

---

# GIỚI THIỆU UML

Lecturer: Nguyễn Đức Cường - FIT  
cuongnguyenduc@gmail.com

Website:

<http://www.nguyenduccuong.com>

Email:

# Mục tiêu

---

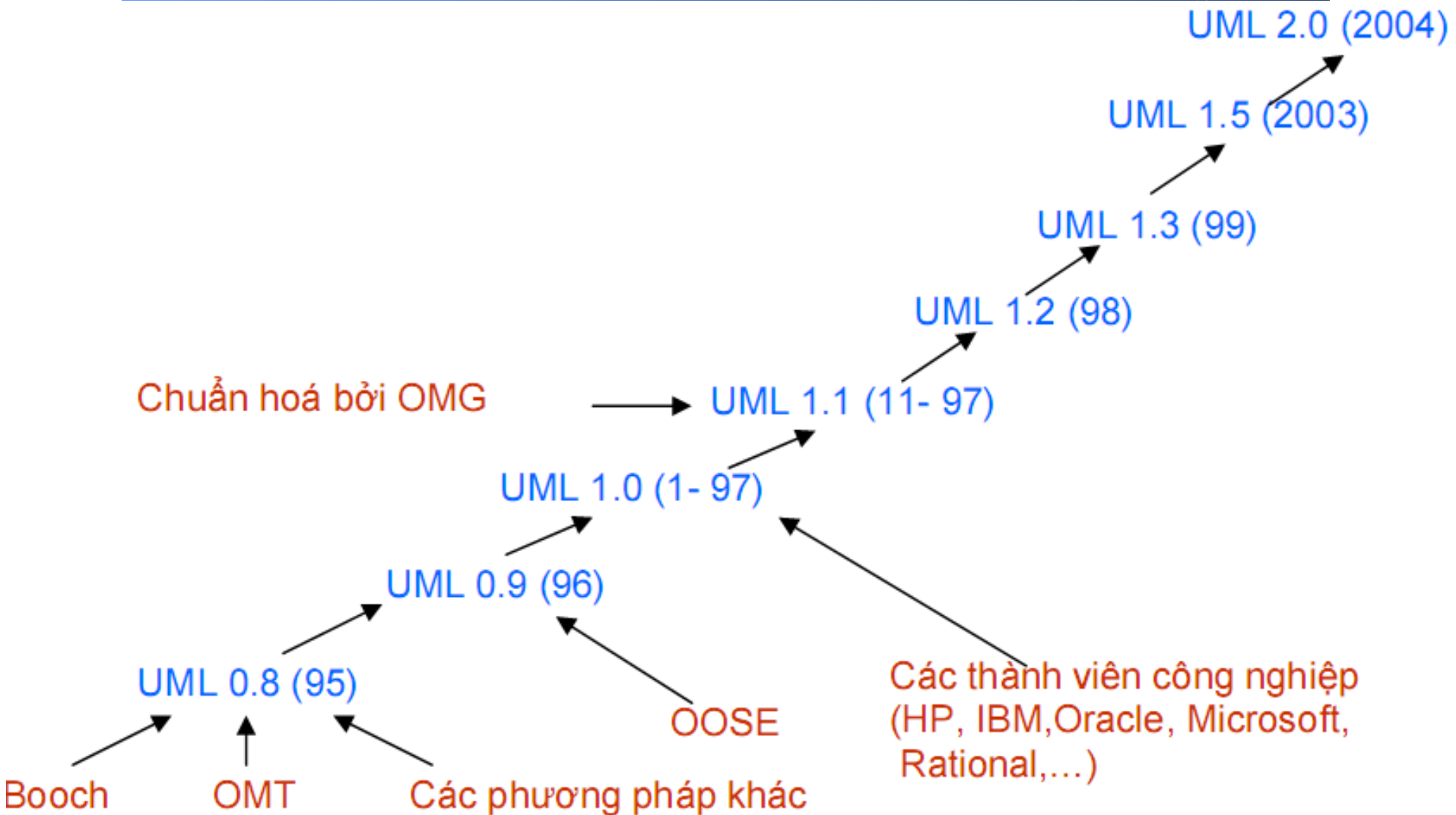
- Sau khi kết thúc chương sinh viên phải
  - Hiểu UML là gì?
  - Đặc tính của UML
  - Thành phần ngôn ngữ UML
  - Các sơ đồ trong UML

# Nội dung

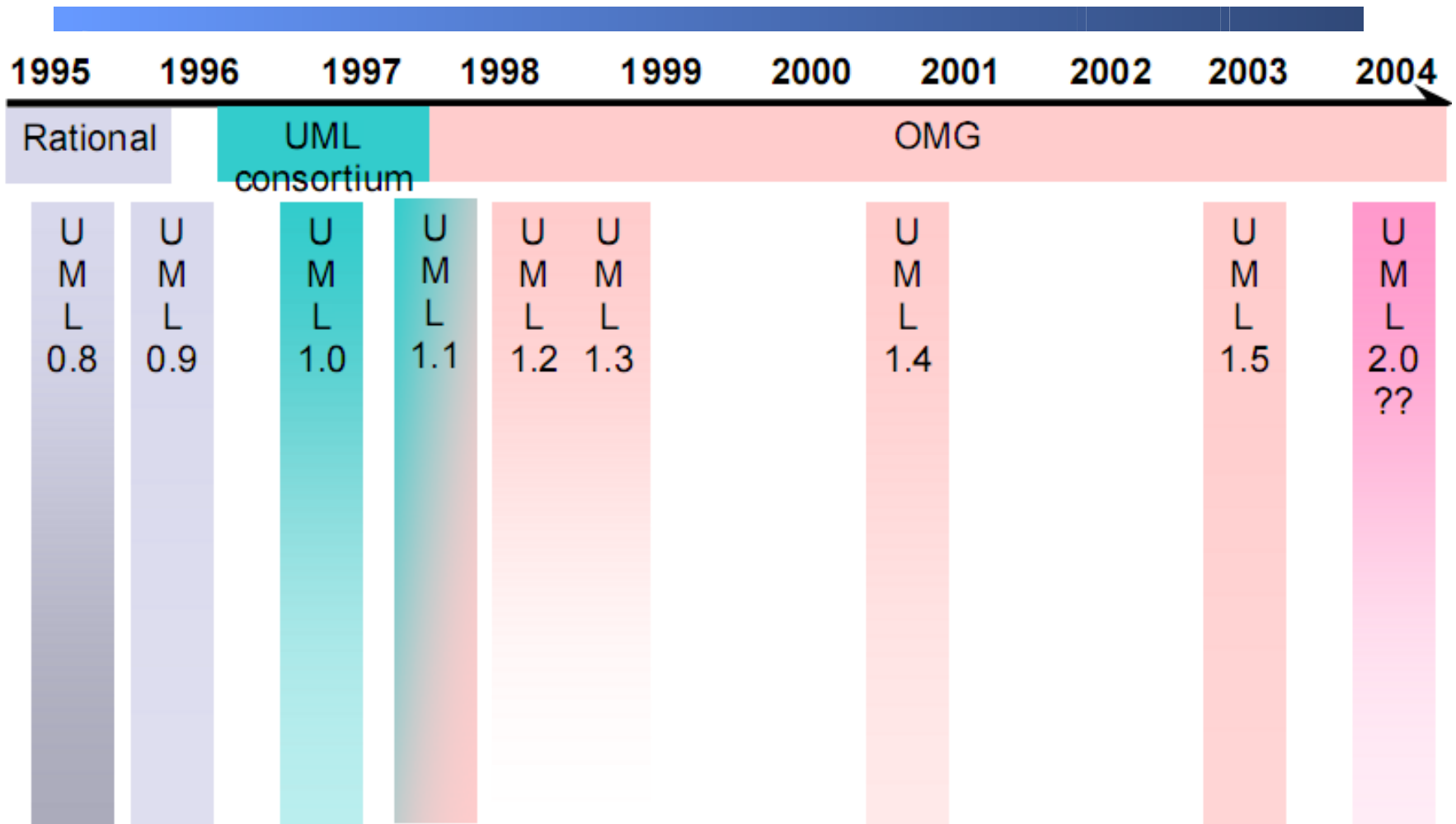
---

- Lịch sử UML

# Lịch sử UML



# Lịch sử UML



# UML là gì?

---

- Unified Modeling Language
- Hệ thống các ký hiệu, hình vẽ có ý nghĩa
- UML không phải là một ngôn ngữ lập trình
- Phải được sử dụng kết hợp với một tiến trình phương pháp luận.
- UML là một ngôn ngữ dùng để đặc tả, trực quan hoá, và tư liệu hoá phần mềm hướng đối tượng

# Đặc trưng của tiến trình sử dụng UML

---

- Nguyên do thất bại của phương pháp tuần tự:
  - Sự giả định ban đầu có sai sót
  - Thất bại trong việc kết hợp các nhân tố con người
  - Các hệ thống ngày càng lớn và thường hay thay đổi
  - Chúng ta vẫn còn đang trong giai đoạn thăm dò của công nghệ phần mềm, và không có nhiều kinh nghiệm.

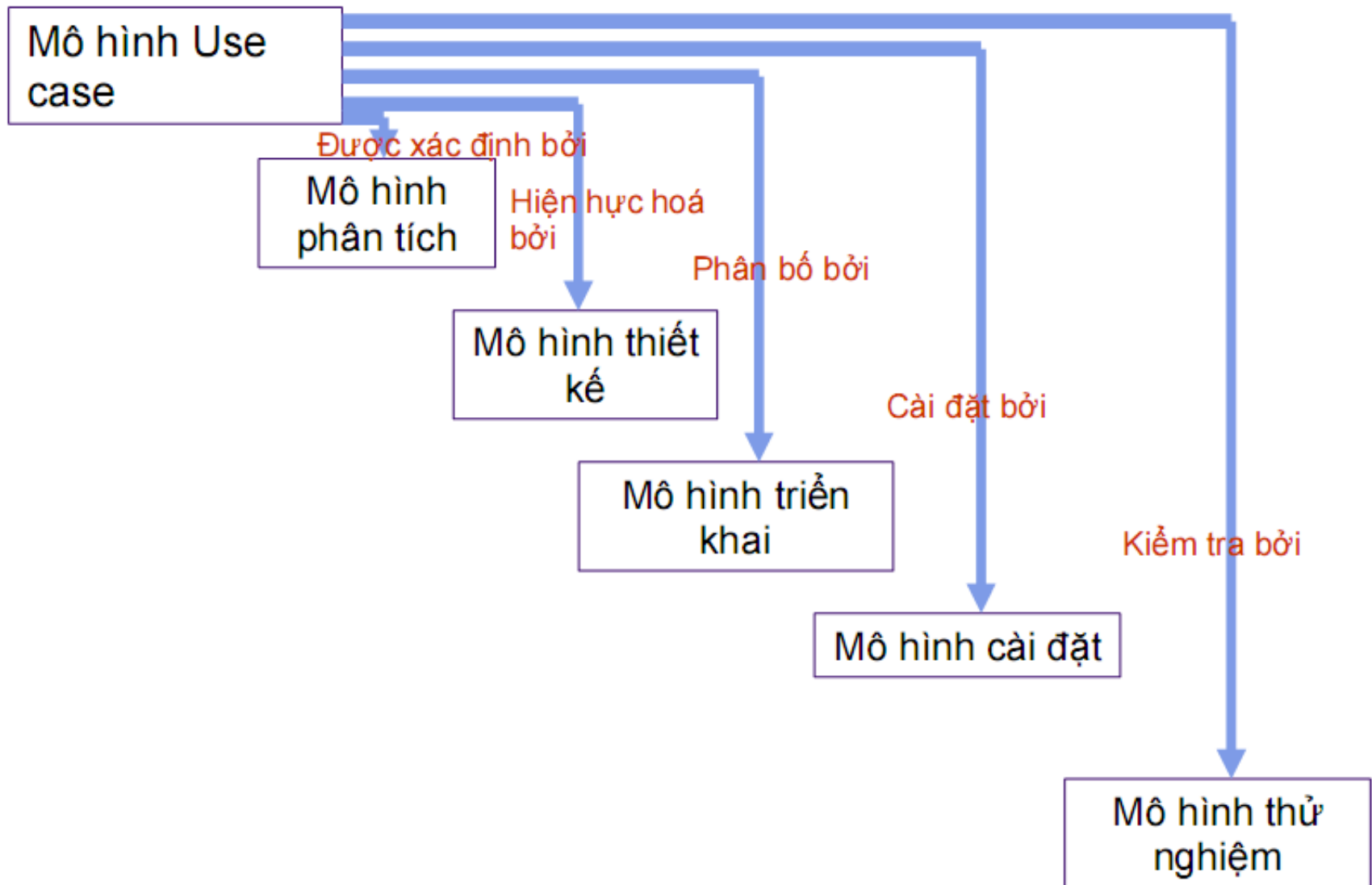
# Đặc trưng của tiến trình sử dụng UML

---

- Tính lặp (iterative)
- Gia tăng (incremental)
- Tập trung vào người dùng (user – concentrated)
- Hướng kiến trúc (well-defined structure)
- Các khung nhìn về hệ thống



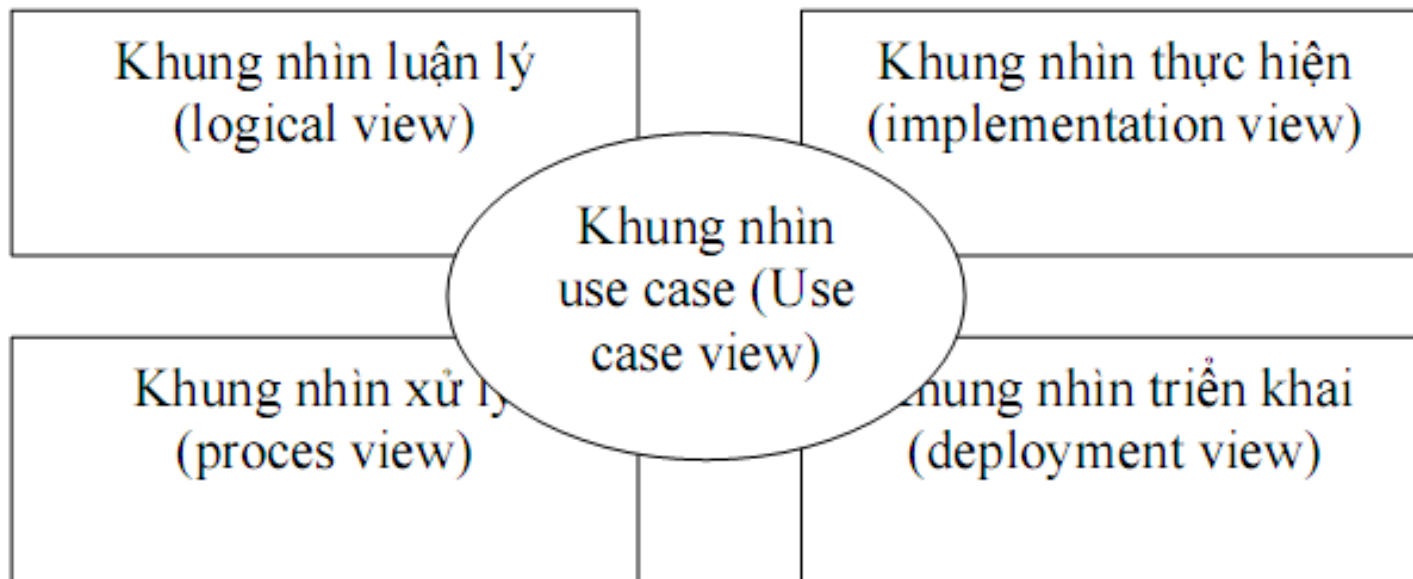
# Tập trung vào người dùng (user – concentrated)



# Khung nhìn (View)

**Người dùng**  
Chức năng

**Lập trình viên**  
Quản trị phần mềm



**Quản trị viên tích hợp hệ thống**  
Hiệu năng  
Tính cơ gián  
Thông lượng

**Thiết kế viên hệ thống**  
Hình thái hệ thống  
Chuyên giao, cài đặt  
Truyền thông

# Khung nhìn luận lý (logical view):

---

- Mô tả các yêu cầu chức năng của hệ thống,
- Sự trừu tượng của mô hình thiết kế và xác định các gói thiết kế chính, các subsystem và lớp chính.
- Trong UML khung nhìn này có thể được trình bày dùng sơ đồ lớp, sơ đồ đối tượng, sơ đồ mô tả các gói, hệ thống con.

# Khung nhìn thực hiện (implementation view)

---

- Mô tả tổ chức của các đơn thể (module) phần mềm tĩnh (như mã nguồn, tập tin dữ liệu, thành phần, tập tin thực thi, và các thành phần kèm theo khác) trong môi trường phát triển.
- Dùng sơ đồ thành phần để trình bày.

# Khung nhìn xử lý (process view)

---

- Mô tả các khía cạnh xảy ra đồng thời của hệ thống thời gian thực (run-time) (tasks, threads, processes ..).
- Tập trung vào sự đồng hành, song song, khởi động và đóng hệ thống, khả năng chịu đựng hư hỏng, và sự phân tán các đối tượng.

# Khung nhìn triển khai (deployment):

---

- Cho thấy các tập tin thực thi và các thành phần khác nhau được triển khai trên các hệ thống như thế nào.
- Nó giải quyết các vấn đề như triển khai, cài đặt, và tốc độ.
- Trong UML, khung nhìn này có thể sử dụng sơ đồ triển khai để mô tả.

# Khung nhìn use-case:

---

- Đóng một vai trò đặc biệt đối với kiến trúc. Nó chứa một vài kịch bản hay use case chủ yếu.
- dùng để khám phá và thiết kế kiến trúc và xác nhận các khung nhìn khác nhau.
- Khung nhìn này có thể sử dụng sơ đồ use case để minh họa.

# Các sơ đồ trong UML

---

- Các sơ đồ mô tả khía cạnh tĩnh
  - Sơ đồ đối tượng (object diagram)
  - Sơ đồ lớp (class diagram)
  - Sơ đồ use case (use case diagram)
  - Sơ đồ thành phần (component diagram)
  - Sơ đồ triển khai (deployment diagram)

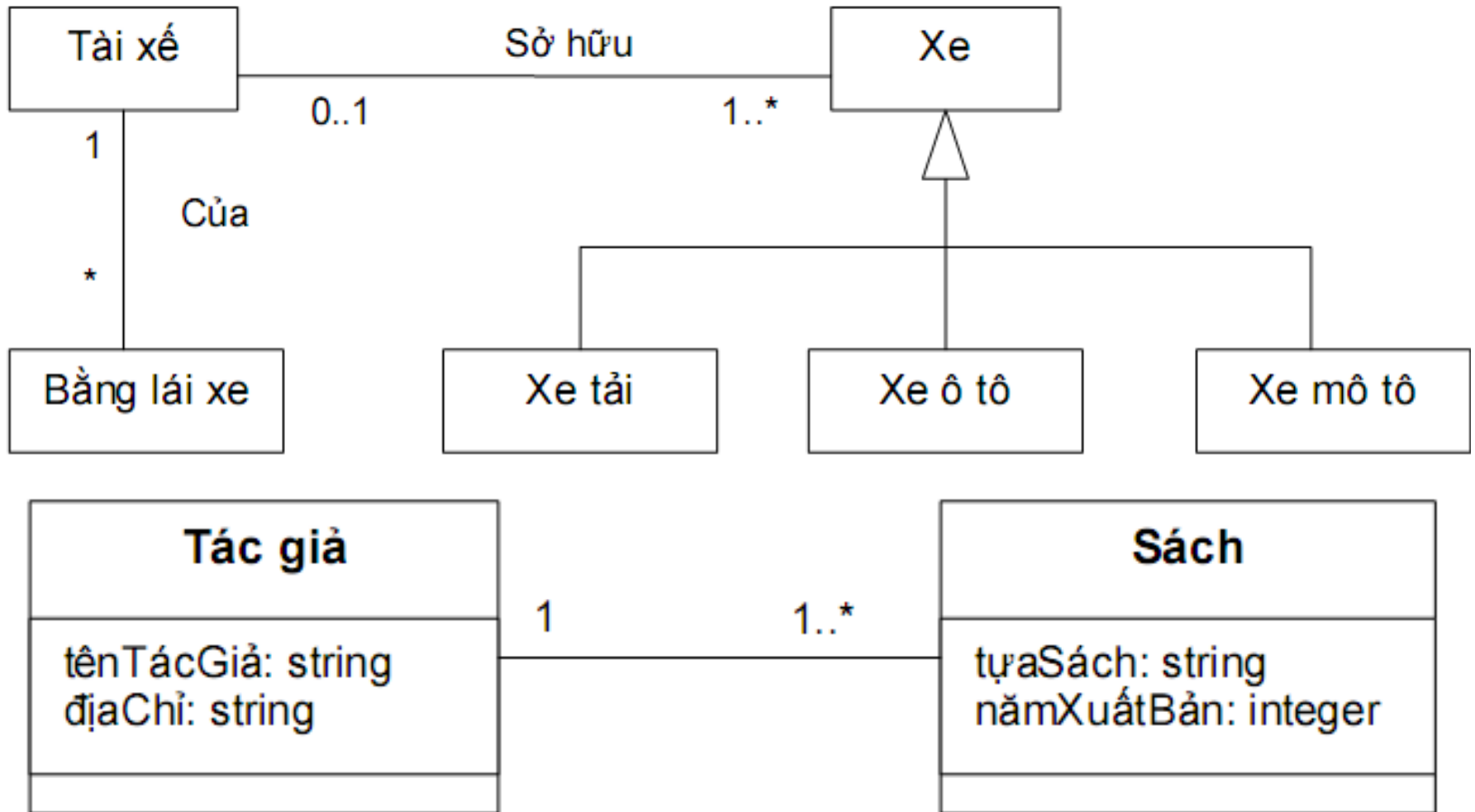


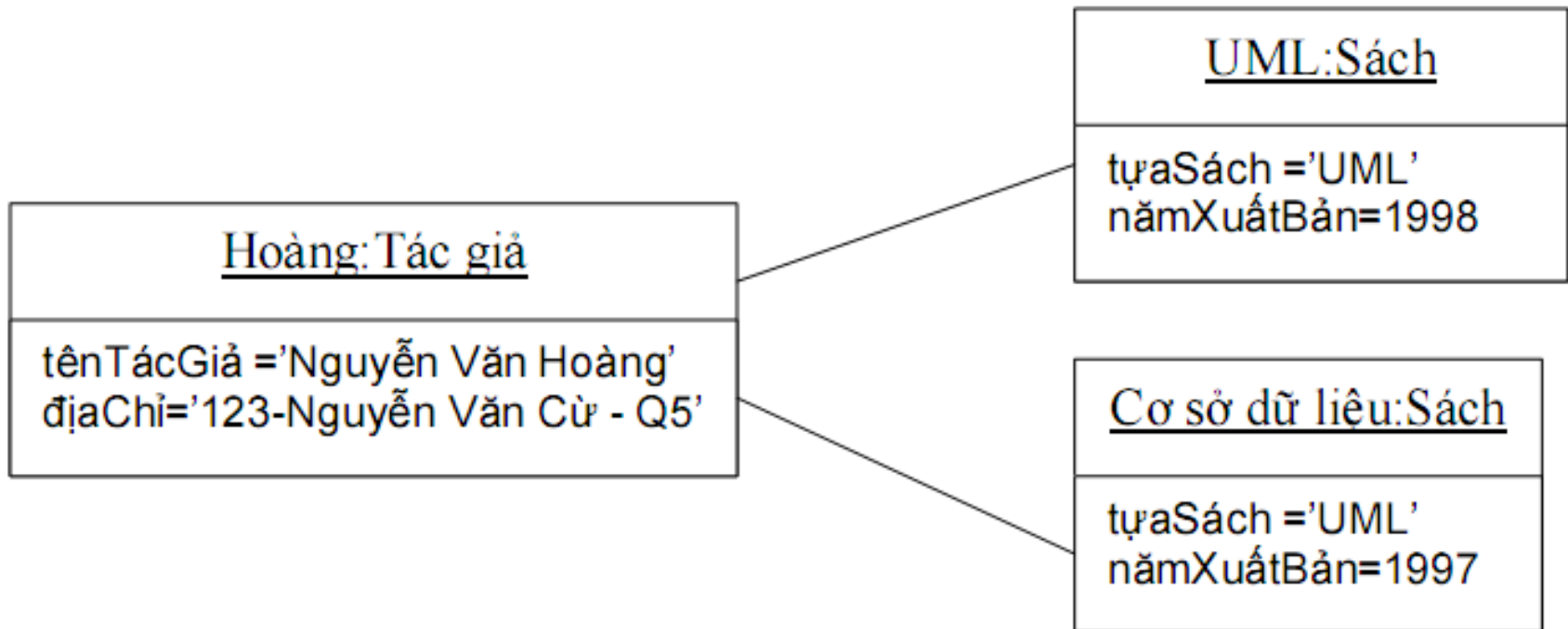
# Các sơ đồ trong UML

---

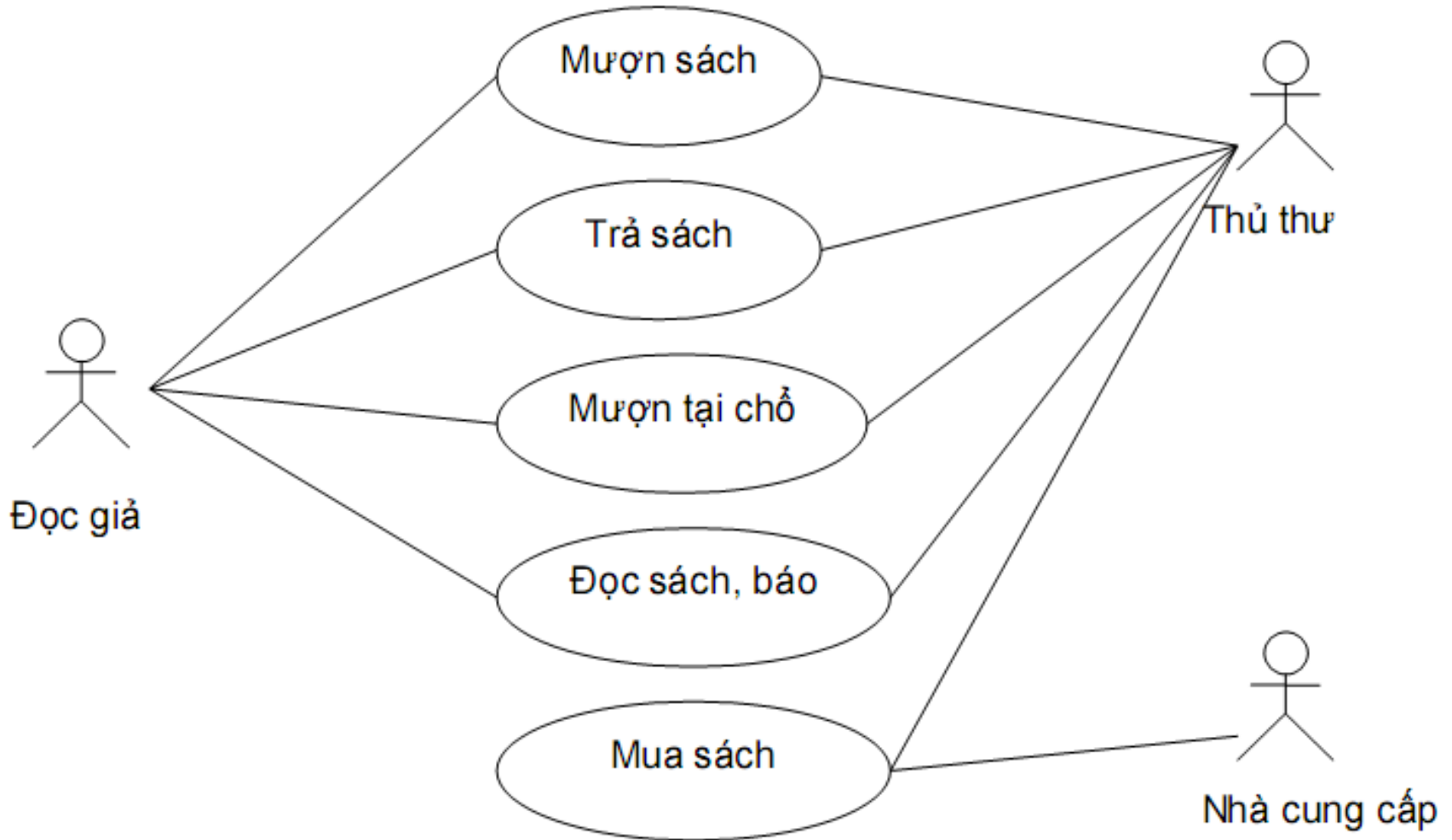
- Các sơ đồ mô tả khía cạnh động
  - Các sơ đồ tương tác (interaction diagram)
    - Sơ đồ tuần tự (sequence diagram)
    - Sơ đồ hợp tác (collaboration diagram)
  - Sơ đồ hoạt động (activity diagram)
  - Sơ đồ chuyển dịch trạng thái (state transition diagram)

# Sơ đồ lớp và đối tượng

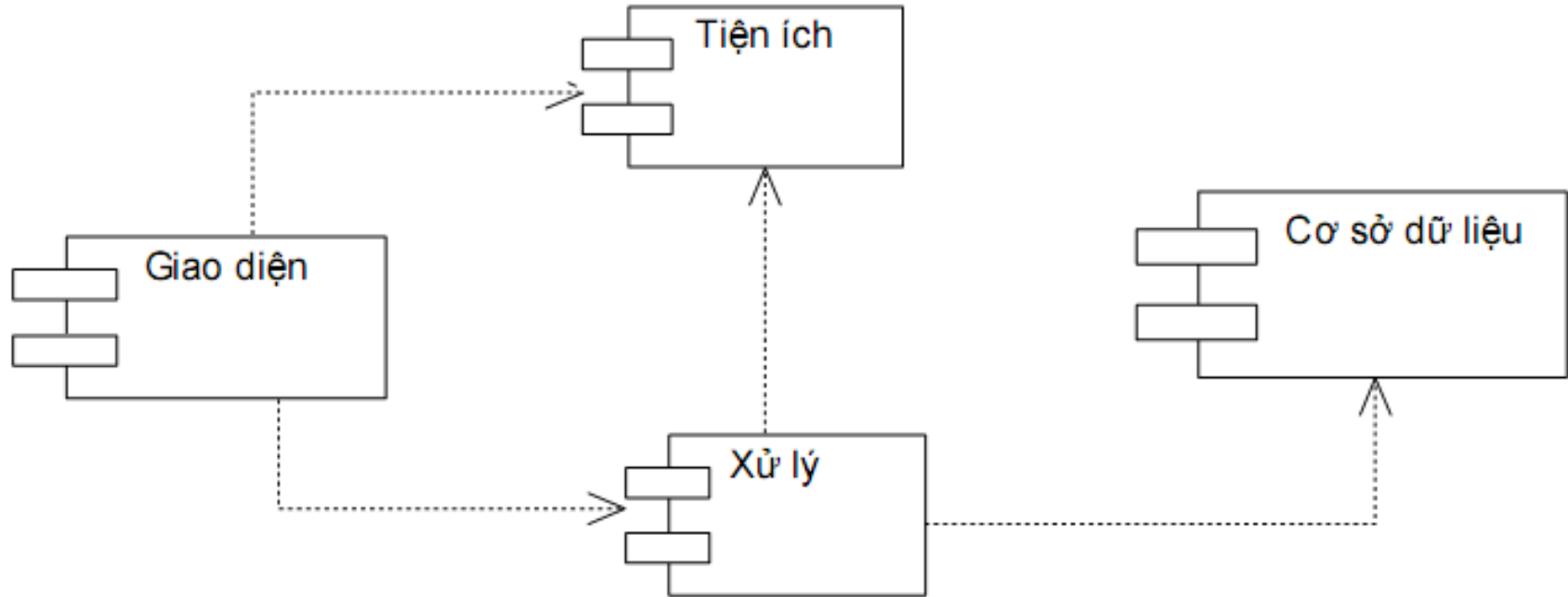




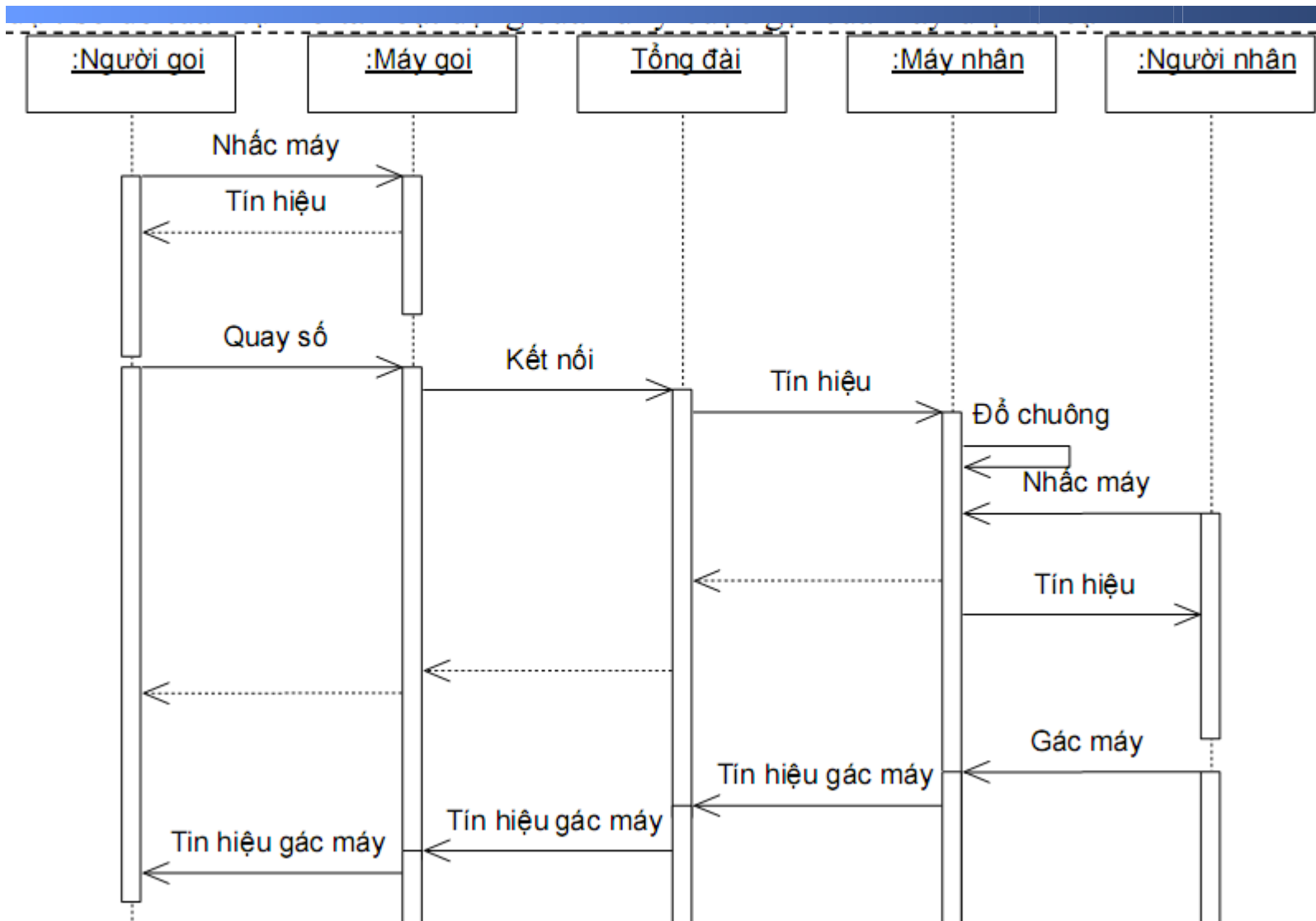
# Sơ đồ use case



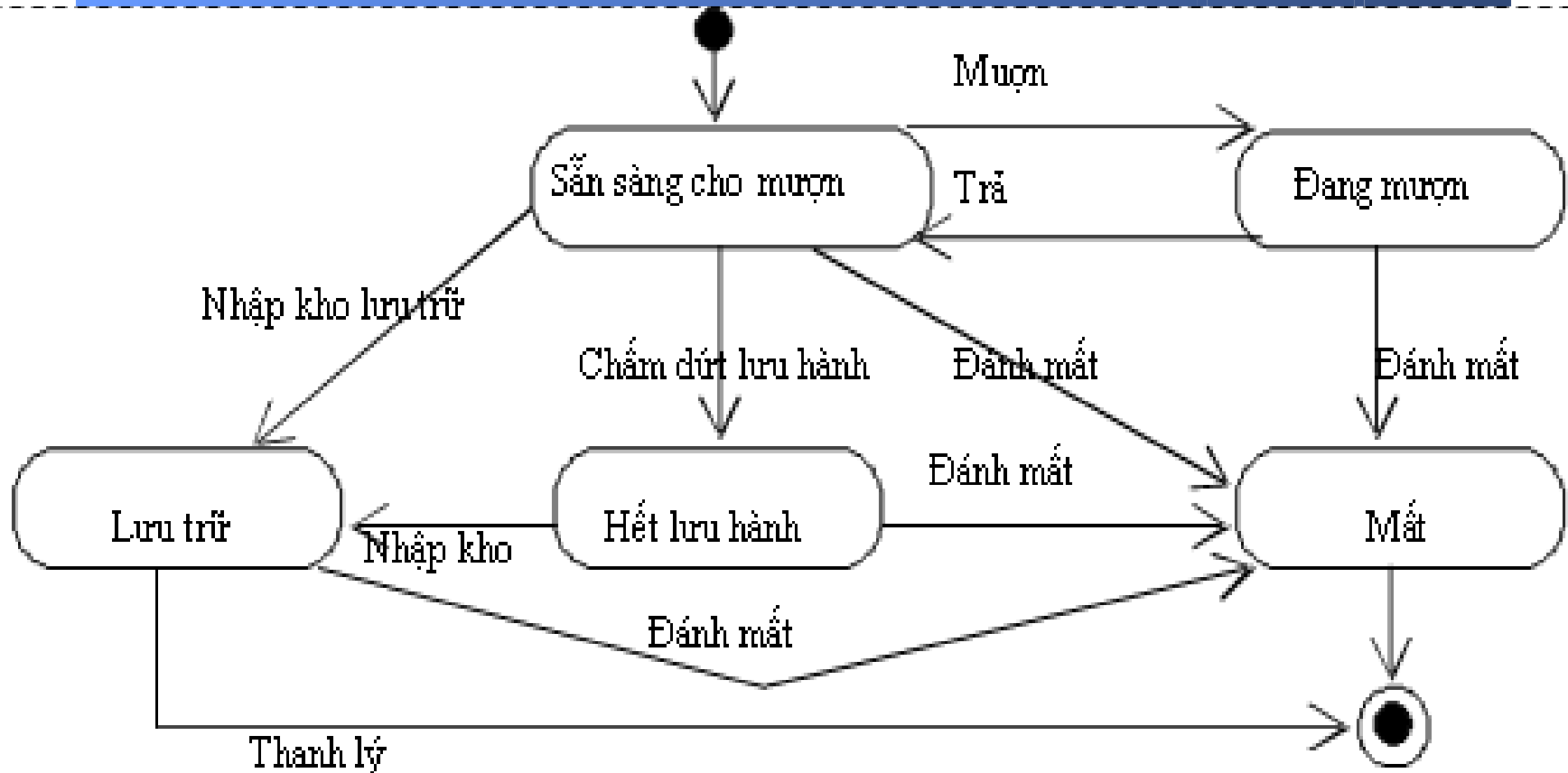
# Sơ đồ thành phần



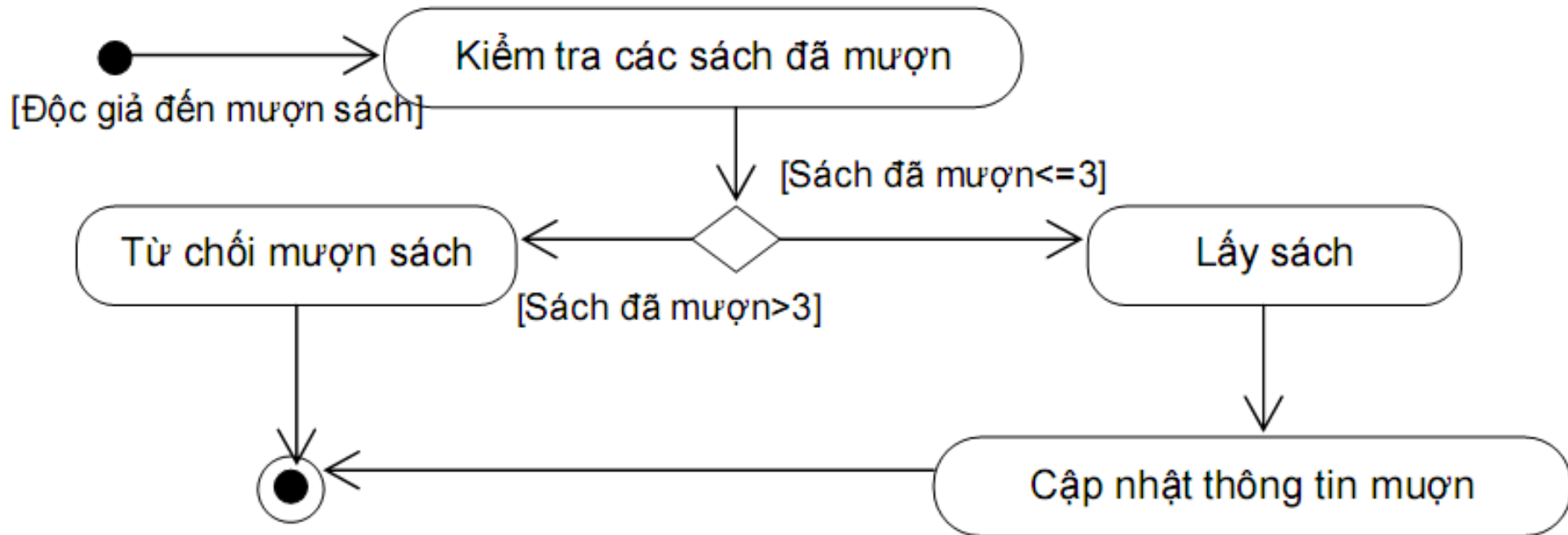
# Sơ đồ tuần tự và sơ đồ hợp tác



# Sơ đồ chuyển đổi trạng thái



# Sơ đồ hoạt động





# Sơ đồ triển khai

