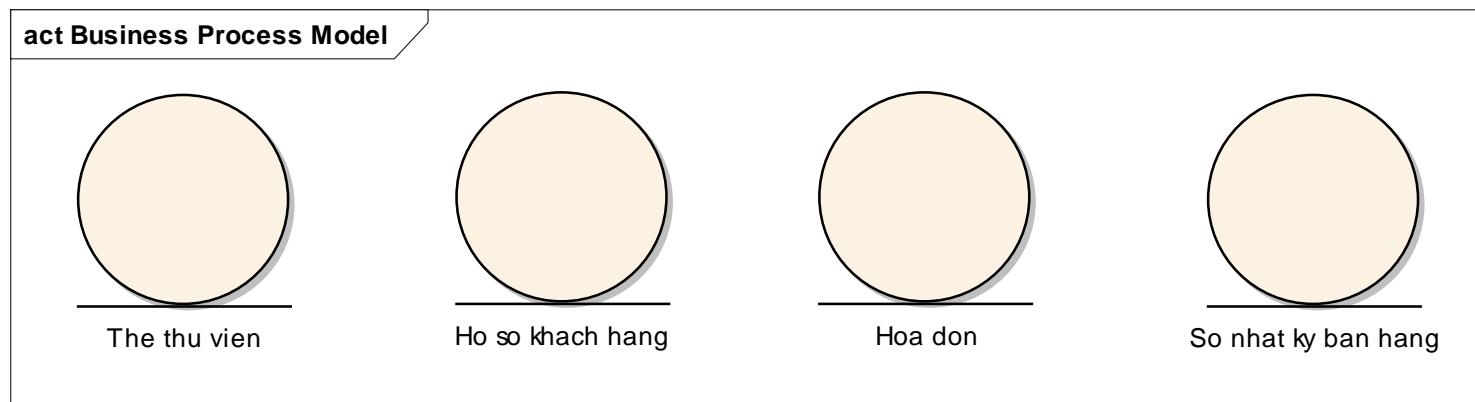
## 3.2 Xác định thừa tác viên và thực thể



### ◆ Thực thể (entity):

#### ◆ Sự vật có thể là:

- ◆ **Đối tượng thông tin:** các đối tượng dùng để chứa thông tin dữ liệu hệ thống như là: sổ sách, chứng từ, hồ sơ, giấy tờ, thẻ, báo cáo, tập tin, CSDL,...

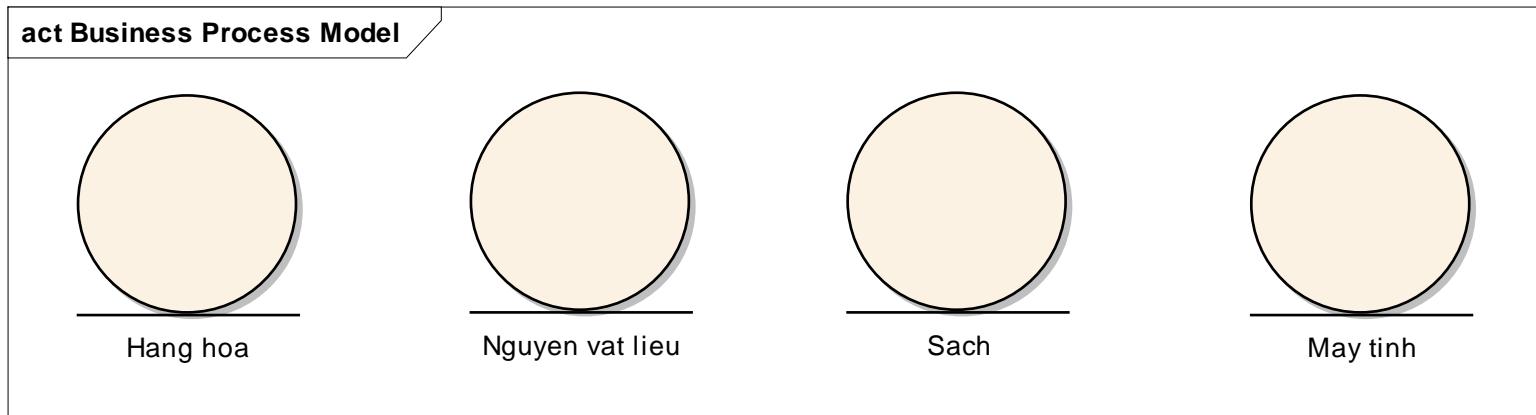


## 3.2 Xác định thừa tác viên và thực thể

### ◆ Thực thể (entity):

#### ◆ Sự vật có thể là:

◆ **Đối tượng sự vật:** các đối tượng mô tả các sự vật trong hoạt động nghiệp vụ như là: các đối tượng trong quá trình sản xuất, các trang thiết bị, ...



## 3.2 Hiện thực hóa Use-Case hệ thống



### ■ Hiện thực hóa Use-Case hệ thống (Use-Case realization):

- ◆ Mô tả cách thức một Use-case cụ thể được hiện thực hóa bên trong mô hình đối tượng dưới dạng các đối tượng cộng tác với nhau thực hiện hoạt động của Use-case.
- ◆ Hiện thực hóa Use-case hệ thống được đặc tả với:
  - Sơ đồ hoạt động (**Activity Diagram**)
  - Sơ đồ tuần tự(**Sequence Diagram**)
  - Sơ đồ cộng tác (**Collaboration Diagram** hoặc **Communication Diagram**)



## 3.2 Hiện thực hóa Use-Case hệ thống



### ■ Sơ đồ hoạt động (Activity Diagram)

- ◆ Sắp xếp các công việc theo thứ tự xử lý một hệ thống, cho thấy vai trò và các thực thể tham gia vào hoạt động.

### ■ Sơ đồ tuần tự (Sequence Diagram)

- ◆ Mô tả trình tự các sự kiện của quy trình.

### ■ Sơ đồ cộng tác (Collaboration Diagram)

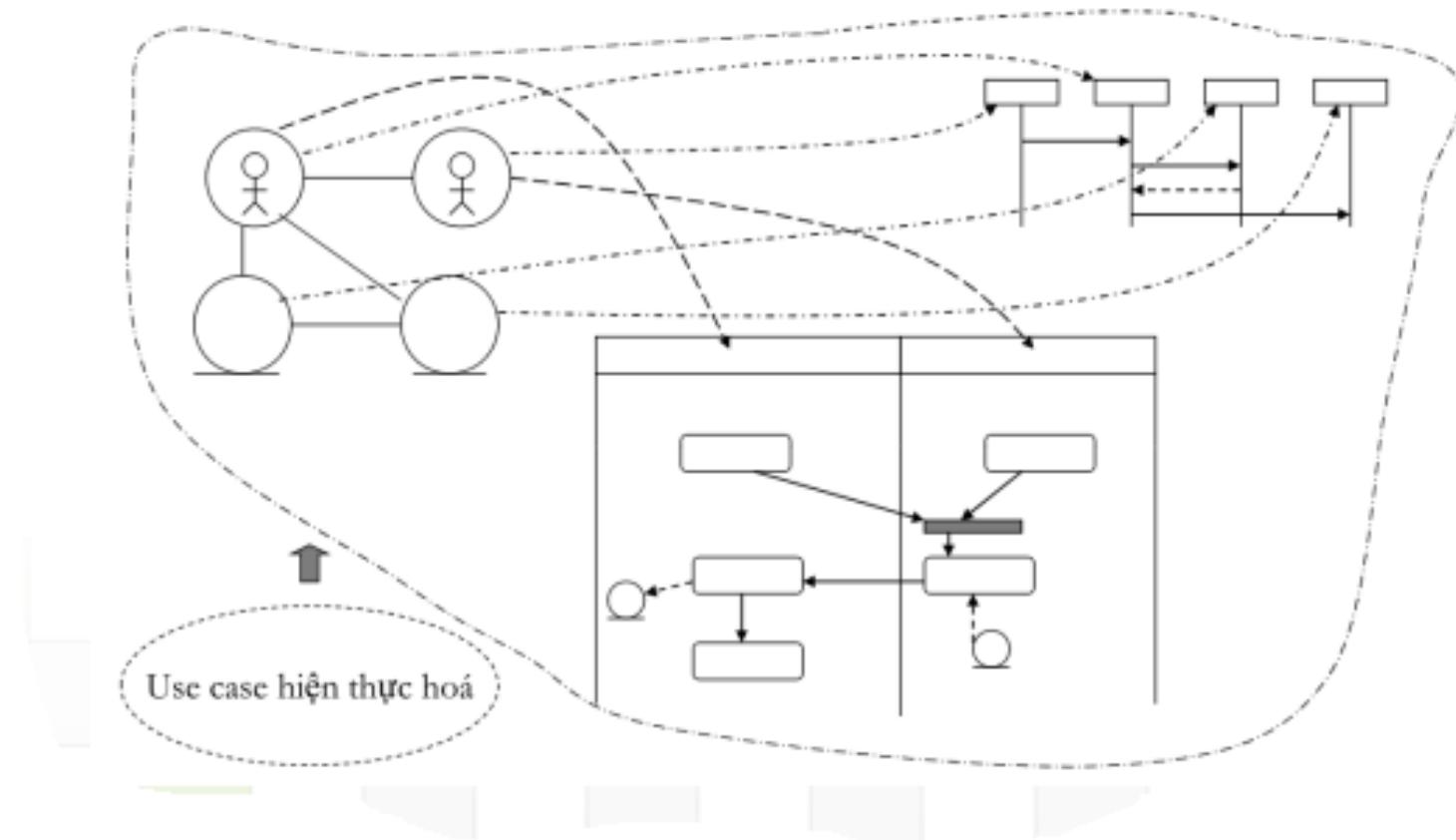
- ◆ Trình bày các liên kết giao tiếp và những thông điệp giữa các đối tượng.



## 3.2 Hiện thực hóa Use-Case nghiệp vụ



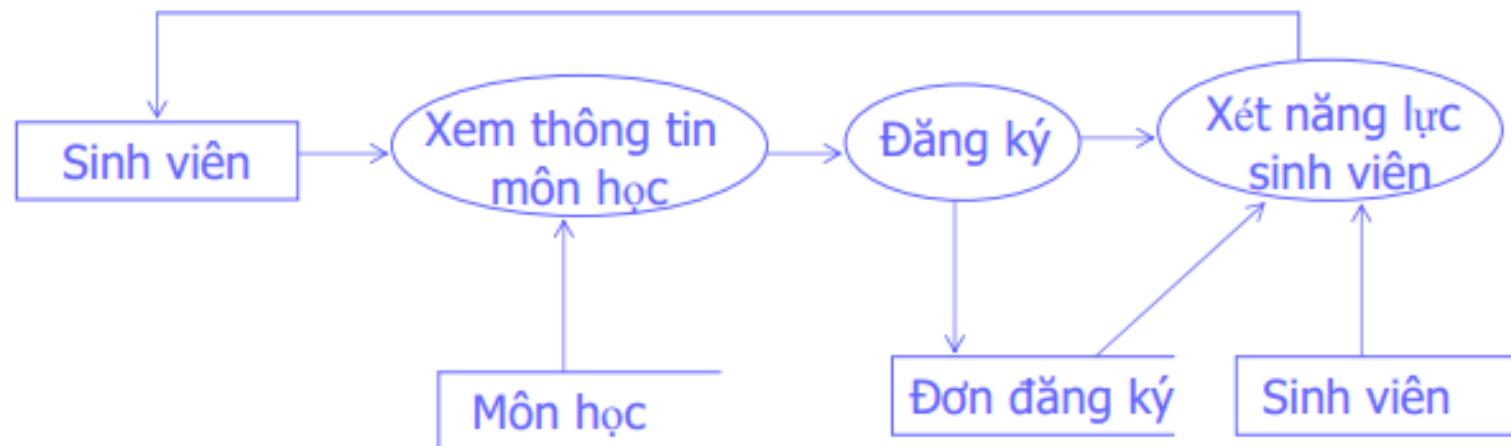
- Kết quả của việc xây dựng mô hình Use-case hiện thực hóa



### 3.2.1 Hiện thực hóa = Activity Diagram



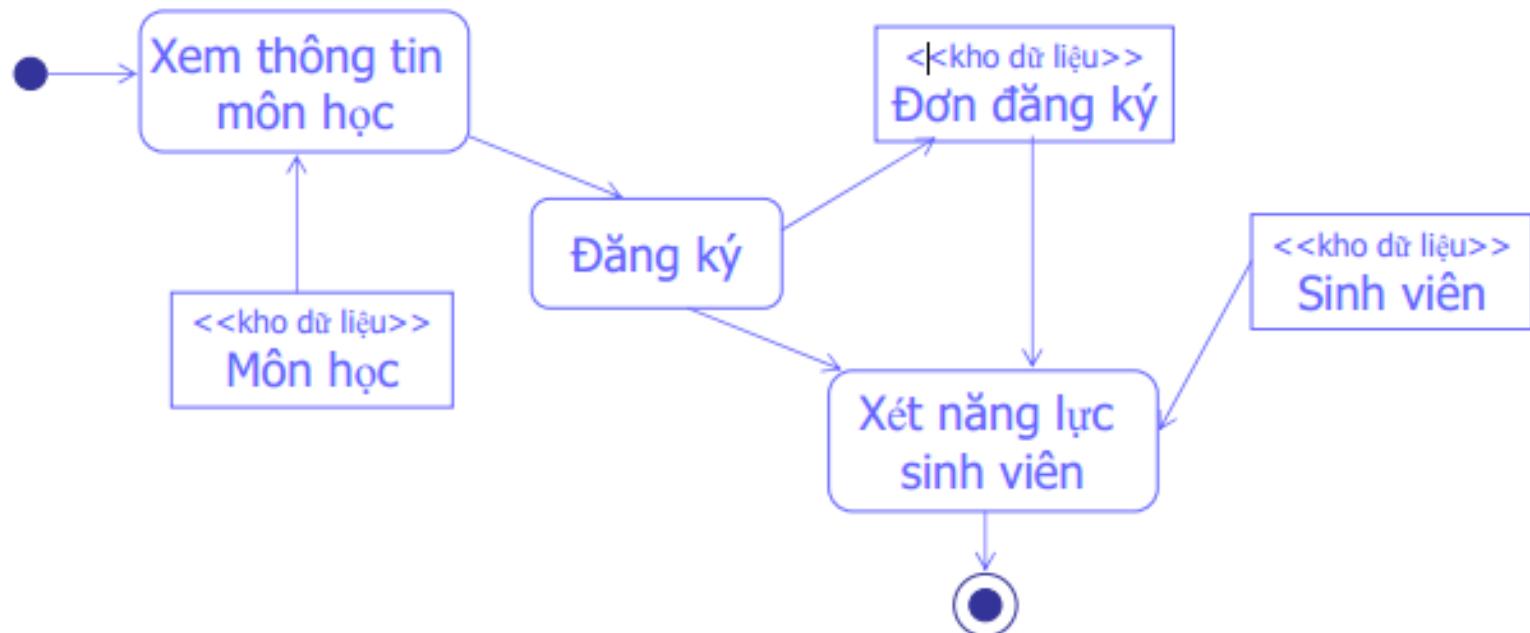
- Phương pháp phân tích thiết kế truyền thống
  - ◆ Sơ đồ dòng dữ liệu (DFD – Data Flow Diagram)
  - ◆ Ví dụ:



## 3.2.1 Hiện thực hóa = Activity Diagram



- Phương pháp phân tích thiết kế HĐT
  - ◆ Sơ đồ hoạt động
  - ◆ Ví dụ:

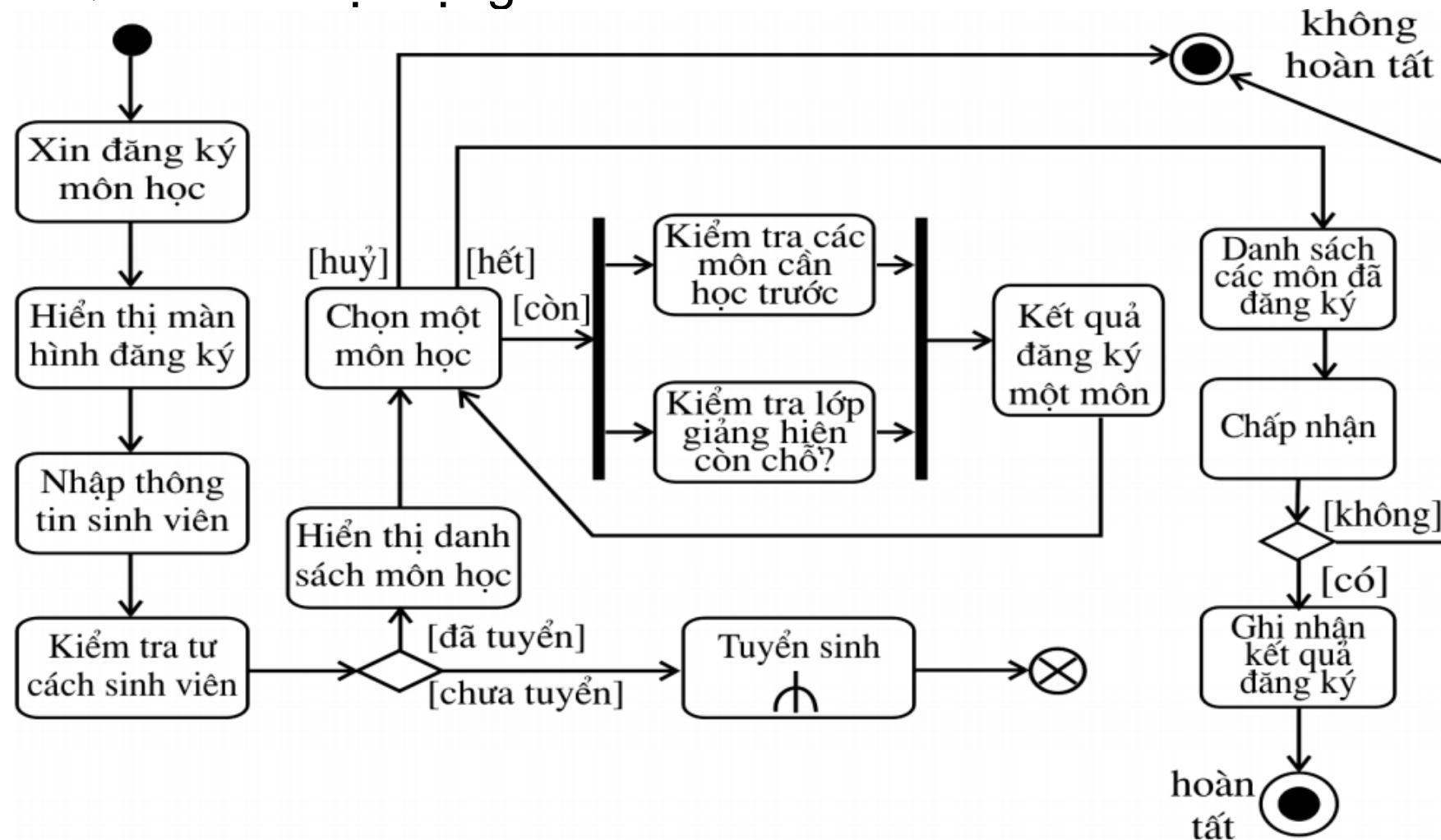


### 3.2.1 Hiện thực hóa = Activity Diagram



#### ■ Phương pháp phân tích thiết kế HDT

##### ◆ Sơ đồ hoạt động



### 3.2.1 Hiện thực hóa = Activity Diagram

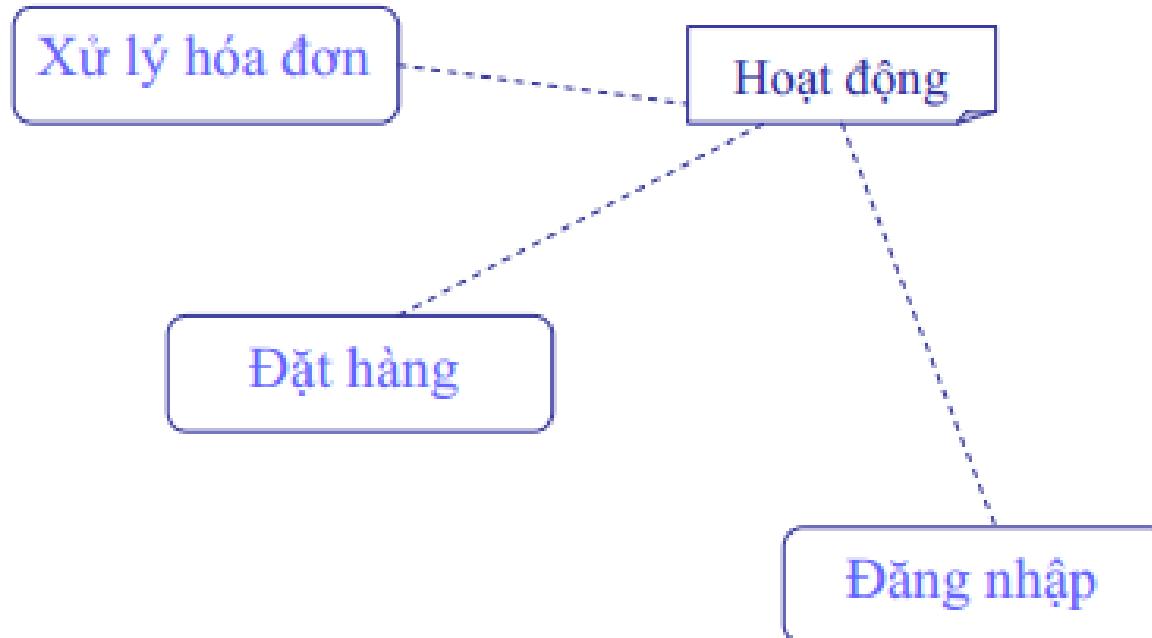


- Sơ đồ hoạt động: cho phép mô tả hoạt động của hệ thống
- Một sơ đồ hoạt động định nghĩa:
  - ◆ Các hoạt động (activity) của hệ thống và của các tác nhân
  - ◆ Thứ tự mà các hoạt động này được thực hiện
  - ◆ Phụ thuộc có thể giữa các hoạt động này
- Một hoạt động tương ứng với một công việc ở mức trừu tượng cao có mục đích xác định

## 3.2.1 Hiện thực hóa = Activity Diagram



- Ví dụ: các hoạt động



### 3.2.1 Hiện thực hóa = Activity Diagram



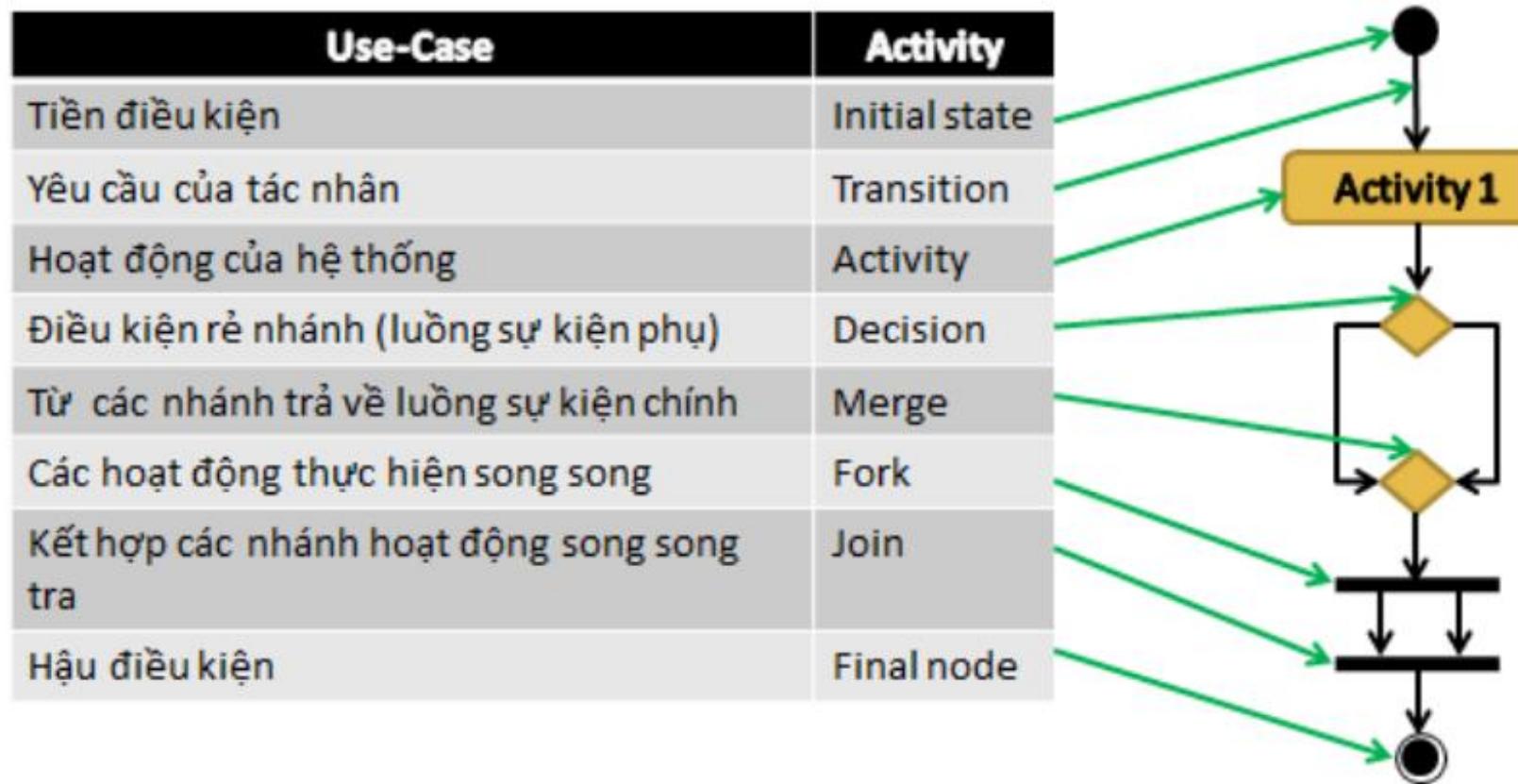
#### ■ Bộ ký hiệu Activity Diagram

Kí hiệu	Diễn giải
	Hoạt động (activity)
	Dòng điều khiển (control flow)
	Dòng điều khiển thông tin (object flow)
	Thanh đồng bộ hóa (synchronisation bar – fork / join node)
	Trạng thái kết thúc ( initial node)
	Trạng thái bắt đầu (final node)
	Rẽ nhánh – hợp nhánh (decision / merge node)
[condition]	Điều kiện rẽ nhánh



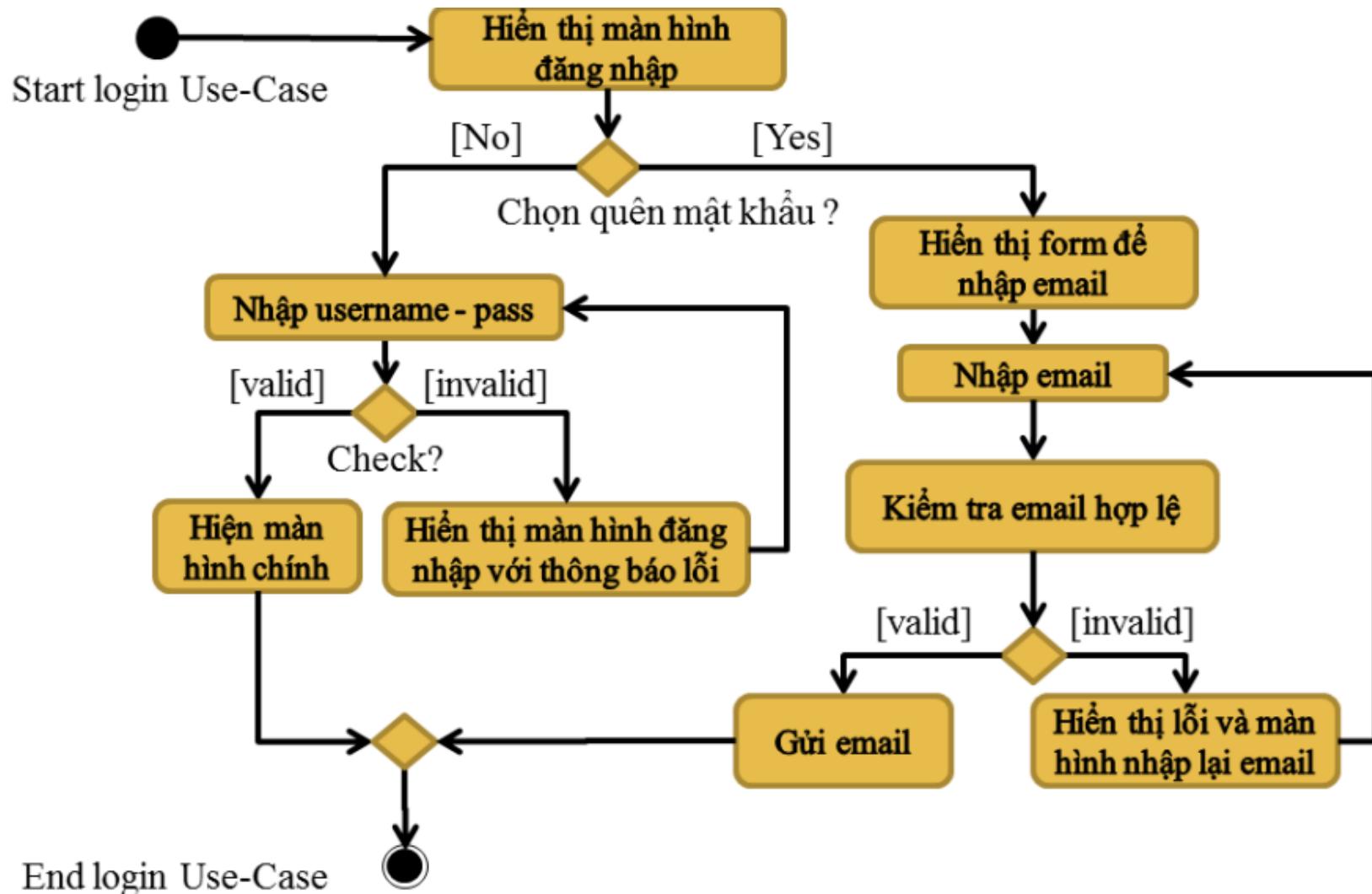
## 3.2.1 Hiện thực hóa = Activity Diagram

### Activity Diagram



### 3.2.1 Hiện thực hóa = Activity Diagram

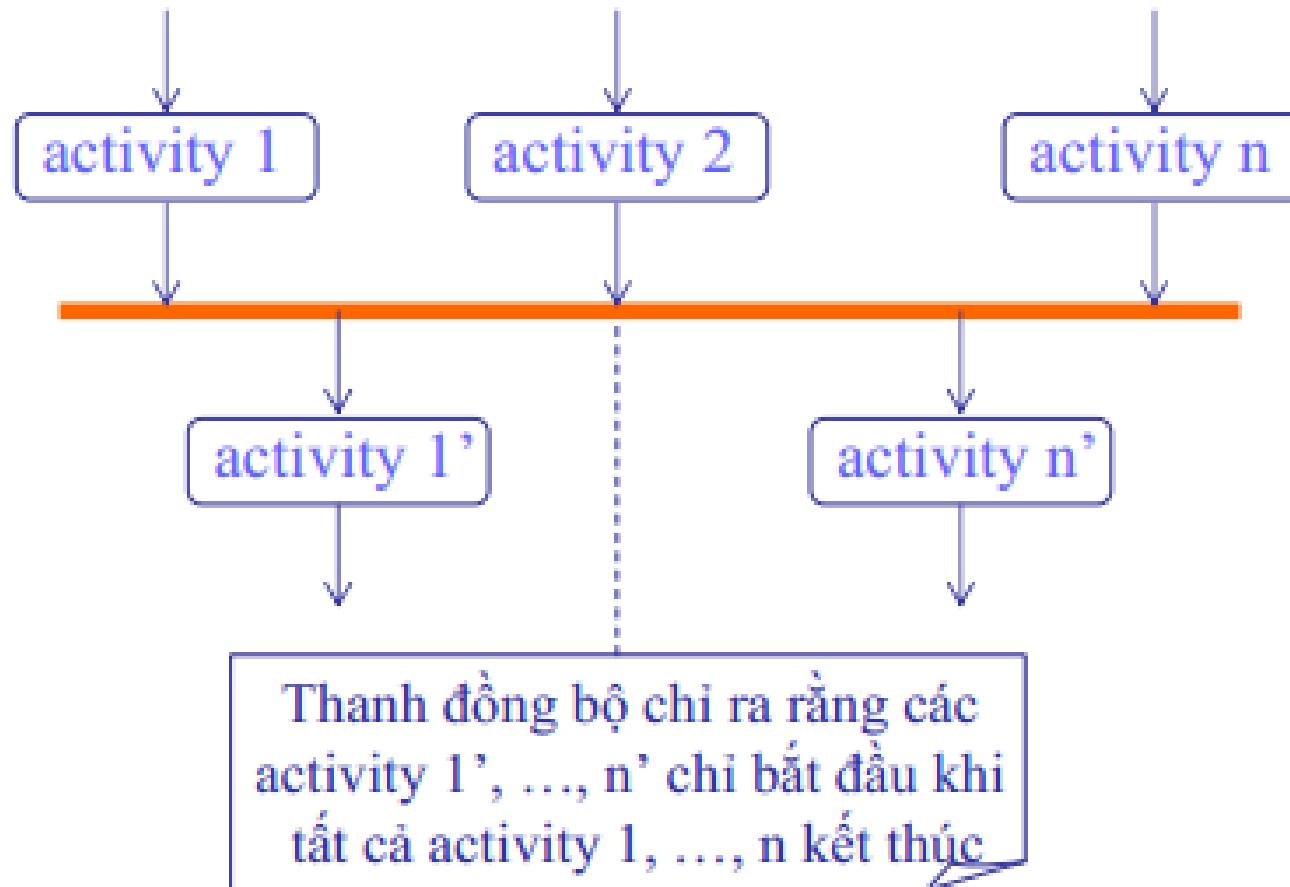
- VD: mô tả Use-case Đăng nhập bằng Activity Diagram



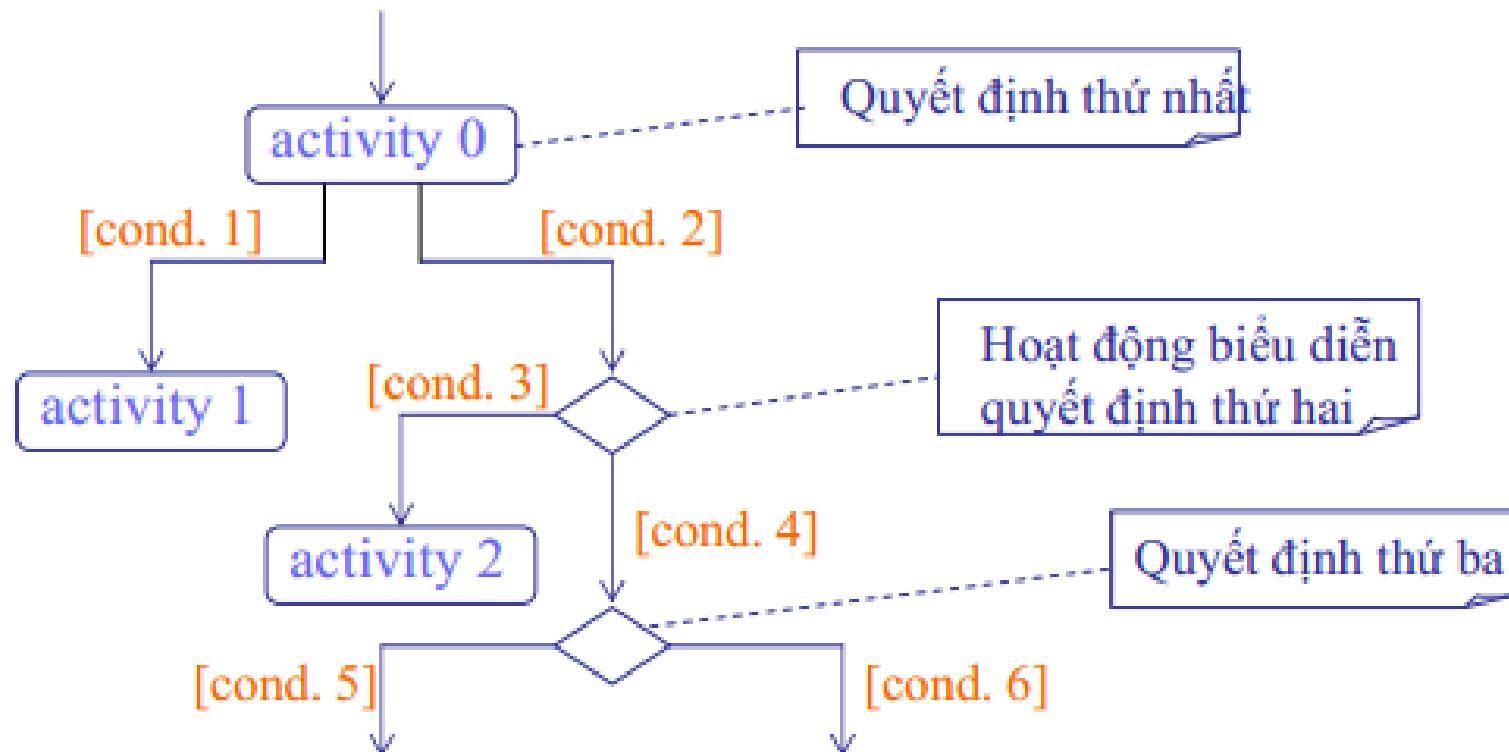
### 3.2.1 Hiệu thực hóa = Activity Diagram



#### ■ Đồng bộ hóa các hoạt động

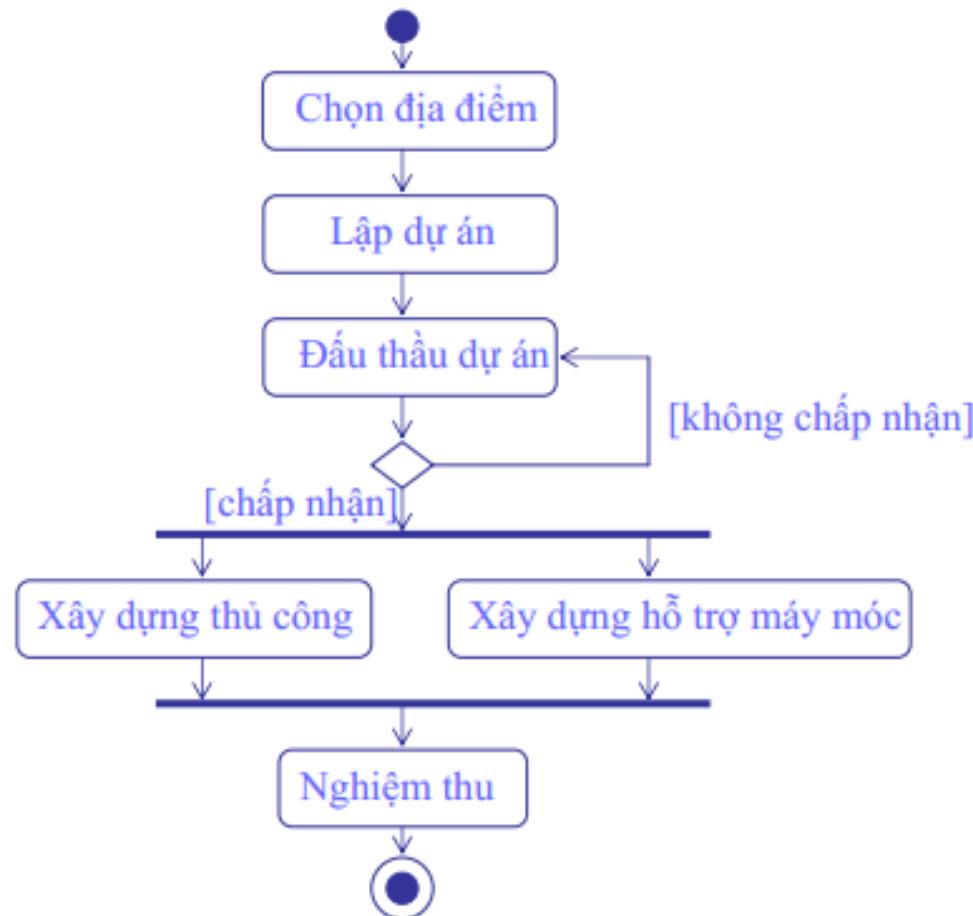


### **3.2.1 Hiện thực hóa = Activity Diagram**



## 3.2.1 Hiệu thực hóa = Activity Diagram

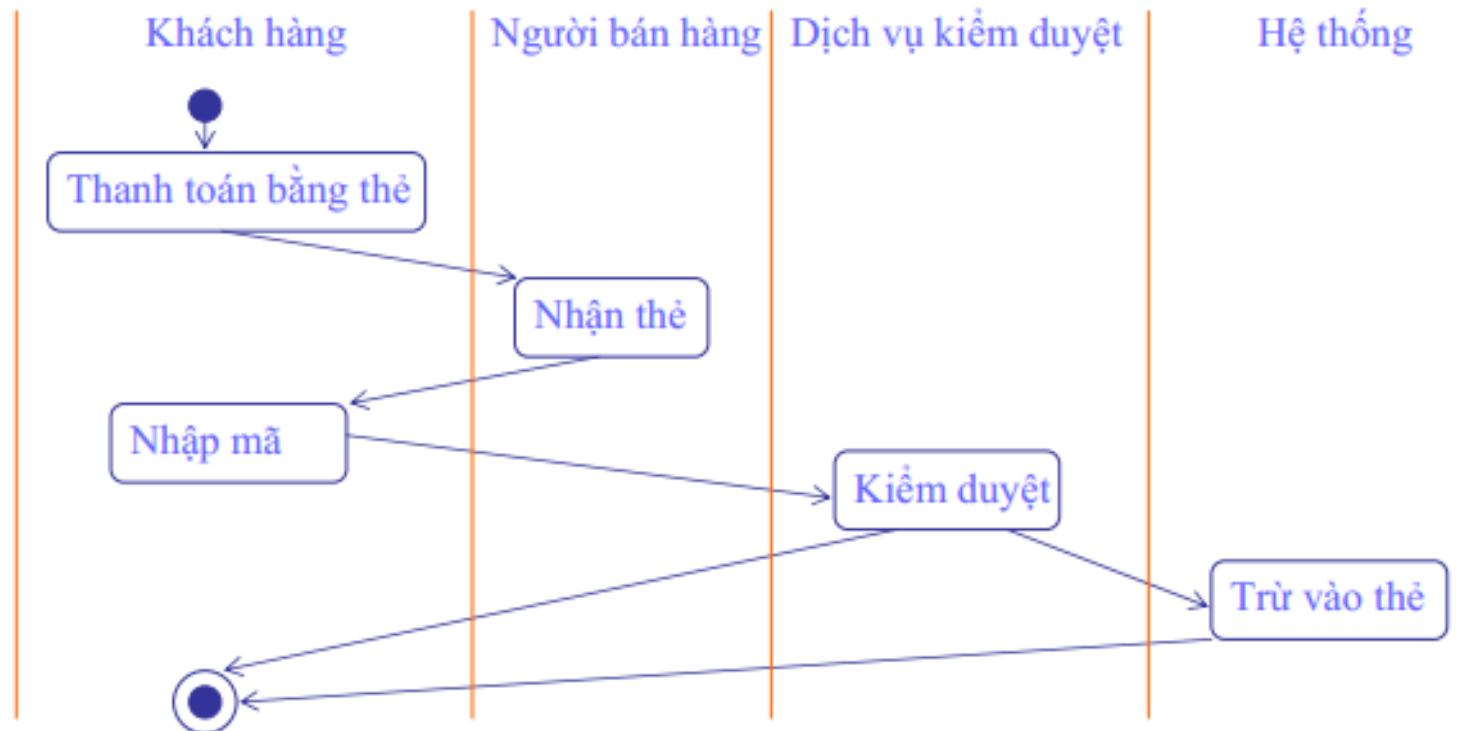
### ■ Ví dụ:



### 3.2.1 Hiện thực hóa = Activity Diagram



- Swimlane: dùng để phân luồng các hoạt động liên quan đến các tác nhân hoặc hệ thống



### 3.2.2 Hiện thực hóa = Sequence Diagram



- Biểu diễn sự tương tác giữa các đối tượng bằng việc nhấn mạnh thứ tự trao đổi thông điệp giữa các đối tượng.
- Biểu đồ tuần tự gồm:
  - ◆ Tác nhân (actor)
    - Là con người hoặc hệ thống bên ngoài
    - Tham gia vào quy trình với chức năng gửi/ nhận thông điệp



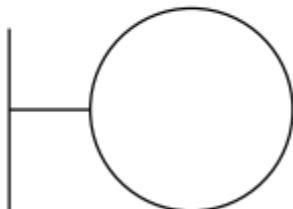
## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Biểu đồ tuần tự gồm(tt):

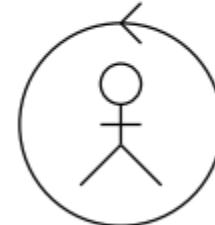
#### ◆ Đối tượng (object)

- Tham gia vào qui trình với các chức năng gửi/ nhận thông điệp.

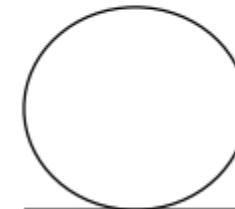
<<đối tượng: tên lớp>>



Thừa tác viên làm việc  
và giao tiếp bên ngoài



Thừa tác viên làm việc  
bên trong



Thực thể nghiệp vụ



## 3.2.2 Hiện thực hóa = Sequence Diagram



### ■ Biểu đồ tuần tự gồm(tt):

#### ◆ Đường sống của đối tượng (Lifeline)

- Thể hiện vòng đời của đối tượng trong suốt quá trình tương tác
- Nếu trên đường sống của đối tượng xuất hiện điểm dừng X, thì đối tượng không còn tương tác với đối tượng khác.



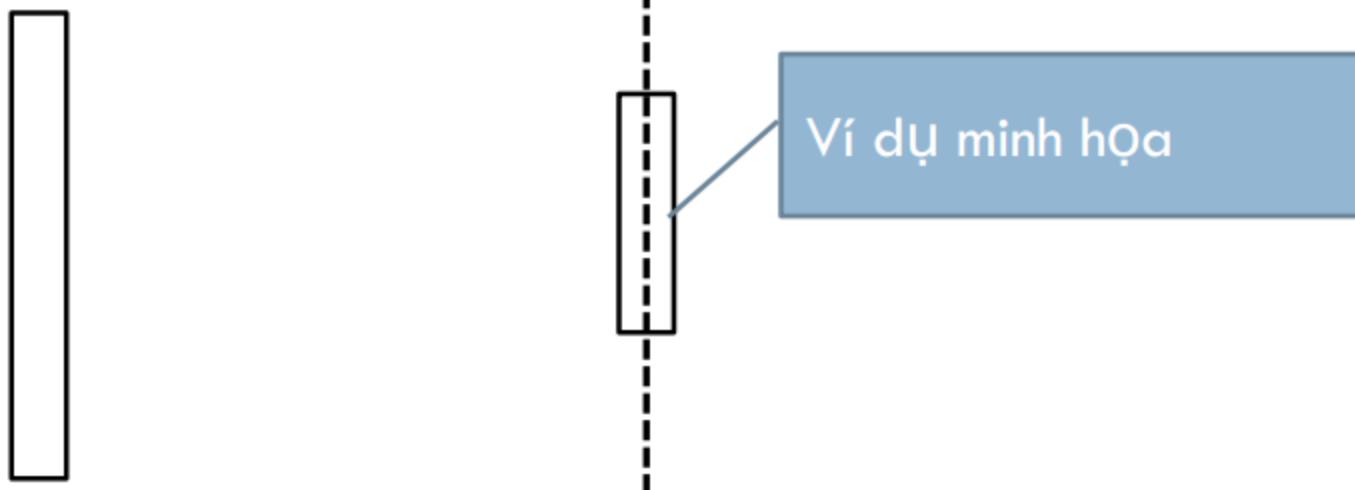
## 3.2.2 Hiện thực hóa = Sequence Diagram



### ■ Biểu đồ tuần tự gồm(tt):

#### ◆ Kích hoạt thực thi (execution occurrence)

- Thể hiện khi một đối tượng gửi/ nhận một thông điệp.
- Được đặt dọc theo đường sống của đối tượng

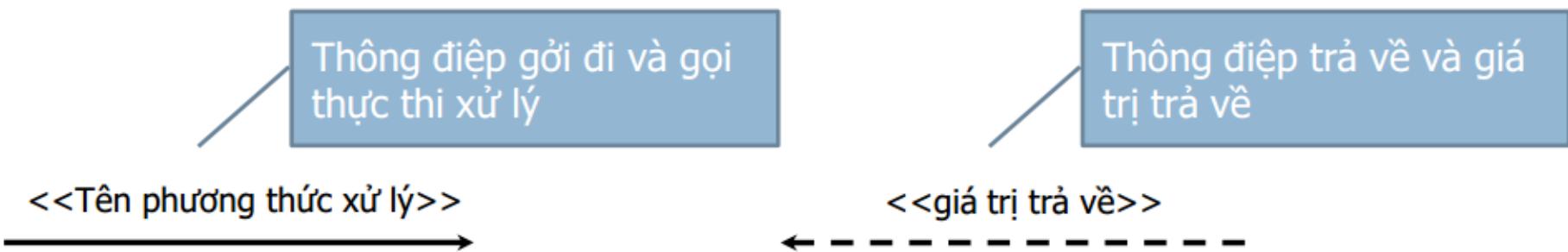


## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Biểu đồ tuần tự gồm(tt):

#### ◆ Thông điệp (messages)

- Thể hiện thông tin đi từ một đối tượng này đến đối tượng khác



## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Biểu đồ tuần tự gồm(tt):

#### ◆ Thông điệp có điều kiện (guard condition)

- Thể hiện thông tin đi từ một đối tượng này đến đối tượng khác

Thông điệp gửi đi và gọi thực thi  
xử lý có điều kiện

<<[điều kiện:] Tên phương thức xử lý>>



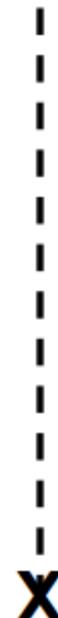
## 3.2.2 Hiện thực hóa = Sequence Diagram



### ■ Biểu đồ tuần tự gồm(tt):

#### ◆ Kết thúc hoạt động của đối tượng (object destruction)

- Thể hiện kết thúc chu kỳ sống của một đối tượng, nghĩa là đối tượng không còn tham gia vào quy trình tương tác.



## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Sequence Diagram

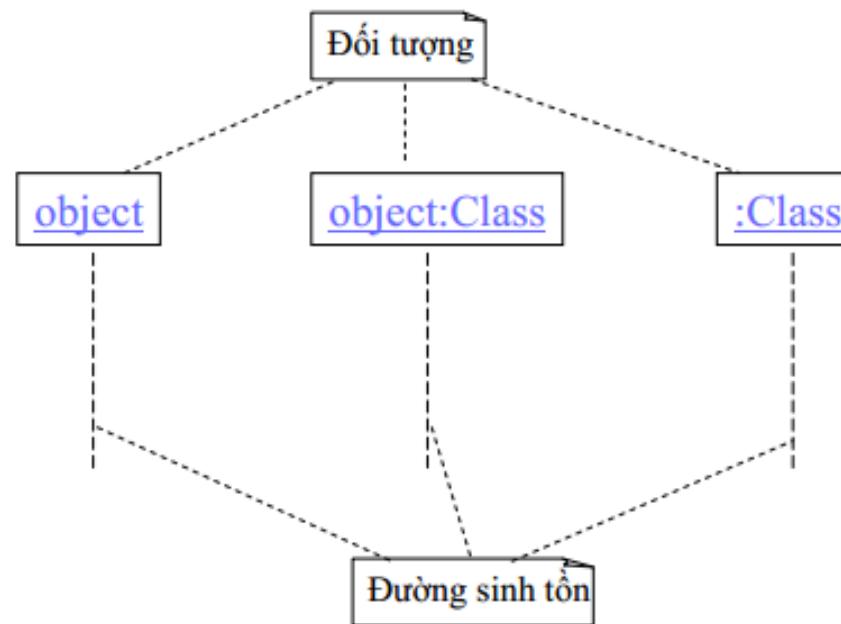
Thành phần	Ký hiệu	Ý nghĩa
Actor		Tác nhân bên ngoài hệ thống
Boundary		Đối tượng giao tiếp giữa tác nhân và hệ thống
Entity		Đối tượng bên trong hệ thống thể hiện các đối tượng thực thể
Controller		Đối tượng bên trong hệ thống cung cấp dịch vụ để xử lý các quy trình
Message		Thông điệp trao đổi giữa các đối tượng, bao gồm:  1. gọi điện (sdt)  <ul style="list-style-type: none"><li>STT: thứ tự trao đổi của thông điệp</li><li>Tên thông điệp</li><li>Các tham số trao đổi</li><li>Mũi tên chỉ chiều trao đổi thông điệp</li></ul>



## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Sơ đồ tuần tự:

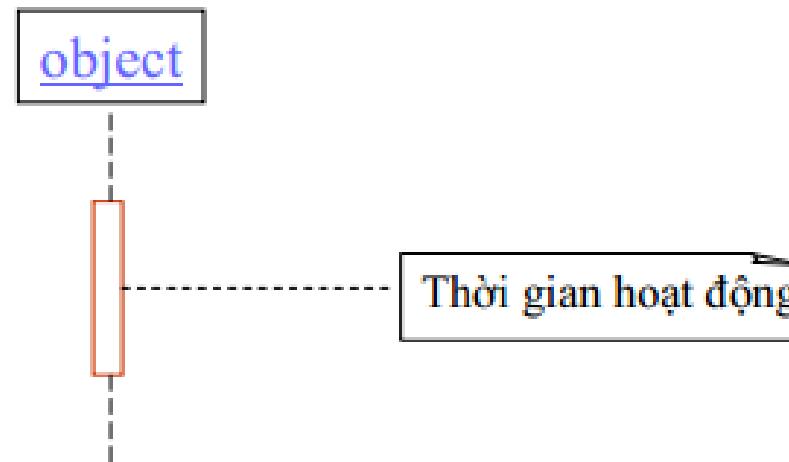
- ◆ Mỗi đối tượng có một đường sinh tồn (lifeline) biểu diễn thời gian tồn tại của nó.



## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Sơ đồ tuần tự:

- ◆ Thời gian hoạt động (activation) là thời gian mà đối tượng đang thực hiện một thao tác



## 3.2.2 Hiện thực hóa = Sequence Diagram

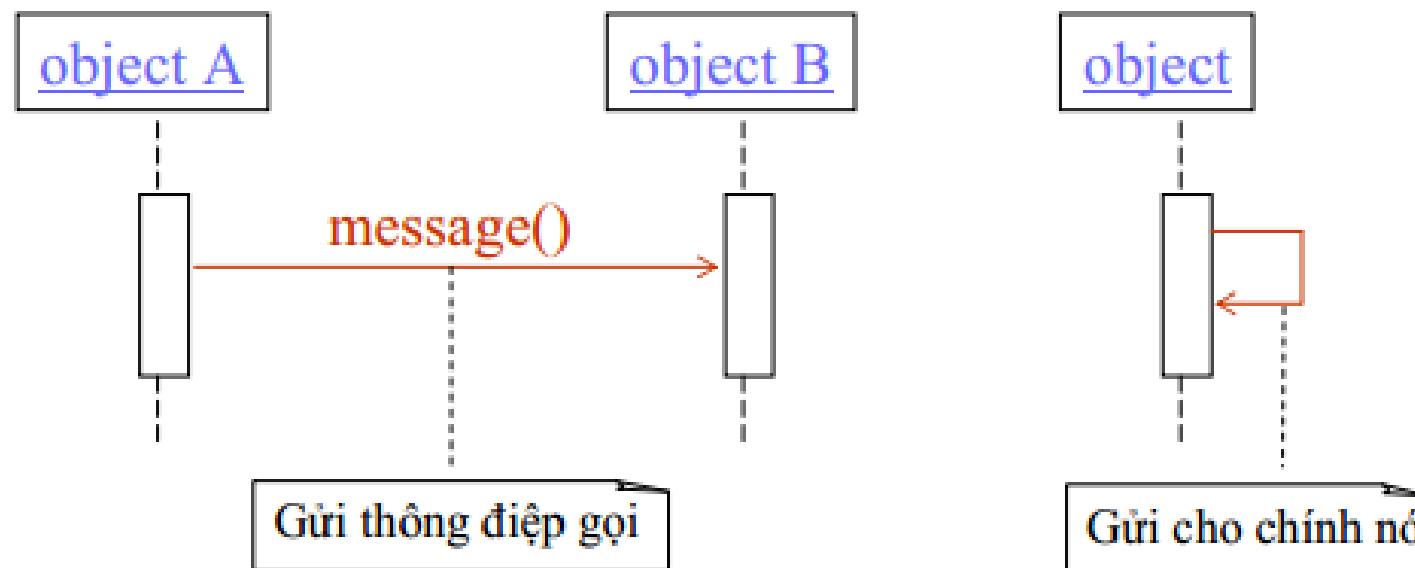
### ■ Sơ đồ tuần tự:

- ◆ Một thông điệp đặc tả trao đổi giữa các đối tượng
- ◆ Các loại thông điệp:
  - Gọi (call)
  - Trả lời (return)
  - Gửi (send)
  - Tạo (create)
  - Hủy (destroy)

## 3.2.2 Hiện thực hóa = Sequence Diagram

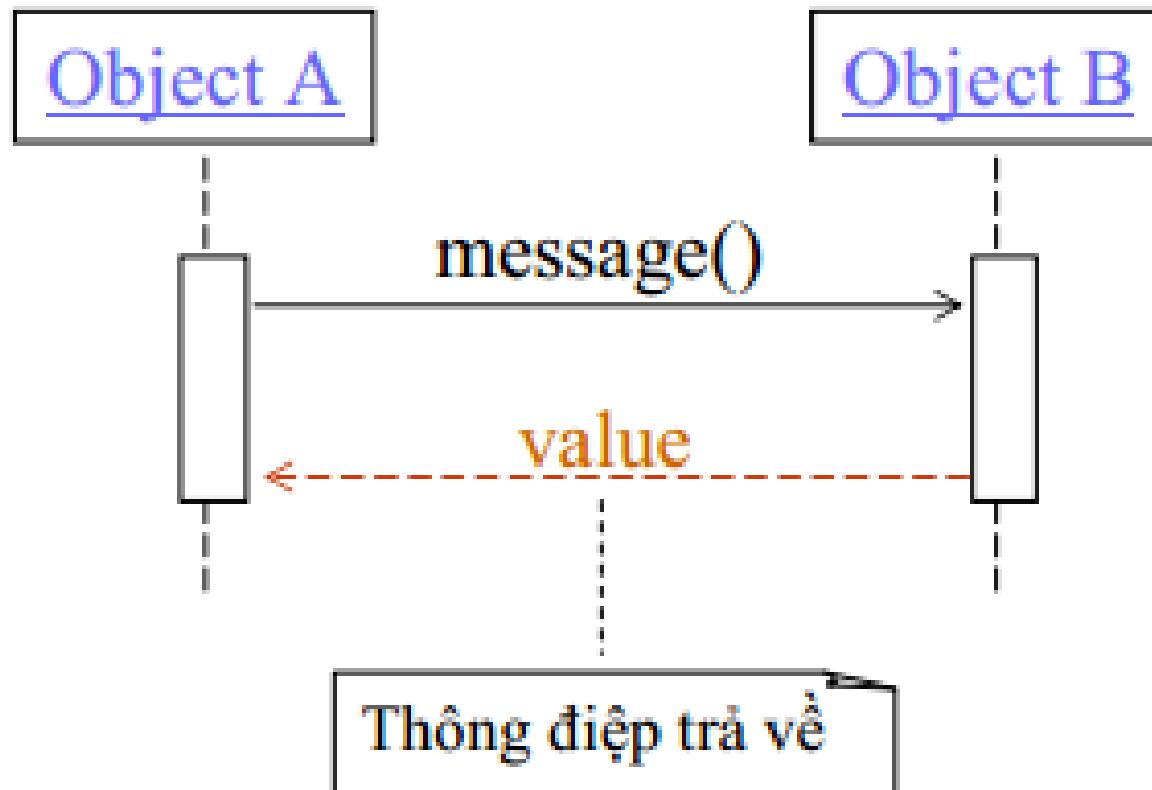
### ■ Sơ đồ tuần tự:

- ◆ Thông điệp gọi (message): gọi một phương thức/thao tác trên đối tượng
  - Đối tượng gọi phải chờ thông điệp được thực hiện kết thúc mới có thể thực hiện công việc khác (thông điệp đồng bộ)
- ◆ Một đối tượng có thể gửi thông điệp cho chính nó (self-message)



## 3.2.2 Hiện thực hóa = Sequence Diagram

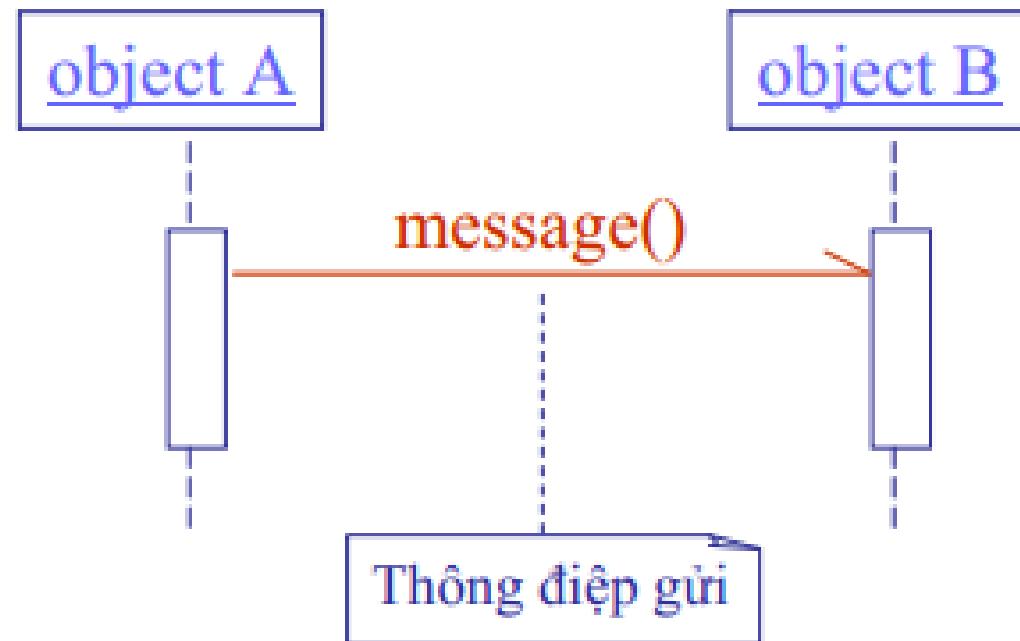
- Sơ đồ tuần tự:
  - ◆ Thông điệp trả về (return): trả về một giá trị cho đối tượng gọi



## 3.2.2 Hiện thực hóa = Sequence Diagram

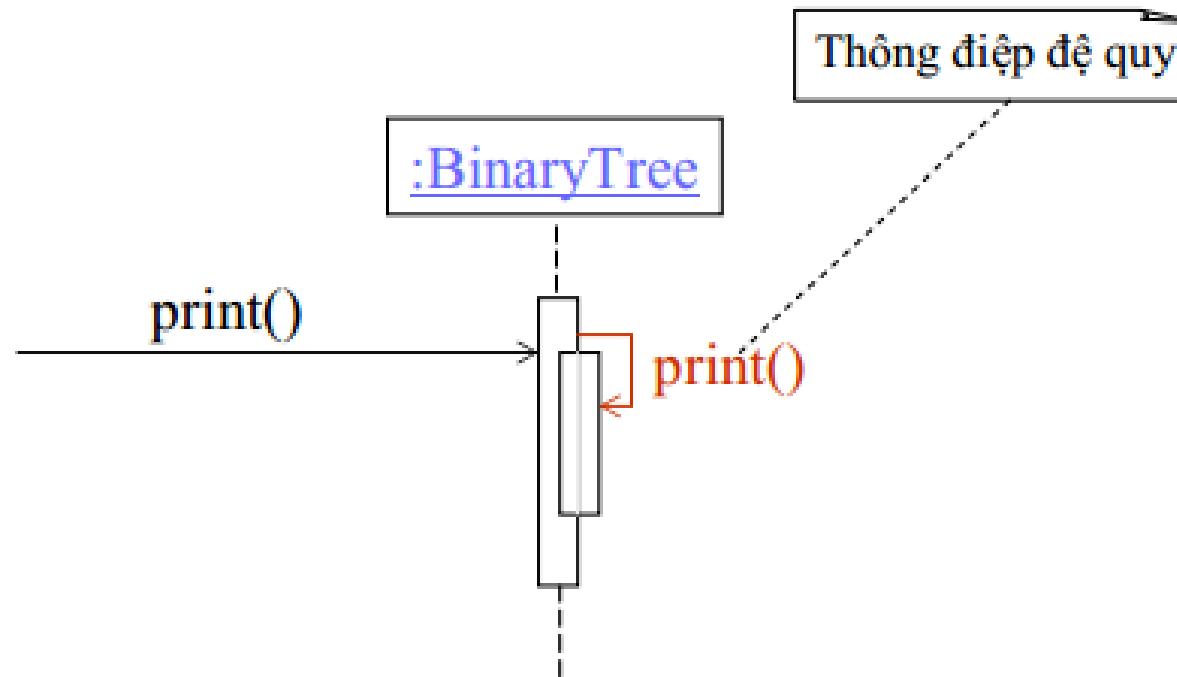
### ■ Sơ đồ tuần tự:

- ◆ Thông điệp gửi: gửi một tín hiệu đến một đối tượng
  - Khác với thông điệp gọi, khi đối tượng gửi thông điệp gửi, nó không chờ đợi, mà tiếp tục thực hiện công việc khác (thông điệp không đồng bộ)
- ◆ Ký hiệu:



## 3.2.2 Hiện thực hóa = Sequence Diagram

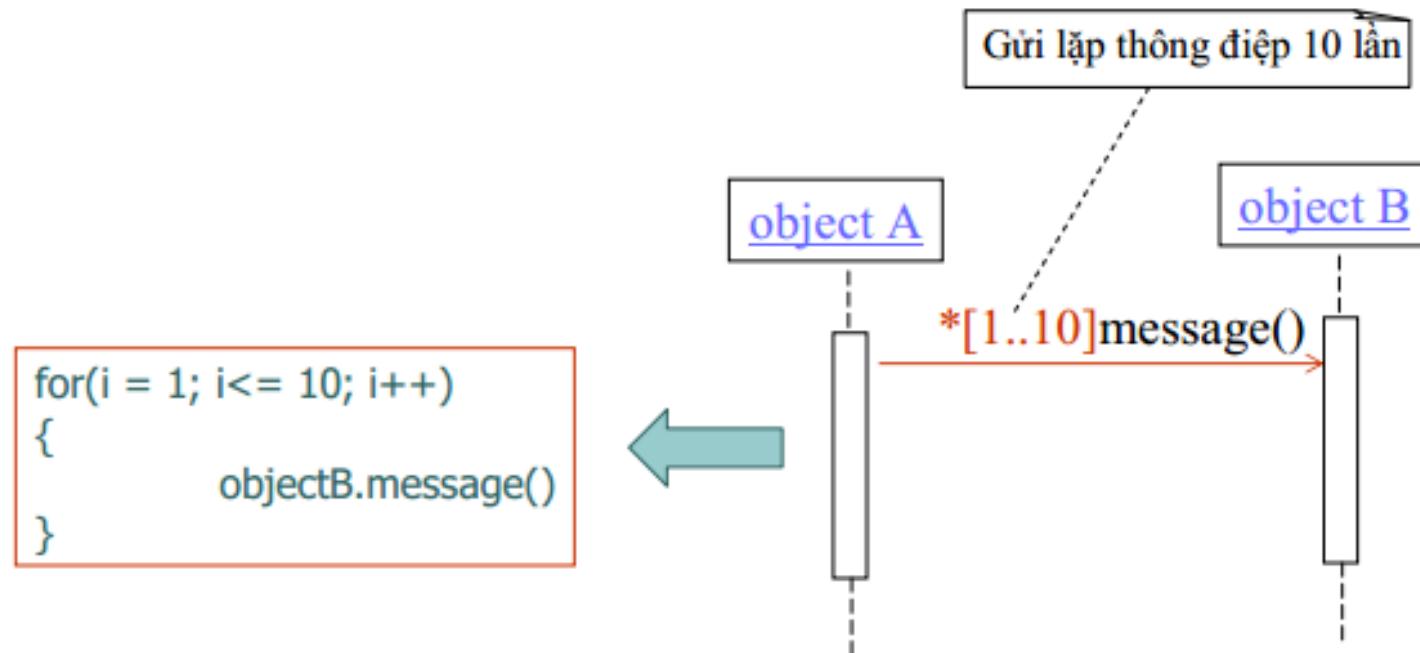
- Sơ đồ tuần tự:
  - ◆ Thông điệp gọi đệ quy (Recursion):



## 3.2.2 Hiện thực hóa = Sequence Diagram

### ■ Sơ đồ tuần tự:

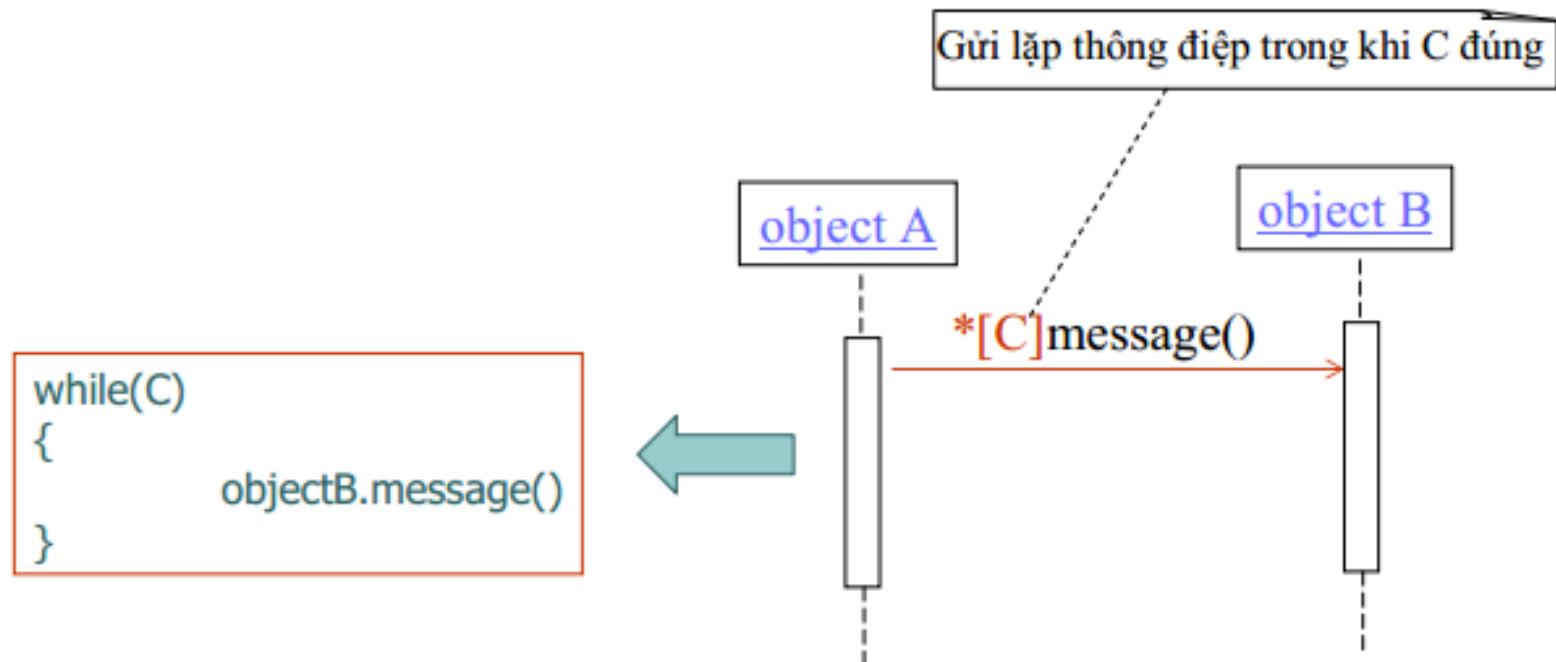
- ◆ Một thông điệp có thể gửi lặp lại nhiều lần:



## 3.2.2 Hiện thực hóa = Sequence Diagram

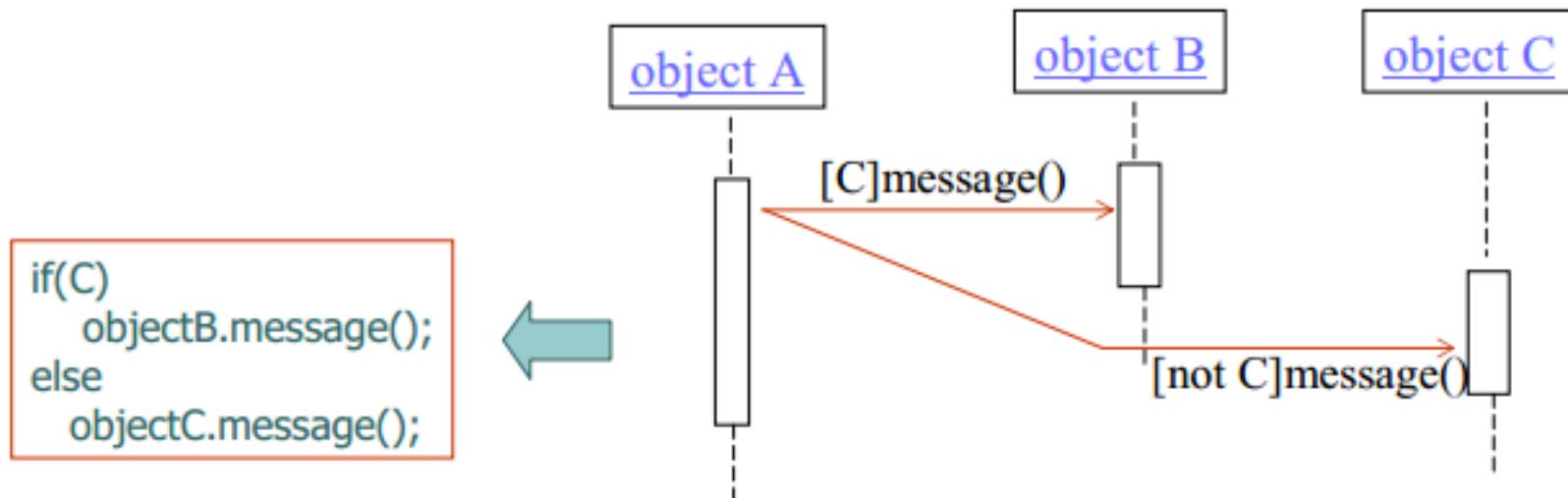
### Sơ đồ tuần tự:

- Một thông điệp có thể gửi lặp lại nhiều lần phụ thuộc vào một điều kiện:



## 3.2.2 Hiện thực hóa = Sequence Diagram

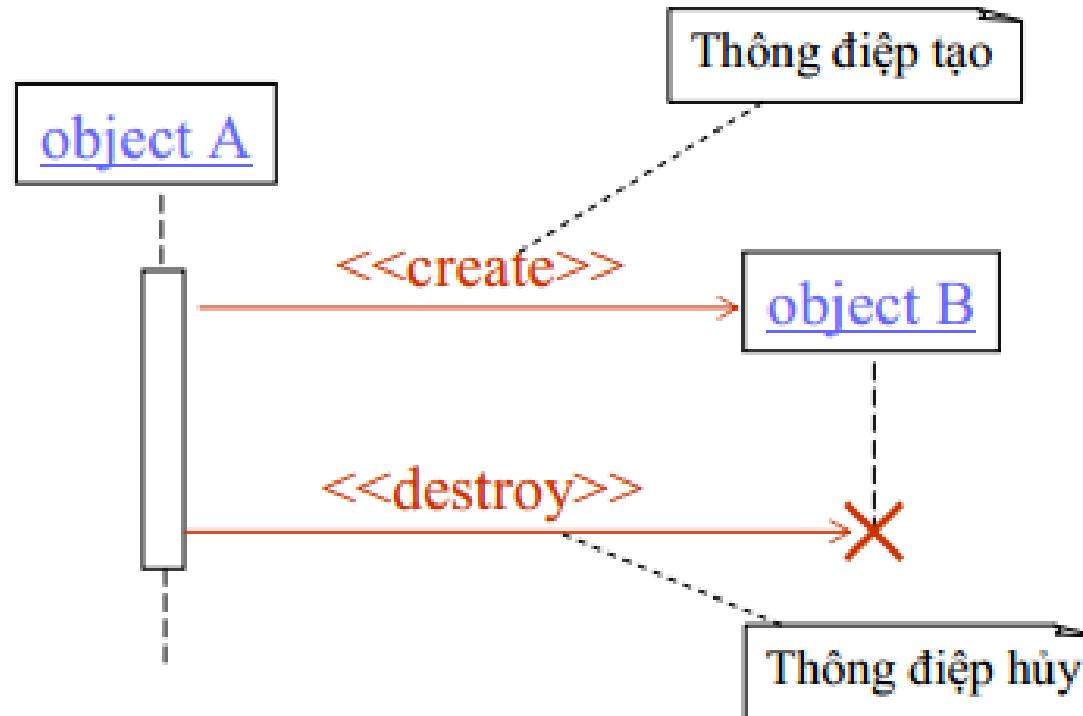
- Sơ đồ tuần tự:
  - ◆ Một thông điệp có thể được gửi phụ thuộc vào điều kiện rẽ nhánh:



## 3.2.2 Hiện thực hóa = Sequence Diagram

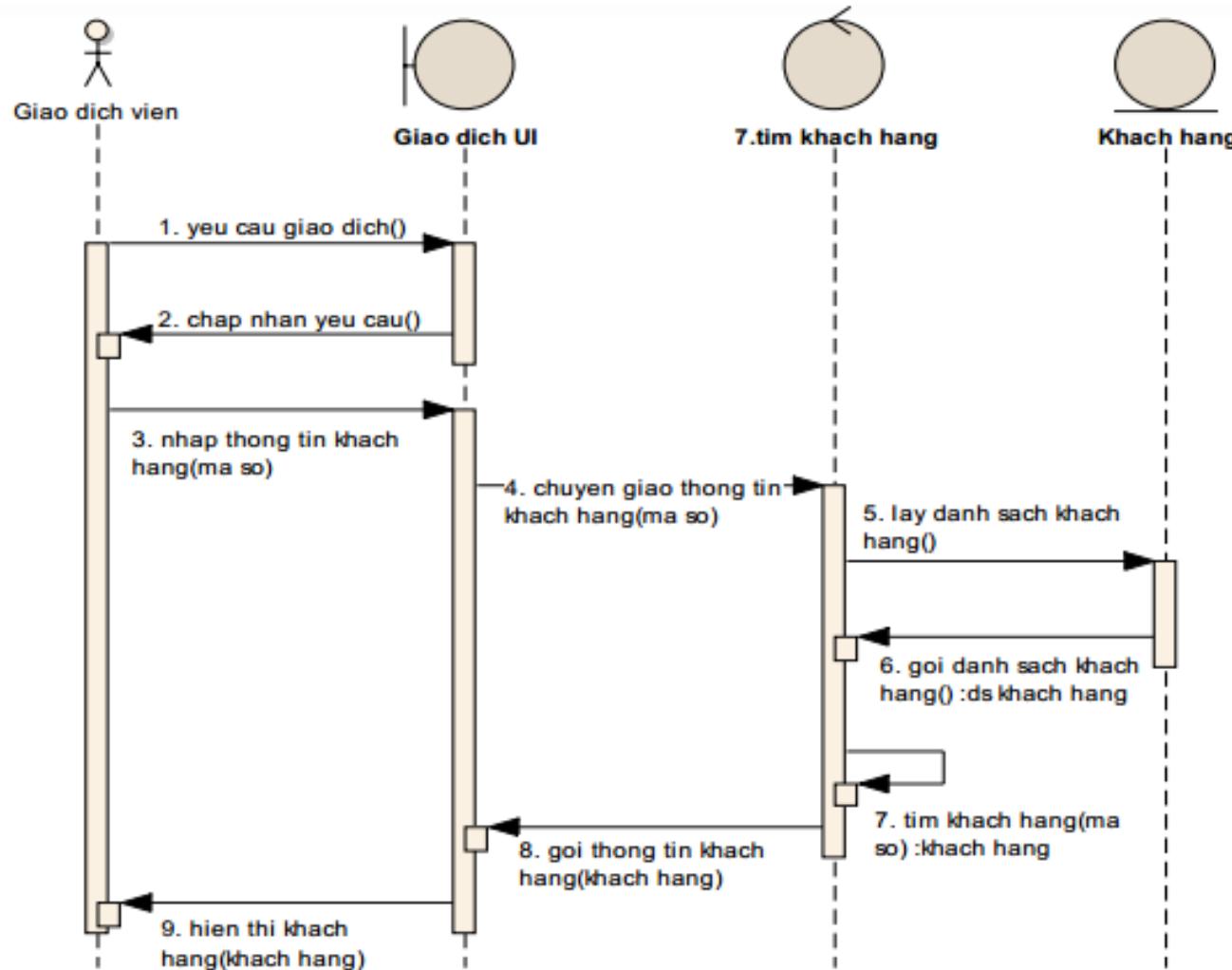
### ■ Sơ đồ tuần tự:

- ◆ Thông điệp tạo mới gọi phương thức tạo một đối tượng.
- ◆ Thông điệp hủy gọi phương thức hủy một đối tượng



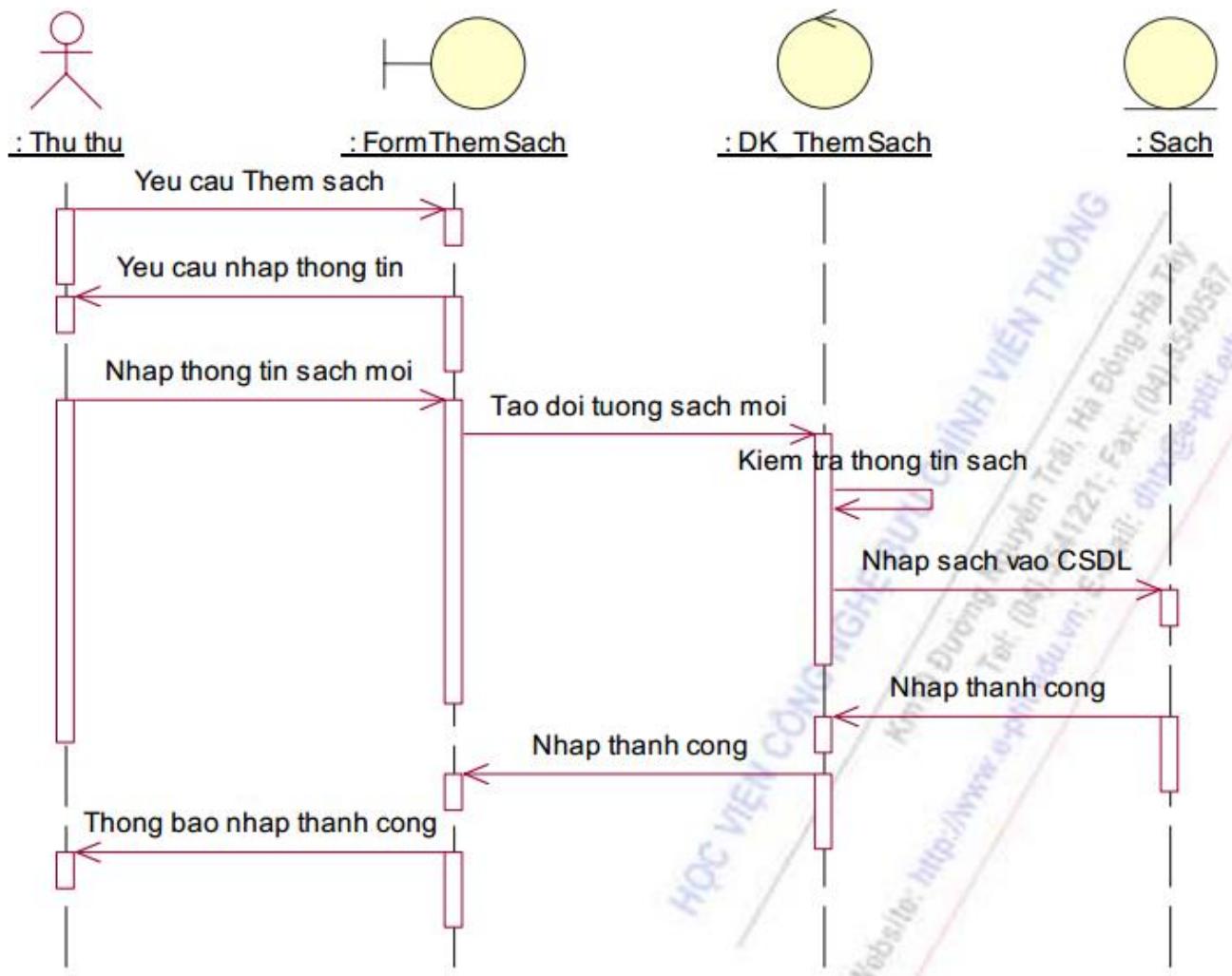
## 3.2.2 Hiện thực hóa = Sequence Diagram

- VD: mô tả use-case Tìm khách hàng



## 3.2.2 Hiện thực hóa = Sequence Diagram

- VD: mô tả use-case Thêm sách



## 3.2.2 Hiện thực hóa = Sequence Diagram

# BÀI TẬP



### 3.2.3 Hiện thực hóa = Collaboration Diagram

#### ■ Sơ đồ cộng tác:

- ◆ Biểu diễn tương tác giữa các đối tượng, trong đó nhấn mạnh sự tham gia của các đối tượng thông qua những mối liên kết giữa chúng và các thông điệp (message) chúng gửi cho nhau.
- ◆ Biểu đồ cộng tác chỉ ra:
  - Thứ tự gửi các thông điệp: mỗi thông điệp được gán một số tuần tự
  - Điều kiện gửi các thông điệp.

### 3.2.3 Hiện thực hóa = Collaboration Diagram

#### ■ Sơ đồ cộng tác:

- ◆ Gồm các thành phần:
  - Tác nhân (actor)
  - Đối tượng (object)
  - Sự kết hợp (association)
  - Thông điệp (message)
  - Thông điệp có điều kiện (guard condition)

### 3.2.3 Hiện thực hóa = Collaboration Diagram

- Gồm các thành phần:

- ◆ Sự kết hợp (association):

- Thể hiện sự kết hợp giữa tác nhân và đối tượng
      - Được sử dụng để gửi thông điệp
- 

- ◆ Thông điệp (message)/ thông điệp có điều kiện:

- Vận chuyển thông tin đi từ một đối tượng này đến đối tượng khác.

*<<Stt thông điệp: Tên thông điệp>>*



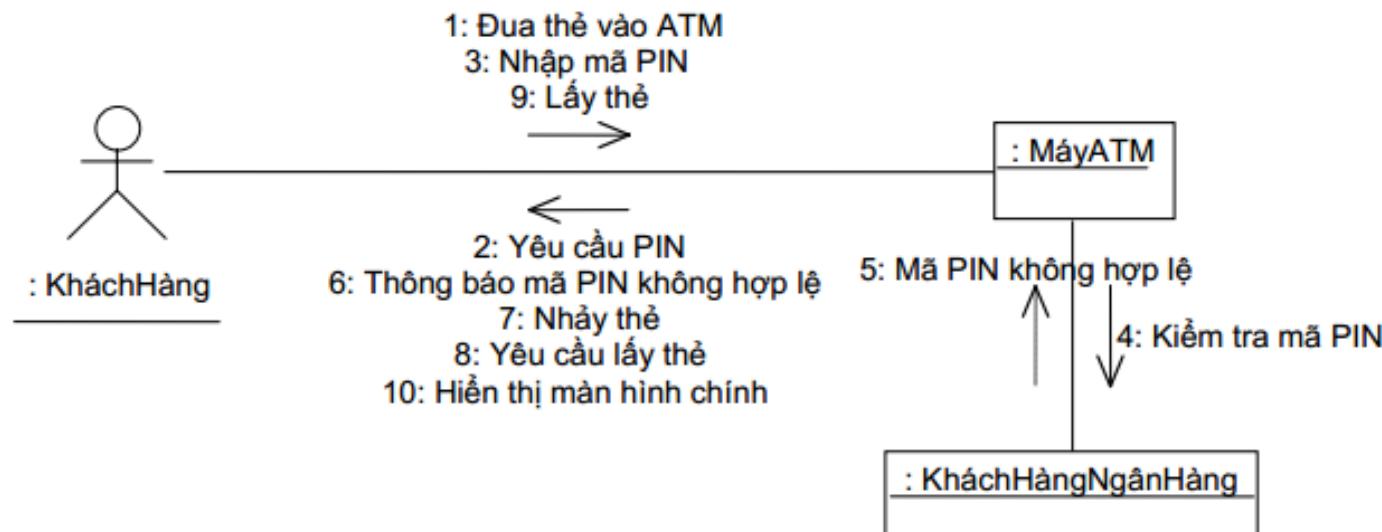
- Kiểm tra thông điệp phải thỏa để được gửi đi.

*<< Stt thông điệp:[điều kiện]: Tên thông điệp>>*



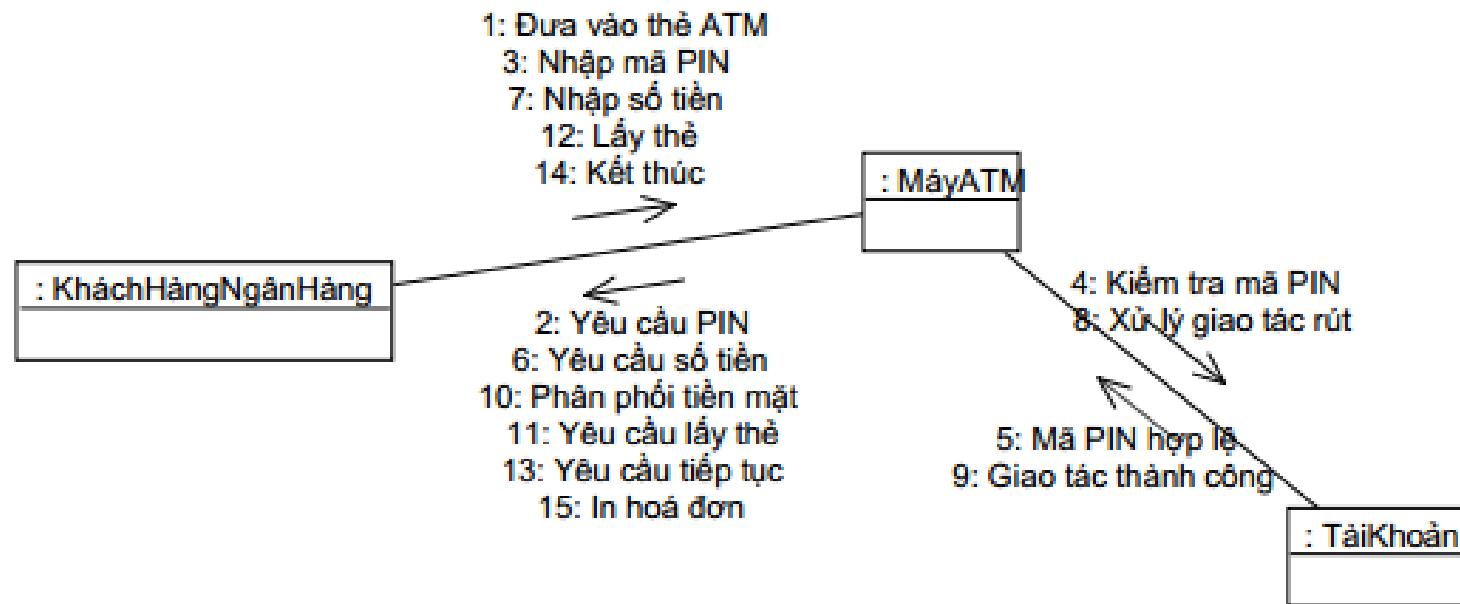
### 3.2.3 Hiện thực hóa = Collaboration Diagram

#### ■ Ví dụ usecase “Giải quyết PIN không hợp lệ”:



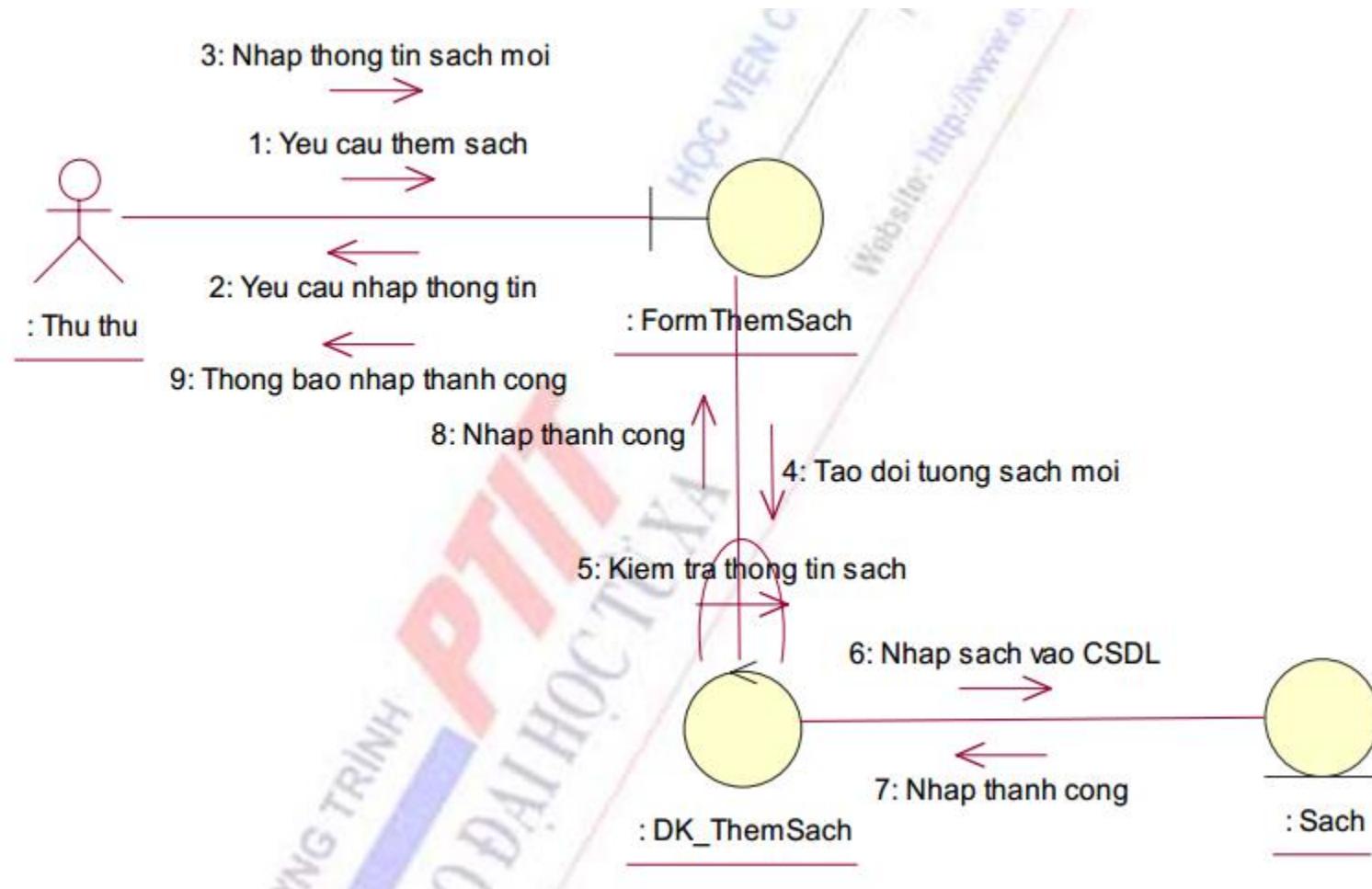
### 3.2.3 Hiện thực hóa = Collaboration Diagram

#### ■ Ví dụ usecase “Rút tiền”:



### 3.2.3 Hiện thực hóa = Collaboration Diagram

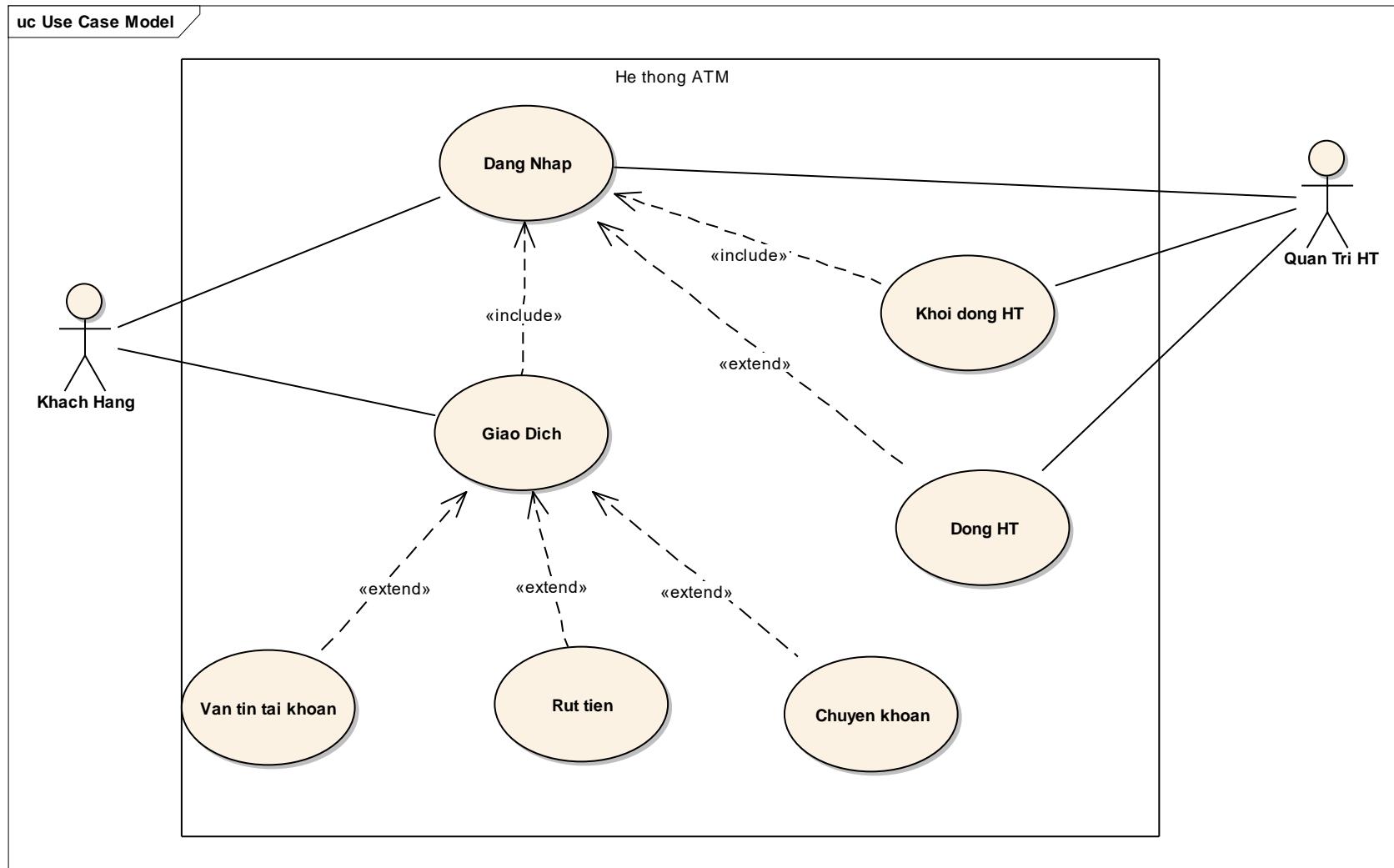
#### ■ Ví dụ usecase “Thêm sách”:



# VD: Hiện thực hóa usecase “Đăng nhập”



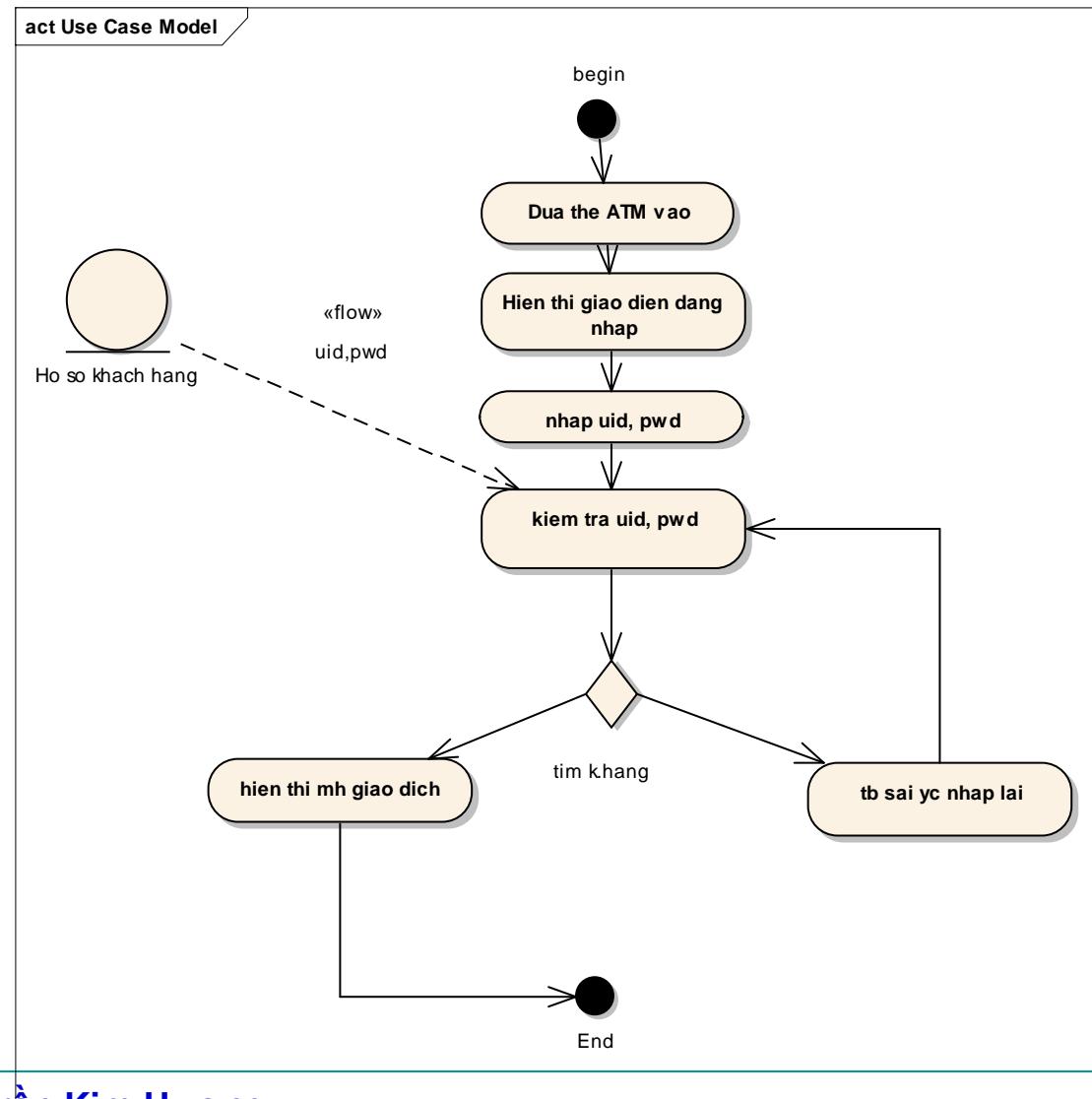
## ■ Ví dụ usecase “Đăng nhập”:



# VD: Hiện thực hóa usecase “Đăng nhập”



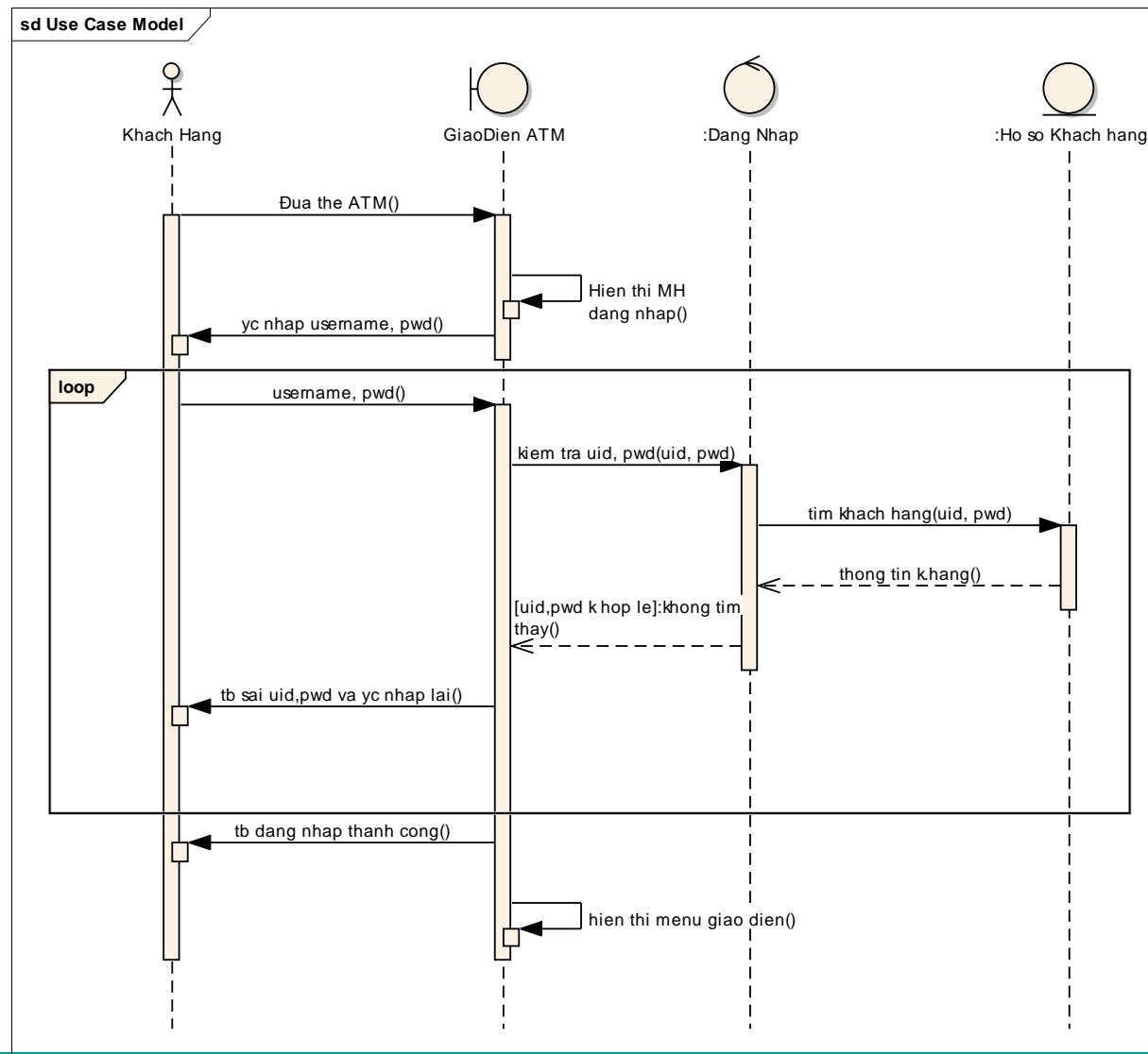
## ■ Ví dụ activity usecase “Đăng nhập”:



# VD: Hiện thực hóa usecase “Đăng nhập”



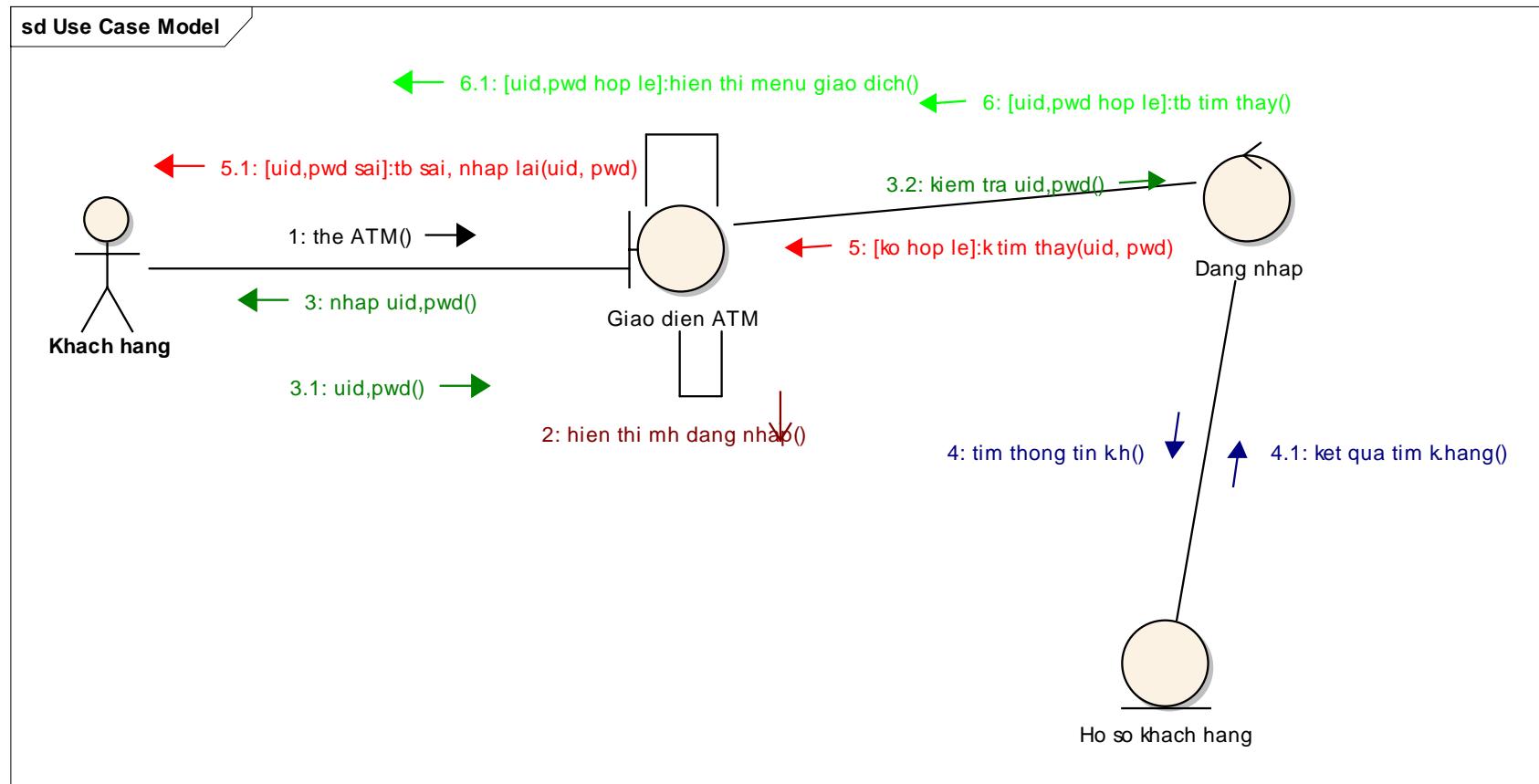
## ■ Ví dụ sequence của usecase “Đăng nhập”:



# VD: Hiện thực hóa usecase “Đăng nhập”



## ■ Ví dụ communication của usecase “Đăng nhập”:



## 3.2.3 Hiện thực hóa = Collaboration Diagram

# BÀI TẬP



# Hướng dẫn thực hành



- Sử dụng phần mềm Enterprise Architect
- Quản lý mượn sách thư viện
  - ◆ Vẽ Activity Diagram
  - ◆ Vẽ Sequence Diagram
  - ◆ Vẽ Communication Diagram

# Câu hỏi thảo luận

