



# Chủ đề 3: Use Case Diagram



# Nội dung

- Tổng quan về xác định yêu cầu
- Các loại yêu cầu
- Xác định yêu cầu
- Mô hình hoá yêu cầu người dùng sử dụng Use Case Diagram



# **YÊU CẦU CỦA NGƯỜI DÙNG**



# Yêu cầu phần mềm

- **Yêu cầu người dùng - User requirements**
  - Các phát biểu bằng ngôn ngữ tự nhiên cộng với các sơ đồ về các dịch vụ mà hệ thống cung cấp và các ràng buộc về vận hành.
  - Được viết cho khách hàng.
- **Yêu cầu hệ thống – System requirements**
  - Một tài liệu có cấu trúc bao gồm các mô tả chi tiết về các chức năng và dịch vụ của hệ thống cùng với các ràng buộc về vận hành.
  - Định nghĩa cái gì cần được cài đặt
    - Có thể là một phần của một hợp đồng giữa khách hàng và người nhận thầu.



# Xác định yêu cầu

- Khi nào thực hiện **xác định yêu cầu**?
- **Quy trình** thực hiện xác định yêu cầu ?
- Đặc tả yêu cầu người dùng như thế nào ?



## Xác định yêu cầu (tt)

- **Xác định yêu cầu** được thực hiện trong các trường hợp sau:
  - Có yêu cầu từ phía khách hàng
  - Kế hoạch thực hiện dự án được chấp nhận (Project proposed)



# Xác định và thu thập yêu cầu

- Phân loại yêu cầu: Có 2 loại yêu cầu chính:
  - Yêu cầu chức năng (functional requirements):
    - Là danh sách các công việc sẽ được thực hiện trên máy tính cùng với các thông tin mô tả tương ứng.
  - Yêu cầu phi chức năng (non--functional requirements)
    - Là các yêu cầu liên quan đến chất lượng phần mềm. (Yêu cầu về chất lượng PM)
    - Là sự ràng buộc trên cách thức thực hiện yêu cầu chức năng.



# Ví dụ Phân loại yêu cầu

Yêu cầu	Loại
Thống kê doanh thu bán hàng	Chức năng
Tra cứu sách	Chức năng
Xử lý và phản hồi nhanh	Phi chức năng
Bảo mật mật khẩu	Phi chức năng





# Yêu cầu chức năng

- Được chia làm 2 loại:
  - Yêu cầu chức năng nghiệp vụ.
  - Yêu cầu chức năng hệ thống.



# Yêu cầu chức năng nghiệp vụ

- Các chức năng của PM
  - tương ứng với công việc có thật trong thế giới thực.
- Có **4 loại chức năng** chính ứng với **4 loại nghiệp vụ** thông dụng trong các lĩnh vực:
  - Chức năng **Lưu trữ**
  - Chức năng **Tra cứu**
  - Chức năng **Tính toán**
  - Chức năng **Kết xuất**



# Yêu cầu chức năng nghiệp vụ (tt)

- Chức năng lưu trữ:

- Tương ứng với các công việc ghi chép thông tin sổ sách.
- VD: ghi nhận điểm thi kết thúc học phần của sinh viên với qui định điểm số (từ 0 đến 10)

- Chức năng tra cứu:

- Tương ứng với các công việc tìm kiếm và xem thông tin tương ứng.
- VD: tìm sách và xem tình trạng sách

- Chức năng tính toán:

- Tương ứng với các công việc tính toán (theo qui định, công thức cho trước)
- VD: tính tiền phạt trả sách trễ hạn theo qui định phạt

- Chức năng kết xuất:

- Tương ứng với các công việc lập các báo cáo (theo biểu mẫu cho trước)
- VD: Lập báo cáo thống kê về số lượt mượn sách theo từng thể loại trong năm



# Yêu cầu chức năng hệ thống

- Là các chức năng PM phải **phát sinh thêm khi tiến hành các công việc trên máy tính** thay vì trong thế giới thực.
- Là các chức năng **không tương ứng với bất kỳ công việc nào trong thế giới thực** (có nhu cầu nhưng không thể thực hiện thủ công).
- Một số chức năng hệ thống thông dụng sau:
  - Phân quyền sử dụng giữa các loại người dùng.
  - Sao lưu, backup, phục hồi thông tin.
  - Định cấu hình thiết bị, ngày giờ làm việc.
  - Mô phỏng hoạt động thế giới thực.
  - Báo động, nhắc nhở người dùng.



# Yêu cầu chức năng hệ thống (tt)

- Phân quyền sử dụng giữa các loại người dùng.
  - VD: Phân quyền cho 3 loại người dùng trong phần mềm quản lý thư viện.
    - Quản trị hệ thống: có quyền sử dụng tất cả các chức năng.
    - Thủ thư: chỉ sử dụng các chức năng liên quan đến việc mượn trả sách.
    - Độc giả: chỉ sử dụng chức năng tra cứu.
- Sao lưu, backup, phục hồi thông tin.
  - VD: Sao lưu thông tin các học sinh đã ra trường và chỉ phục hồi khi cần thiết.



# Phân quyền sử dụng

**Thủ thư và Quản trị phải đăng nhập trước khi sử dụng**

STT	Chức năng	Độc giả	Thủ thư	Quản trị Hệ thống
1	Lập thẻ độc giả		X	X
2	Tiếp nhận sách mới		X	X
3	Tra cứu sách	X	X	X
4	Cho mượn sách		X	X
5	Nhận trả sách		X	X
6	Lập báo cáo		X	X
7	Thay đổi qui định			X
8	Sao lưu, phục hồi			X
9	...			X



## Yêu cầu chức năng hệ thống (tt)

- Định cấu hình thiết bị, ngày giờ làm việc.
  - Chọn loại máy in, kích thước giấy, niên khoá hiện hành, tháng làm việc hiện tại...
- Báo động, nhắc nhở người dùng.
  - Nhắc nhở thủ thư gửi giấy báo đòi sách khi có độc giả mượn sách quá hạn.
  - Báo động khi có khách hàng thiếu nợ quá lâu hay số tiền quá lớn.



# Yêu cầu phi chức năng

- Yêu cầu về sản phẩm:
  - Yêu cầu khả dụng
  - Yêu cầu hiệu quả (về hiệu năng, về không gian), tốc độ xử lý
  - Yêu cầu về độ tin cậy
  - Yêu cầu khả chuyển
- Yêu cầu về tổ chức:
  - Yêu cầu chuyển giao
  - Yêu cầu triển khai
  - Yêu cầu về chuẩn
- Yêu cầu mở rộng:
  - Yêu cầu hoạt động bên trong
  - Yêu cầu đạo đức
  - Yêu cầu pháp lý
  - Yêu cầu về cá nhân
  - Yêu cầu an toàn





# Xác định và thu thập yêu cầu

- **Khái niệm:**
  - Quá trình nắm bắt yêu cầu
  - Làm rõ yêu cầu
  - Gợi mở yêu cầu
- **Kỹ thuật thu thập yêu cầu:**
  - Phỏng vấn
  - Bảng câu hỏi
  - Nghiên cứu các tài liệu
  - Quan sát thực tế
  - Lập kịch bản (scenarios)
  - Phân tích thiết kế nhóm (JAD)

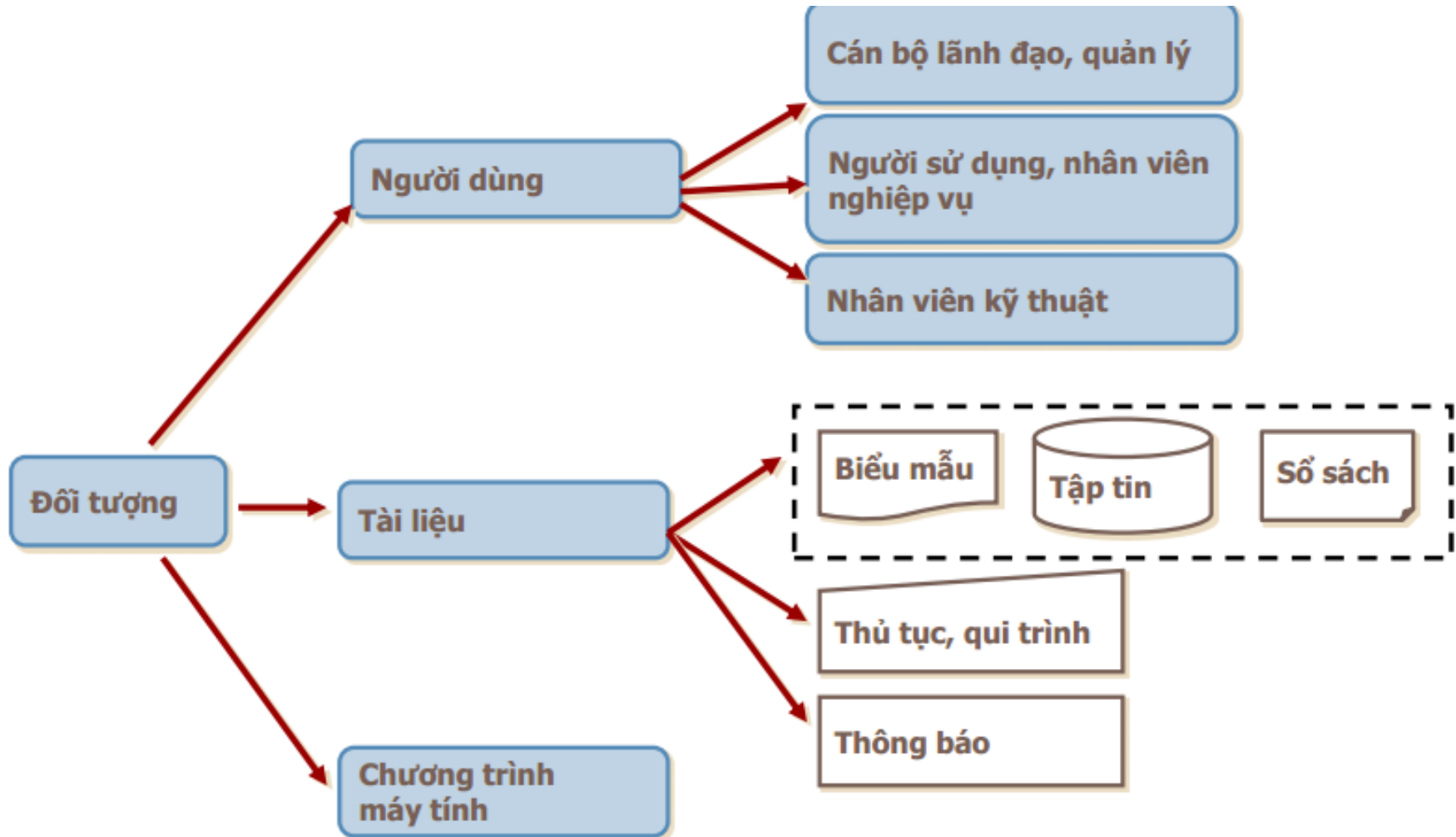


# Nội dung khảo sát

- Hiện trạng nghiệp vụ
- Hiện trạng tổ chức
- Hiện trạng tin học



# Đối tượng khảo sát





# Phỏng vấn

- Phỏng vấn cá nhân/phỏng vấn nhóm?
  - Phỏng vấn cá nhân
  - Phỏng vấn nhóm
- Phỏng vấn tự do/phỏng vấn có định hướng?
  - Phỏng vấn tự do
    - Người được hỏi có cảm giác thoải mái, cung cấp nhiều thông tin sâu sắc
    - Nguy cơ: không có được những thông tin cần thiết, thông tin khó hệ thống được
  - Phỏng vấn có định hướng
    - Người được hỏi có thể cảm thấy không thoải mái, ít có khả năng ghi nhận được nhận xét, ý kiến, suy nghĩ riêng của người được phỏng vấn, ít cảm nhận được thái độ của họ đối với hiện trạng.
    - Có thể định hướng nội dung cần tìm hiểu, có thể hệ thống hóa các vấn đề ghi nhận được



# Phỏng vấn (tt)

- Làm việc với cấp lãnh đạo để nắm mục tiêu của hệ thống phần mềm cần xây dựng, những đối tượng cần phỏng vấn
- Yêu cầu cấp lãnh đạo thông báo xuống các phòng ban, đơn vị để hợp tác
- Phân tích để xác định đúng và đủ những đối tượng cần phỏng vấn
- Hẹn lịch làm việc
- Xác định trước vị trí/trách nhiệm của người sắp phỏng vấn



# Phỏng vấn (tt)

- Khi tìm hiểu, cần ghi nhận các thông tin:
  - Nội dung: cái gì?
  - Bao giờ có: thời gian + thời hạn
  - Bằng cách nào có nội dung thông tin đó
  - Nội dung đó ở dạng gì?
  - Đánh giá của người được phỏng vấn về tình hình hiện tại thực hiện nghiệp vụ
- Không nên:
  - Đưa nhận xét cá nhân của người phỏng vấn
  - Dùng thuật ngữ/ngôn ngữ Tin học



# Phỏng vấn (tt)

- Sau khi phỏng vấn

- Lập báo cáo phỏng vấn sau khi kết thúc cuộc phỏng vấn. Thường theo mẫu:

## Báo cáo phỏng vấn

- Biên bản được duyệt bởi
- Người thực hiện
- Những người được phỏng vấn
- Ngày phỏng vấn
- Mục tiêu
- Tóm tắt nội dung
- Vấn đề mở
- Nội dung chi tiết



# Sử dụng bảng câu hỏi

- Phải trình bày rõ:
  - Mục đích của bảng câu hỏi,
  - Mục đích sử dụng những thông tin trong bảng câu hỏi,
  - Tính bảo mật thông tin trả lời (không tiết lộ ai là người cung cấp thông tin, không để lộ ra ngoài tổ chức...)
- Hướng dẫn cách điền: rất cần thiết, cần lưu ý để tránh hiểu nhầm
- Thời hạn trả về
  - Cần nhắc khi gần đến thời hạn
- Câu hỏi trình bày rõ ràng
- Hình thức bảng câu hỏi phải dễ dàng để xử lý tự động
- Cần để dành chỗ để ghi câu trả lời.
  - Thêm chỗ cho lời bình
  - Không phải chỉ ở cuối trang, hay cuối bảng câu hỏi,
  - Nên dự kiến những câu hỏi nào sẽ có ý kiến thêm thì nên có sẵn chỗ để ghi lời bình ngay dưới câu hỏi đó)





# Nghiên cứu các tài liệu

- Các tài liệu (có thể tìm hiểu những văn bản chung)
- Những quy định nội bộ
- Các báo cáo liên quan
- Những quy định về quy trình nghiệp vụ
  - Rất khó có đầy đủ văn bản quy định về quy trình nghiệp vụ
  - Đơn vị đạt chuẩn ISO?
- Những quy định “bất thành văn” !!!
- Thường dễ tiến hành hơn kỹ thuật phỏng vấn hay bảng câu hỏi
- Thường được tiến hành trước làm cơ sở chuẩn bị cho việc phỏng vấn hay dùng bảng câu hỏi



# Quan sát thực tế

- Tiến hành sau cùng (nếu cần thiết)
- Kiểm tra lại:
  - Đã hiểu đúng nghiệp vụ hiện tại?
  - Có những ngoại lệ?
  - Phát hiện những khó khăn, lỗ hổng trong quy trình nghiệp vụ



# Phương pháp lập kịch bản

- Ý tưởng:
  - Xem phần mềm như một vở kịch
  - Phân chức năng thành những tình huống sử dụng
  - Mỗi tình huống sử dụng cho một kịch bản diễn
- Ưu điểm:
  - Yêu cầu được gom nhóm và có cấu trúc
  - Yêu cầu được mô tả chi tiết và rõ ràng



# Phương pháp lập kịch bản (tt)

Chức năng (màn diễn)	Tình huống (cảnh diễn)	Mô tả (kịch bản)
Đăng nhập	Người dùng đăng nhập thành công	<ul style="list-style-type: none"> <li>Hệ thống: yêu cầu tên đăng nhập và mật khẩu</li> <li>Người dùng: cung cấp thông tin rồi đề nghị đăng nhập</li> <li>Hệ thống: Chuyển tới màn hình chính</li> </ul>
	Người dùng đăng nhập thành công	<ul style="list-style-type: none"> <li>Hệ thống: yêu cầu tên đăng nhập và mật khẩu</li> <li>Người dùng: cung cấp thông tin rồi đề nghị đăng nhập</li> <li>Hệ thống: thông báo sai và đề nghị cung cấp lại</li> </ul>
	Quản trị đăng nhập thành công	<ul style="list-style-type: none"> <li>Hệ thống: yêu cầu tên đăng nhập, mật khẩu và câu hỏi bí mật</li> <li>Người dùng: cung cấp thông tin rồi đề nghị đăng nhập</li> <li>Hệ thống: Chuyển tới màn hình chính</li> </ul>
	Quản trị đăng nhập thành công	<ul style="list-style-type: none"> <li>Hệ thống: yêu cầu tên đăng nhập, mật khẩu và câu hỏi bí mật</li> <li>Người dùng: cung cấp thông tin rồi đề nghị đăng nhập</li> <li>Hệ thống: thông báo sai và đề nghị cung cấp lại</li> </ul>



# Phân tích thiết kế nhóm - JAD

- JAD – Joint Application Design
- Kỹ thuật áp dụng cho các giai đoạn **phân tích yêu cầu và đặc tả**
- Các nhà phát triển và khách hàng làm việc như một nhóm chung và **có trách nhiệm chung** đối với kết quả đầu ra.
- Nhóm làm việc sẽ **thảo luận** các yêu cầu cần có, thiết kế các màn hình và báo cáo, **xây dựng mô hình định khung nhanh**, rút ra các đặc tả
- Chủ yếu dựa trên sự **đồng thuận** (consensus)



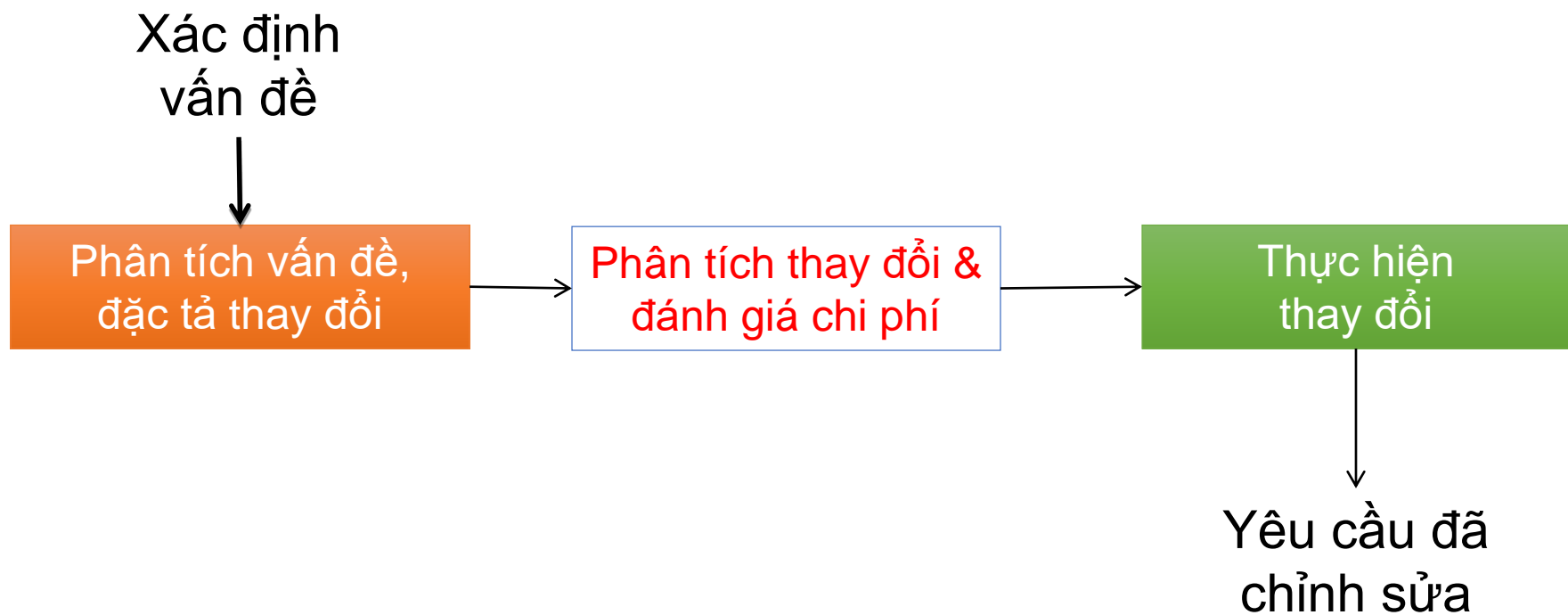
# Kỹ thuật thu thập yêu cầu

- Một số tiêu chí để chọn kỹ thuật phù hợp

	Phỏng vấn	JAD	Bảng câu hỏi	Quan sát	Phân tích tài liệu có sẵn
1) Loại thông tin	Hiện có và sắp có	Hiện có và sắp có	Hiện có	Hiện có	Hiện có
2) Độ sâu của thông tin	Lớn	Lớn	Vừa	Thấp	Thấp
3) Bề rộng của thông tin	Thấp	Vừa	Lớn	Thấp	Lớn
4) Tính tích hợp của thông tin	Thấp	Lớn	Thấp	Thấp	Thấp
5) Có liên quan đến NSD	Vừa	Cao	Thấp	Thấp	Thấp
6) Chi phí	Vừa	Cao	Thấp	Thấp	Thấp



# Quản lý thay đổi





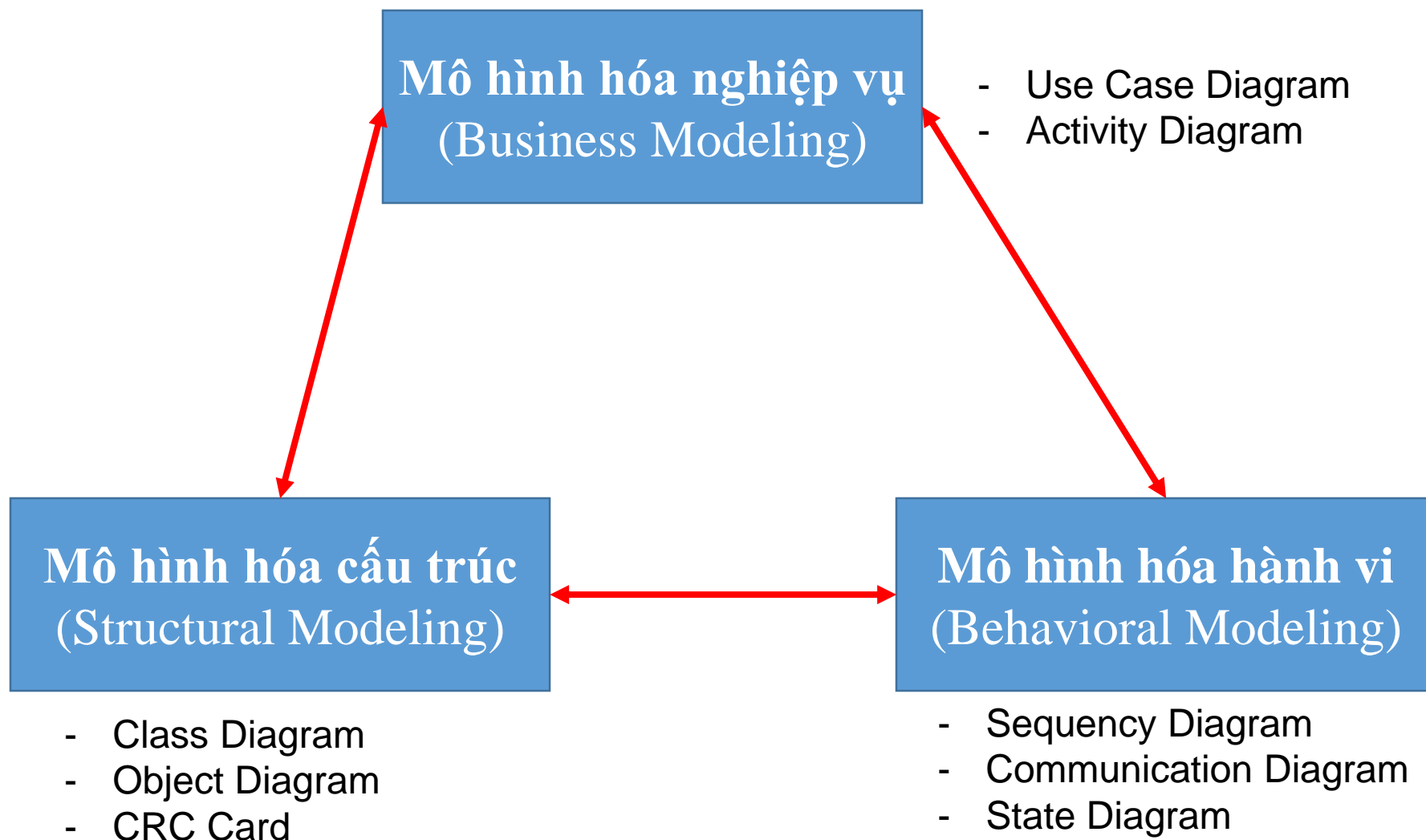
# Quản lý thay đổi yêu cầu

- Nên áp dụng cho tất cả các thay đổi được đề xuất đối với bộ yêu cầu.
- Các giai đoạn chính
  - **Phân tích vấn đề:** Thảo luận về vấn đề của các yêu cầu và đề xuất thay đổi; Bổ sung chi tiết; Chốt lại những điểm sẽ thay đổi.
  - **Phân tích thay đổi và đánh giá chi phí.** Đánh giá hiệu ứng của thay đổi đối với các yêu cầu khác; Ra quyết định có thực hiện thay đổi hay không.
  - **Thực hiện thay đổi.** Cập nhật tài liệu yêu cầu và các tài liệu khác để thực hiện thay đổi đã xét.



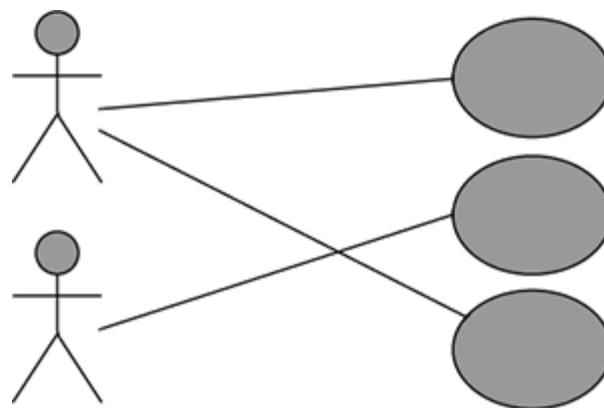


# Mô hình hóa nghiệp vụ





# Use Case Diagram





# Ví dụ

## Xét phần mềm Quản lý học sinh cấp III







STT	Yêu cầu	Nhóm người dùng
1	Tiếp nhận học sinh	Giáo vụ?
2	Lập danh sách lớp	Giáo vụ?
3	Tra cứu học sinh	Mọi người? Phụ huynh? Học sinh?
4	Nhận bảng điểm môn	Giáo viên? Giáo vụ?
5	Xem báo cáo tổng kết	Ban giám hiệu?
6	Thay đổi quy định	Ban giám hiệu? Quản trị hệ thống?

- ❖ **Một nhóm người dùng** tương ứng với **một Actor**
- ❖ **Mỗi Nhóm người dùng (Actor)** được quyền sử dụng **một hay nhiều chức năng trong hệ thống**
- ❖ **Một chức năng** có thể cho phép **nhiều Nhóm người dùng** sử dụng
- ❖ **Nhiều nhóm người dùng** có cùng các **quyền hạn giống nhau**
- ➔ Nên xét là **1 Actor hay nhiều Actor?**
- ❖ Việc xác định Actor phụ thuộc **ngữ cảnh** và **quy trình thực tế**



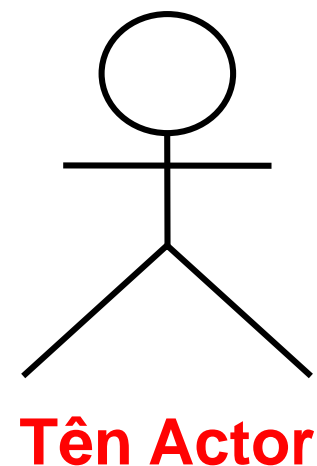
# Ví dụ

## Xét phần mềm Quản lý thư viện

STT	Yêu cầu		Nhóm người dùng
1	Lập thẻ độc giả		Thủ thư
2	Nhận sách mới		Thủ thư
3	Tra cứu sách		Thủ thư? Độc giả? Khách bất kỳ?
4	Lập phiếu mượn		Thủ thư
5	Nhận trả sách		Thủ thư
6	Thay đổi quy định		Thủ thư? Quản trị hệ thống?



# Actor ⇔ Phần cứng ngoại vi



- Tác nhân **BÊN NGOÀI** hệ thống
- Có tương tác với hệ thống



Phần mềm khác





# Ví dụ

## • Ví dụ:

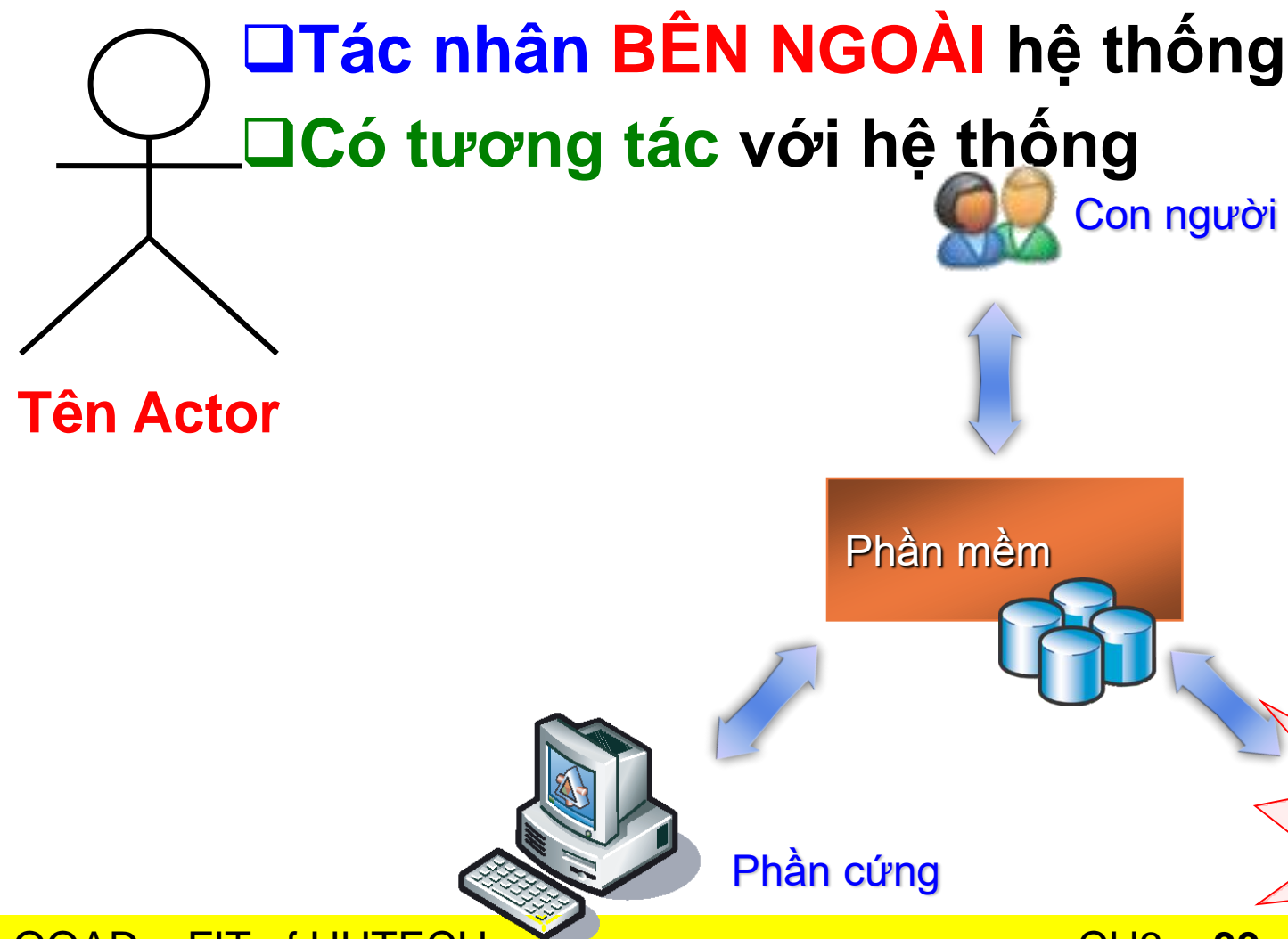
- Phần mềm quản lý Siêu thị:
  - Đọc thông tin từ thiết bị đọc mã vạch
- Phần mềm quản lý cửa tự động:
  - Đọc thông tin từ camera
  - Phát lệnh điều khiển mở cửa
- Phần mềm quản lý ra vào các phòng trong công sở
  - Đọc tín hiệu từ đầu đọc thẻ từ
  - Phát lệnh điều khiển mở cửa
- Phần mềm chống trộm
  - Đọc tín hiệu từ camera, sensor
  - Phát lệnh điều khiển ra loa, đèn, điện thoại...

Các thiết bị ngoại vi  
mà phần mềm  
cần tương tác

Có cần liệt kê  
tất cả thiết bị ngoại vi?



# Actor ⇔ Phần mềm khác





## Ví dụ

- Kết xuất/nạp dữ liệu từ Excel
- Kết xuất dữ liệu báo cáo ra phần mềm gửi email (Microsoft Outlook, Outlook Express...)
- Phần mềm trung gian kết nối để chuyển đổi email từ dạng Web-based sang POP3 (ví dụ Yahoo!Pop)
- ...

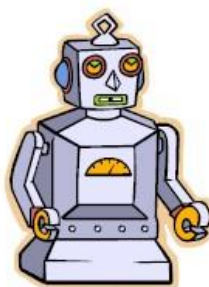




# Tim Actor?



Humans



Machines



External systems



Organizational Units



Sensors



# Nhận diện các Actor

Trả lời một số câu hỏi như:

- Ai là người sử dụng chức năng chính của hệ thống?
- Ai cần sự hỗ trợ từ hệ thống để thực hiện công việc thường nhật của họ?
- Ai phải thực hiện công việc bảo dưỡng, quản trị và giữ cho hệ thống hoạt động?
- Hệ thống sẽ kiểm soát thiết bị phần cứng nào?
- Hệ thống đang xây dựng cần tương tác với những hệ thống khác hay không ?
- Ai hoặc vật thể nào quan tâm đến hay chịu ảnh hưởng bởi kết quả mà hệ thống phần mềm tạo ra?



# What is a user story?

- An abbreviated description of a use case
- Used in [agile development](#)

Answers 3 questions:

1. **Who**?
2. Does **what**?
3. And **why**?

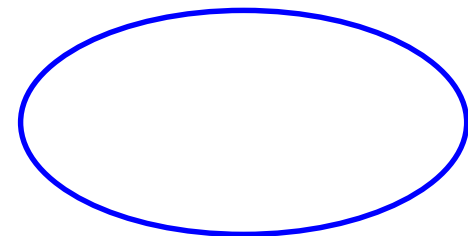
*As a <type of user>,  
I want <some behavior from the system>  
so that <some value is achieved>*





# Khái niệm Use-Case

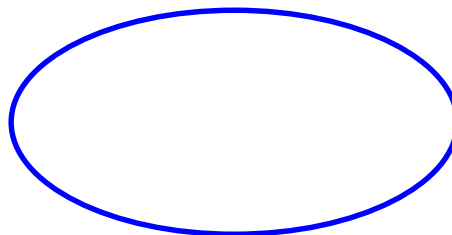
- Một Use-Case là một chuỗi các hành động mà hệ thống thực hiện mang lại một kết quả quan sát được đối với actor.
- Có thể hiểu một Use-Case là một chức năng của hệ thống, mang một ý nghĩa nhất định đối với người dung
- Được biểu diễn bằng hình Ellipse, bao gồm: tên, luồng sự kiện, kịch bản



Use-Case



# Khái niệm Use-Case

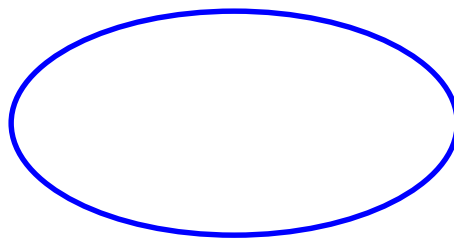


Use-Case

- Mỗi use case có các thuộc tính sau:
  - Action Steps
  - Extension Points
  - Exceptions
  - Pre-Conditions
  - Post-Conditions



# Khái niệm Use-Case



Use-Case

- **Action Steps:**

- Mô tả các bước thông thường tương tác giữa người dùng và hệ thống khi thực hiện Use case này



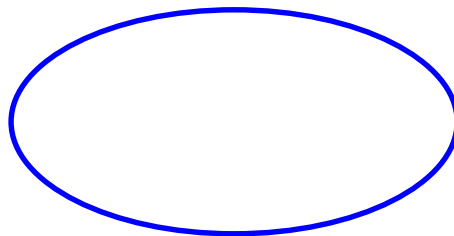
# Khái niệm Use-Case

## •Action Steps:

- Chọn chức năng xem thời khóa biểu
- Hiển thị màn hình cho phép người dùng chọn thông tin cần xem gì
- Chọn niên khóa từ danh sách hiện có trong hệ thống
- Chọn học kỳ từ danh sách hiện có trong hệ thống
- Chọn lớp hoặc tên giảng viên cần xem trong danh sách lớp và giảng viên hiện có trong hệ thống
- Hiển thị thông tin chi tiết thời khóa biểu lớp hoặc thời khóa biểu của giảng viên



# Khái niệm Use-Case



Use-Case

- **Extension Points:**

- Mô tả các trường hợp ngoại lệ khi sử dụng Use case này





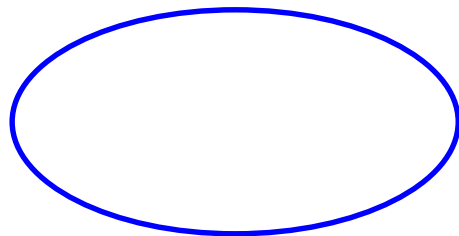
# Khái niệm Use-Case

## •Extension Points:

- **Nếu** người dùng hủy bỏ chức năng xem thời khóa biểu **thì** hệ thống quay trở lại màn hình trước khi người dùng chọn chức năng xem thời khóa biểu
- **Nếu** người dùng chọn xem thời khóa biểu của tất cả các lớp **thì** hiển thị TKB của tất cả các lớp các lớp
- Người dùng có thể chọn xem thời khóa biểu của tất cả các giảng viên
- Người dùng có thể in thời khóa biểu của từng lớp
- Người dùng có thể in thời khóa biểu của từng giảng viên



# Khái niệm Use-Case



Use-Case

- Exceptions
  - Xử lý lỗi xảy ra trong quá trình người dùng sử dụng chức năng này



# Khái niệm Use-Case

## • Exceptions

- Người dùng chọn lớp chưa có thông tin, hệ thống báo lỗi và yêu cầu người dùng chọn lớp khác
- Người dùng chọn giảng viên chưa có thông tin, hệ thống báo lỗi và yêu cầu người dùng chọn giảng viên khác
- Trình duyệt không hiển thị được do lỗi javascript, hệ thống báo lỗi và yêu cầu người dùng chọn trình duyệt khác (ví dụ Firefox, Mozilla,..)



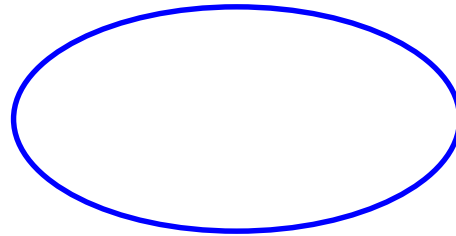
# Khái niệm Use-Case

## • Exceptions

- Trình duyệt không hiển thị được do lỗi đường truyền mạng (Request timeout) hệ thống yêu cầu người dùng refresh lại chức năng này
- Trình duyệt không hiển thị được do lỗi kết nối CSDL (Connection) hệ thống yêu cầu người dùng refresh lại chức năng này
- ...



# Khái niệm Use-Case

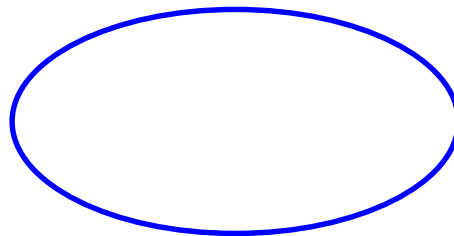


Use-Case

- Pre-Conditions
  - Điều kiện cần để thực hiện hành động



# Khái niệm Use-Case



Use-Case

- Post-Conditions
  - Điều kiện cần để kết thúc hành động



# Ví dụ

## Xét phần mềm Quản lý học sinh cấp III

STT	Yêu cầu
1	Tiếp nhận học sinh
2	Lập danh sách lớp
3	Tra cứu học sinh
4	Nhận bảng điểm môn
5	Xem báo cáo tổng kết
6	Thay đổi quy định

Bao gồm cả tính năng  
Thêm mới, Xóa, và Sửa

❖ **Có bao nhiêu Use-case trong ví dụ này?**



# Ví dụ

## Xét phần mềm Quản lý học sinh cấp III

STT	Yêu cầu
1	Tiếp nhận học sinh
2	Lập danh sách lớp
3	Tra cứu học sinh
4	Nhận bảng điểm môn
5	Xem báo cáo tổng kết
6	Thay đổi quy định

Bao gồm cả tính năng  
Thêm mới, Xóa, và Sửa

❖ Có bao nhiêu Use-case trong ví dụ này?





# Ví dụ

## Xét phần mềm Quản lý thư viện

STT	Yêu cầu
1	Lập thẻ đọc giả
2	Nhận sách mới
3	Tra cứu học sinh
4	Lập phiếu mượn
5	Nhận trả sách
6	Thay đổi quy định

❖ **Có bao nhiêu Use-case trong ví dụ này?**



## Phần mềm thí nghiệm mạch điện

STT	Yêu cầu
1	Sắp đặt mạch điện
2	Cung cấp nguồn điện
3	Thay đổi thông số
4	Lưu bài thí nghiệm
5	Lấy lại thí nghiệm
6	Thay đổi quy định

❖ **Có bao nhiêu Use-case trong ví dụ này?**



# Tìm kiếm Use Case

Trả lời một số câu hỏi như:

- Actor yêu cầu chức năng gì của hệ thống?
- Actor cần phải đọc, tạo, xoá, sửa đổi hoặc lưu trữ thông tin nào đó của hệ thống không?
- Actor cần thiết phải được cảnh báo về những sự kiện trong hệ thống, hay actor cần phải báo hiệu cho hệ thống về vấn đề nào đó không?
- Hệ thống có thể hỗ trợ một số công việc thường nhật của actor nào đó hay không?



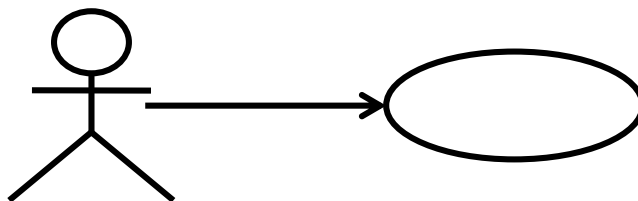
# Tìm kiếm Use Case (tt)

Một số câu hỏi khác cần chú ý:

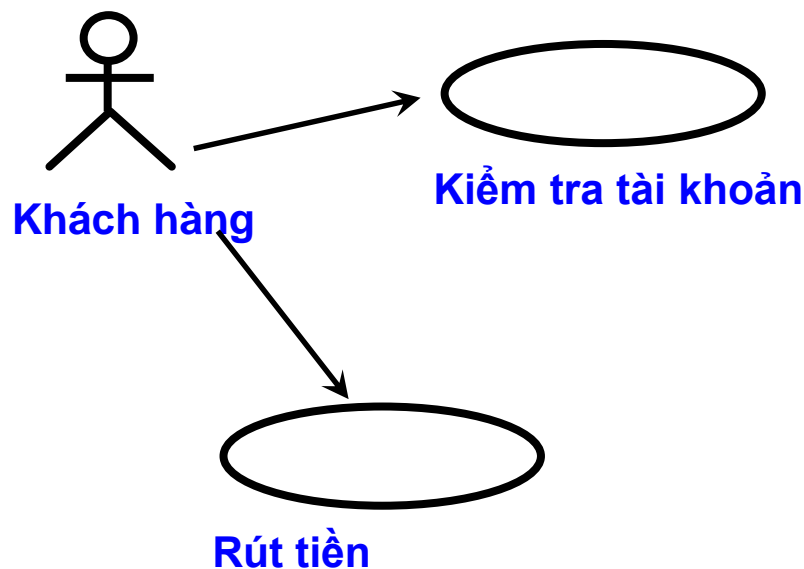
- Hệ thống cần dữ liệu input/output nào? Dữ liệu đó đến từ đâu?
- Những khó khăn nào liên quan đến hiện thực của hệ thống hiện tại (chẳng hạn hệ thống quản lý bằng giấy tờ nên được thay thế bằng hệ thống quản lý trên máy tính)?



# Sơ đồ Use-case



Sự tương tác giữa Actor và Use-case   
Chiều của mũi tên thể hiện vai trò chủ động trong sự tương tác



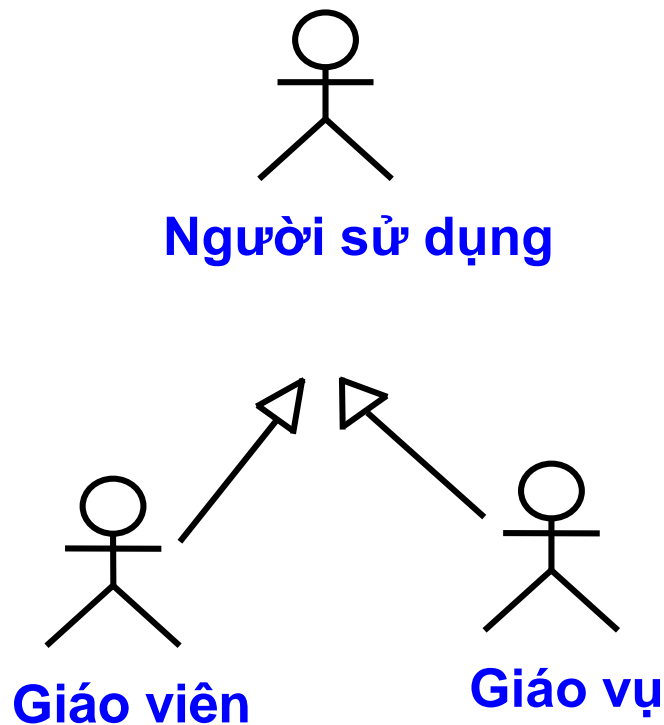


# Các loại mối quan hệ

- Phụ thuộc
- Tổng quát hóa
- Kết hợp



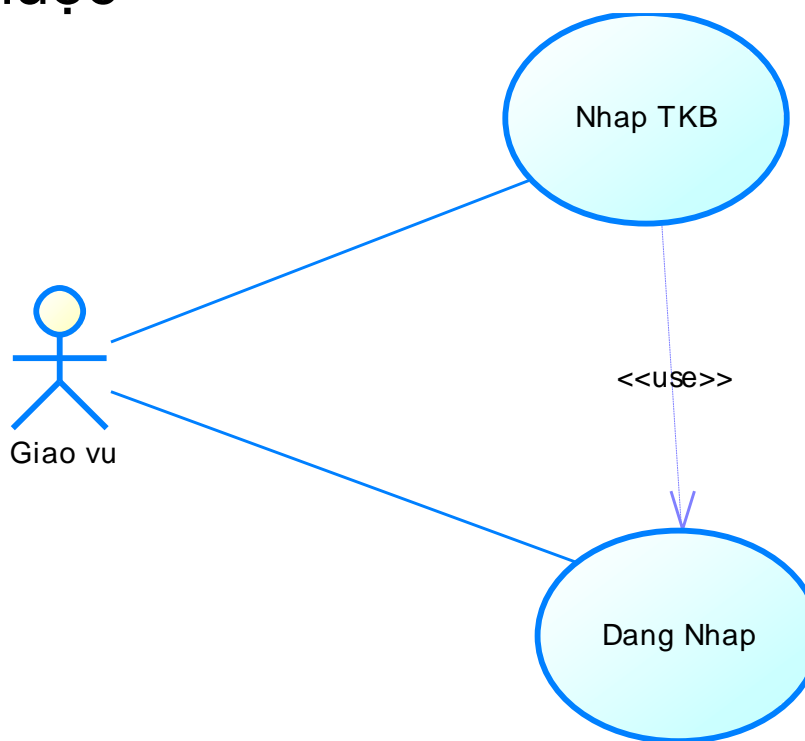
# Tổng quát hóa giữa các Actor





# Quan hệ giữa các Use Case

- Use case – Use case
  - Dependency: Phụ thuộc

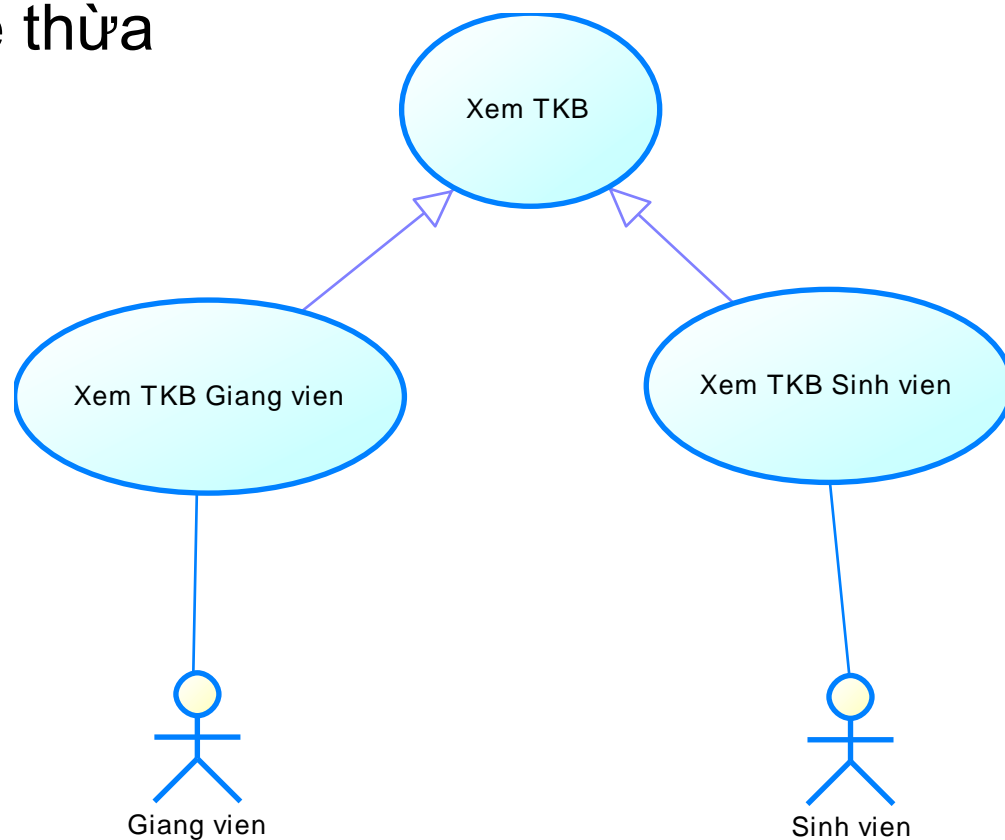






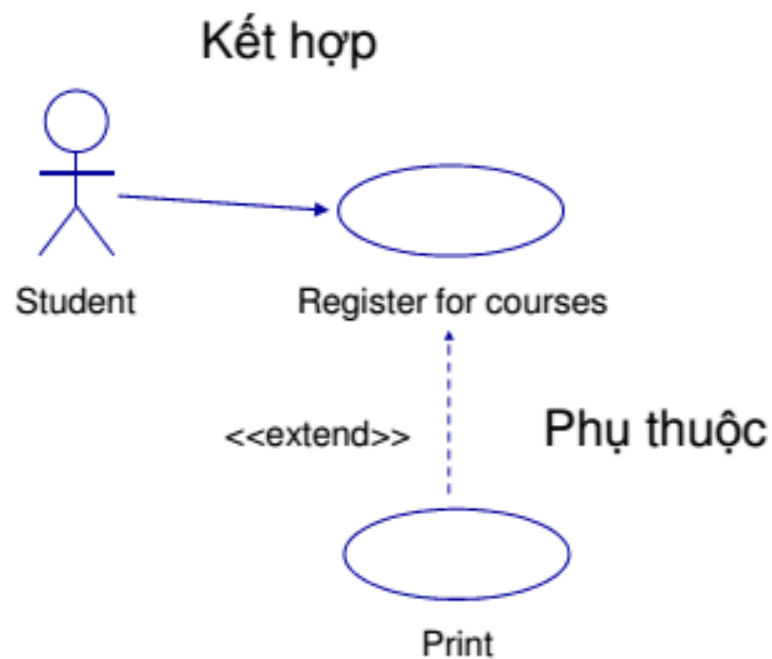
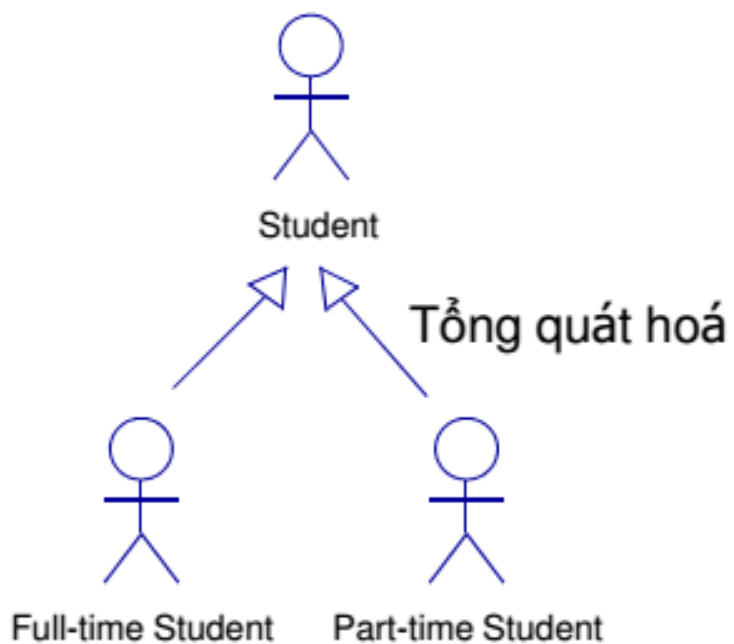
# Quan hệ giữa các Use Case

- Use case – Use case
  - Generalization: Kế thừa



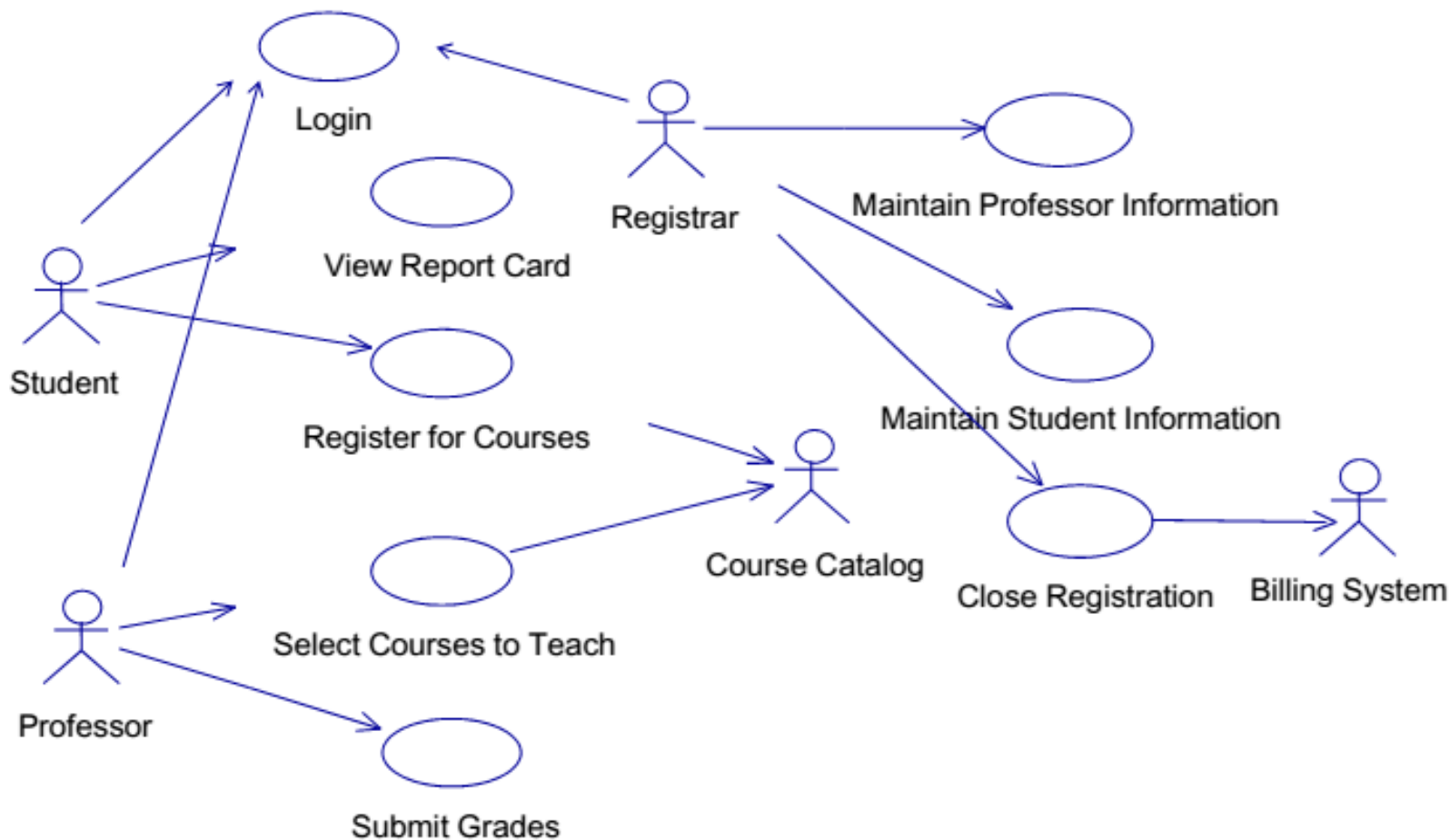


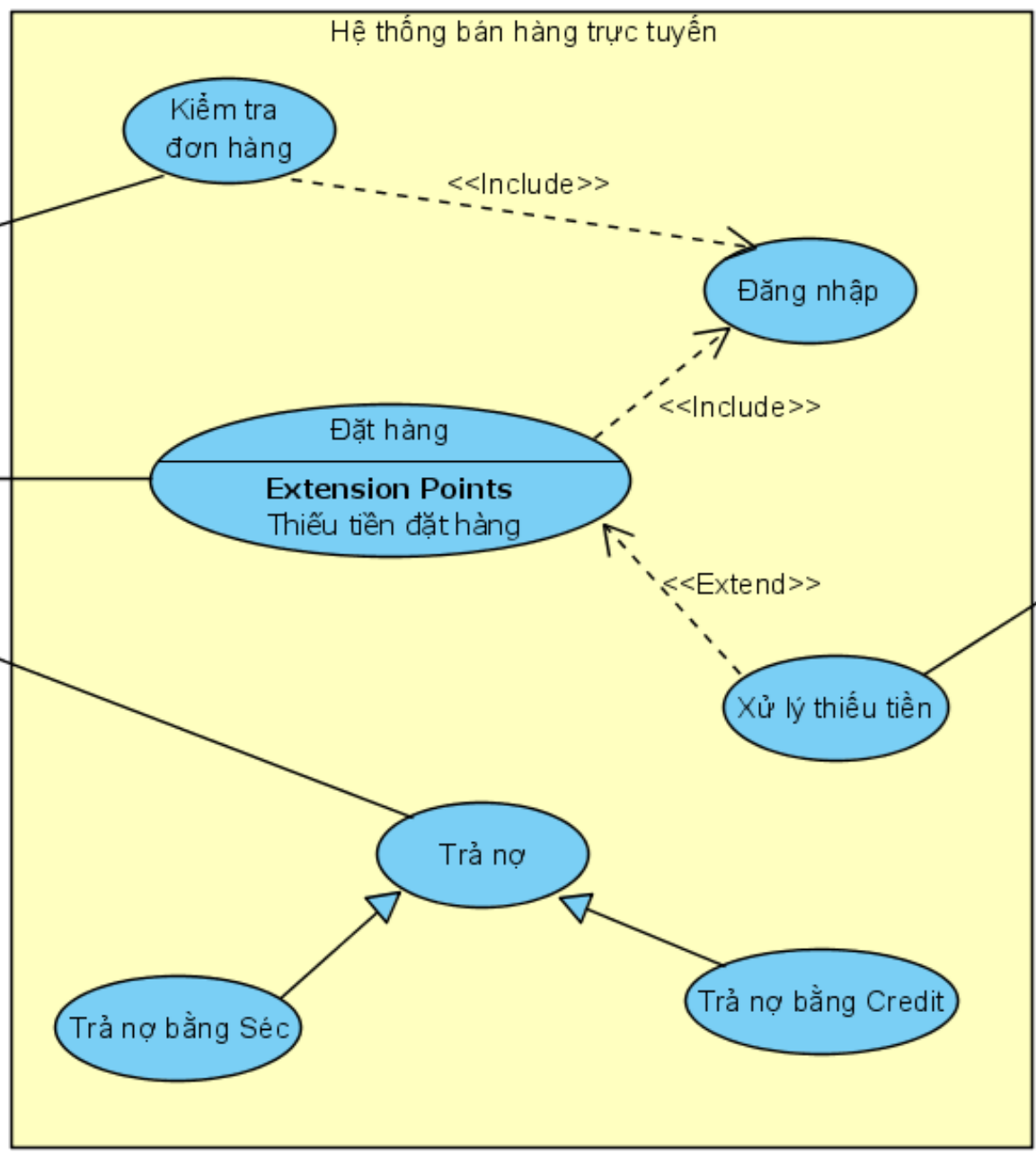
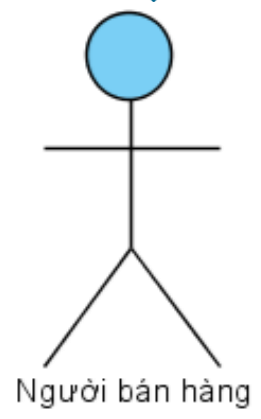
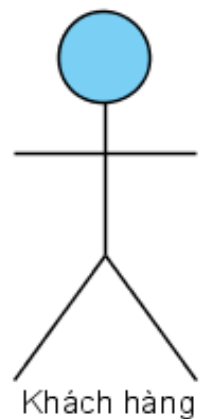
# Ví dụ





# Ví dụ









# Chú ý:

- Khi nào thì vẽ quan hệ **<include>** (bao hàm)

→ Use case A được gọi là include B nếu trong xử lý của A có gọi đến B ít nhất 1 lần !

- Minh họa thông qua Code

```
Class B { public void X () { .... } }
```

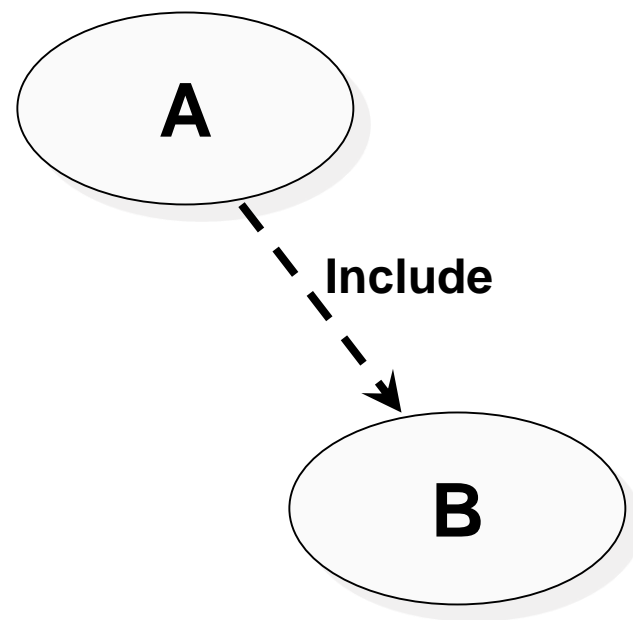
```
Class A {
```

```
    Pubic void Y () {
```

```
        B objB = new B(); objB.X (); ...
```

```
    }
```

```
}
```





## Chú ý:

- Khi nào thì vẽ quan hệ <extend> (mở rộng)  
 → Use case B được gọi là **extend** A nếu use case B được gọi bởi A nếu thỏa mãn điều kiện nào đó.

- Minh họa thông qua Code

```
Class B { public void InẤn () { .... } }
```

```
Class A {
```

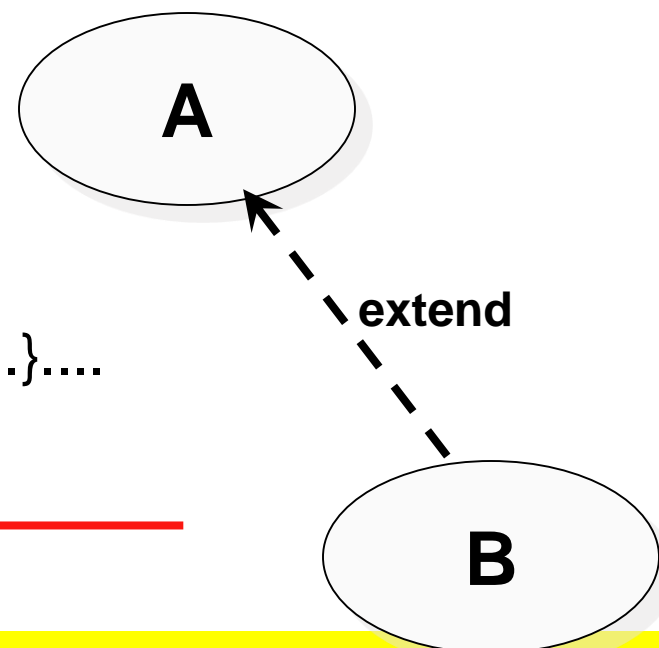
```
    public void XemDSSV () {
```

```
        ... If (Click_Nút_InẤn)
```

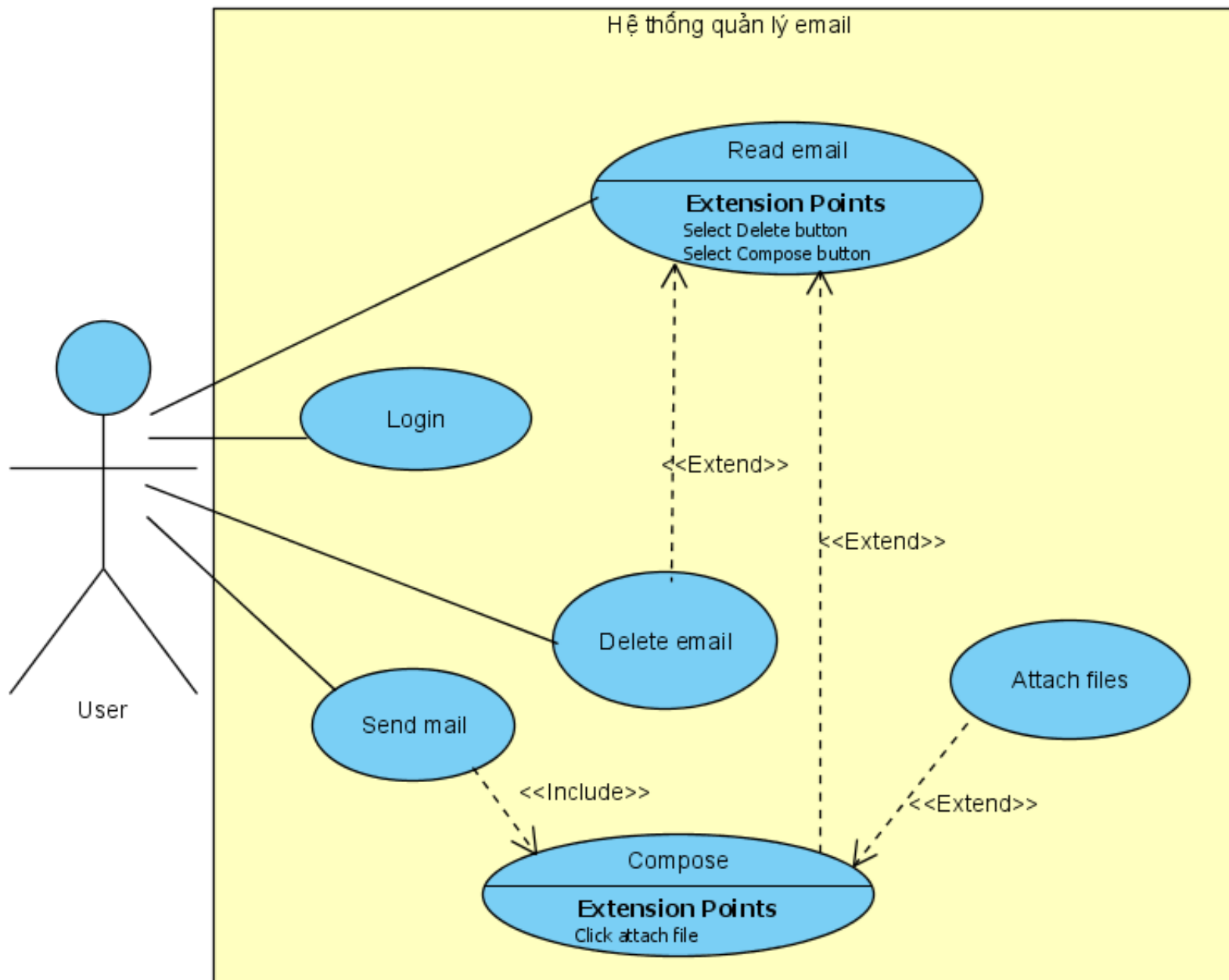
```
        { B objB = new B(); objB.InẤn(); ...}....
```

```
    }
```

```
}
```

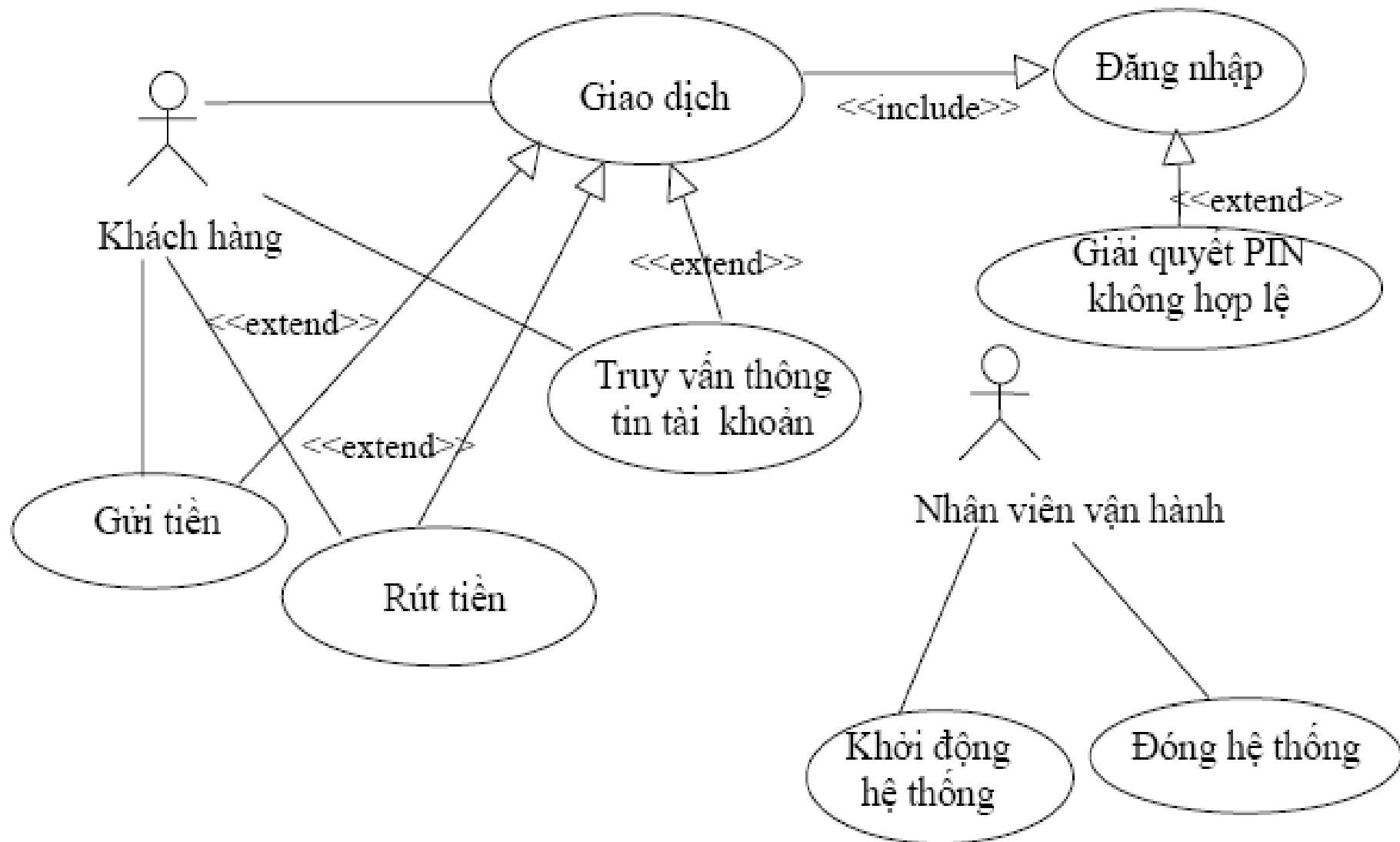


# HIENLTH Một số hình vẽ đúng



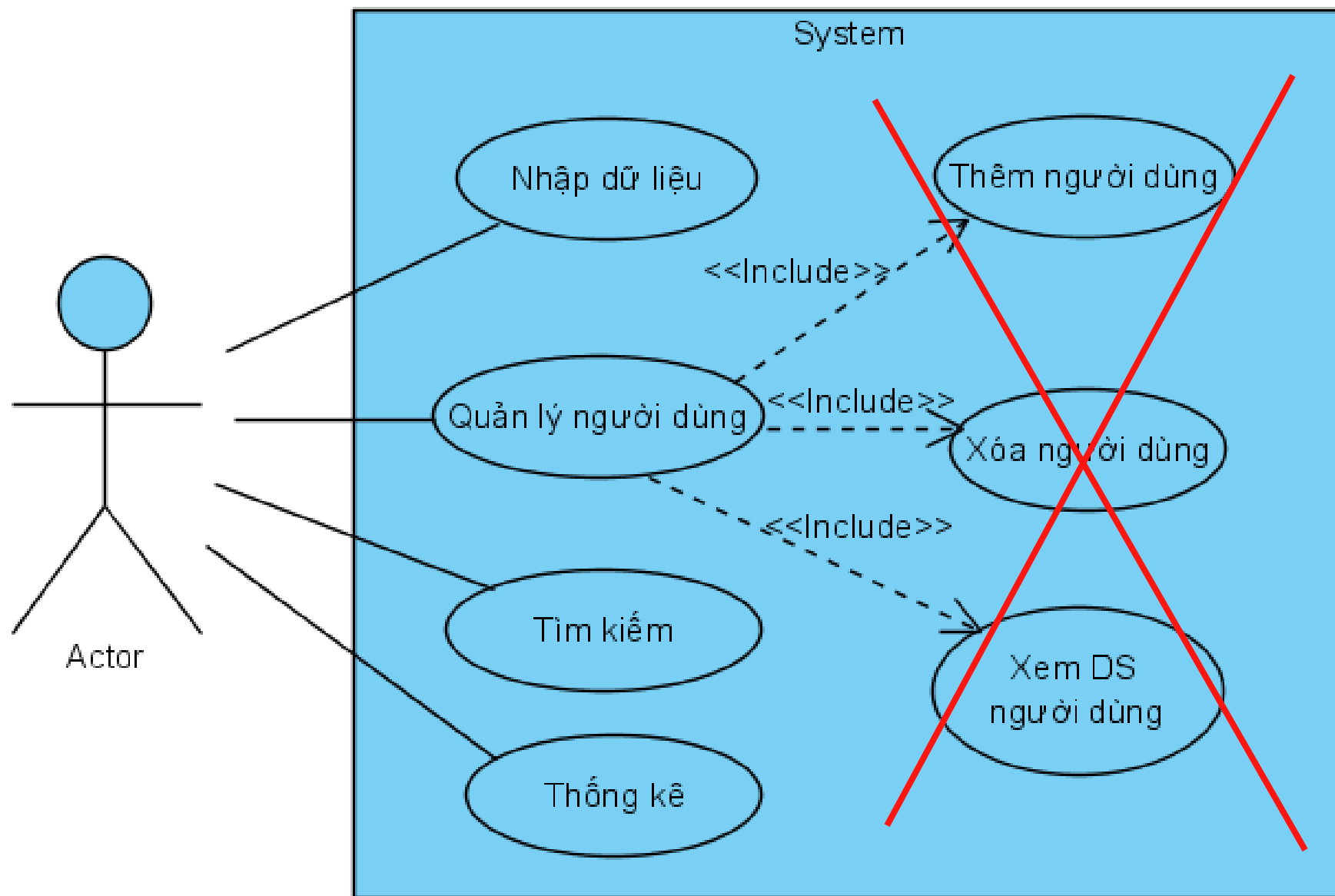


# Một số hình vẽ đúng



Mô hình use case của hệ thống máy ATM

# Một số hình vẽ sai





# Vẽ quan hệ tổng quát hóa (thừa kế)

- ❖ Khi nào thì vẽ quan hệ <Generalization> (tổng quát hóa)  
→ Use case A được gọi là Generalization B nếu B là một trường hợp riêng của A!

- ❖ Nếu A Generalization B thì code có dạng như thế nào

```
Class A {
```

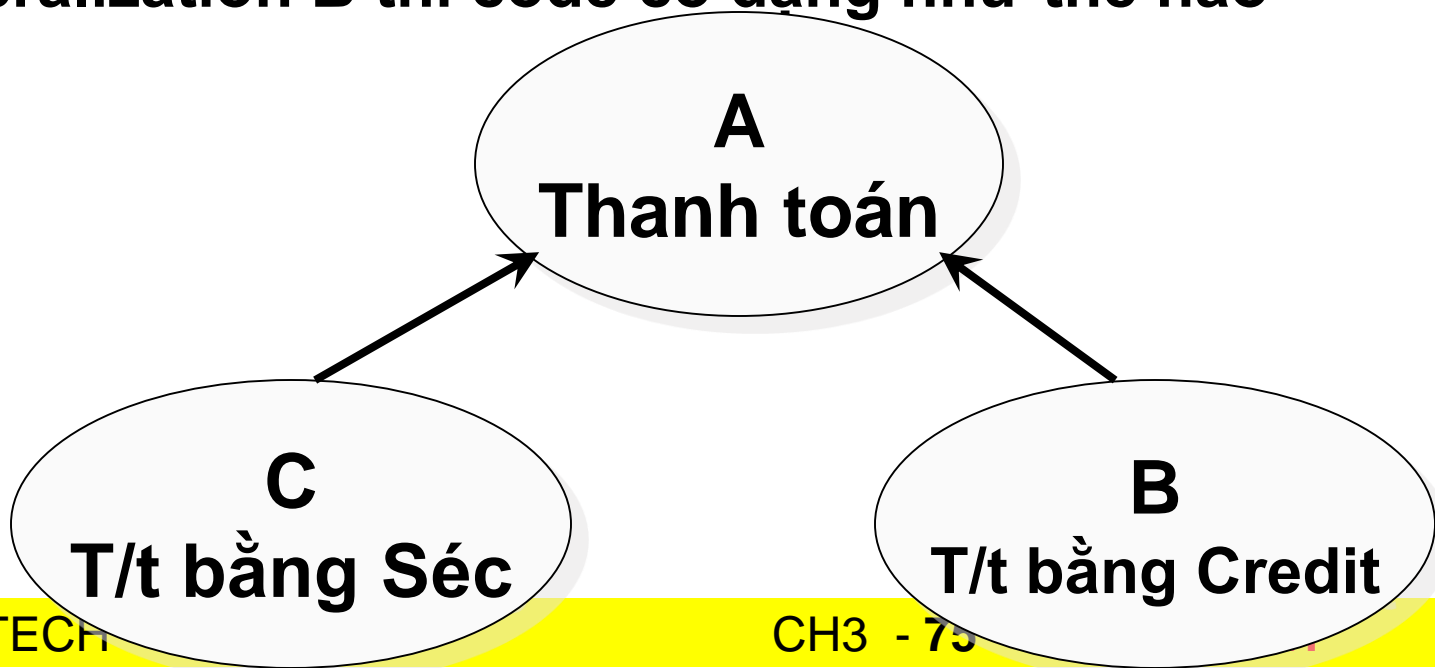
```
.....
```

```
}
```

```
Class B : A
```

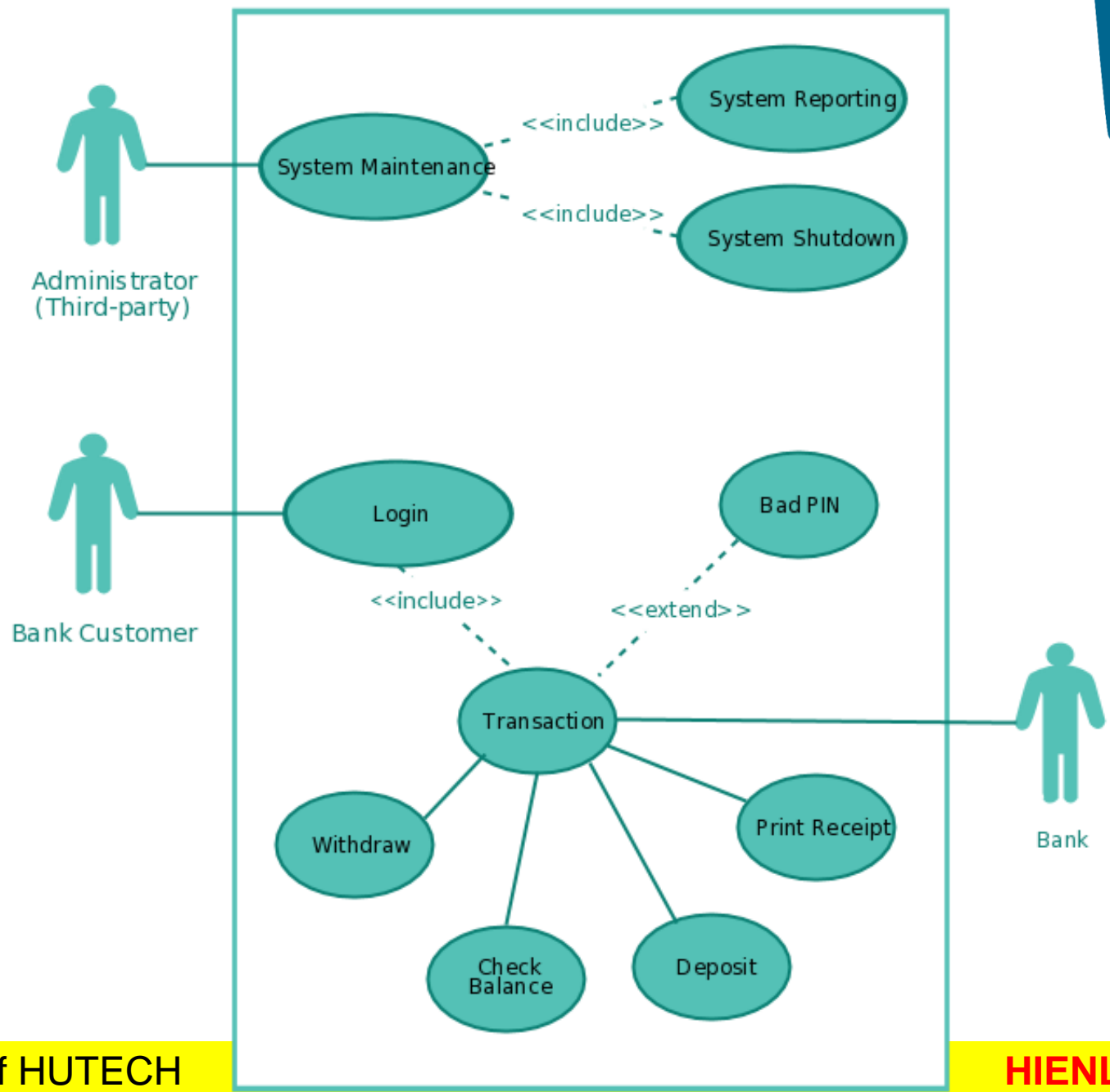
```
{
```

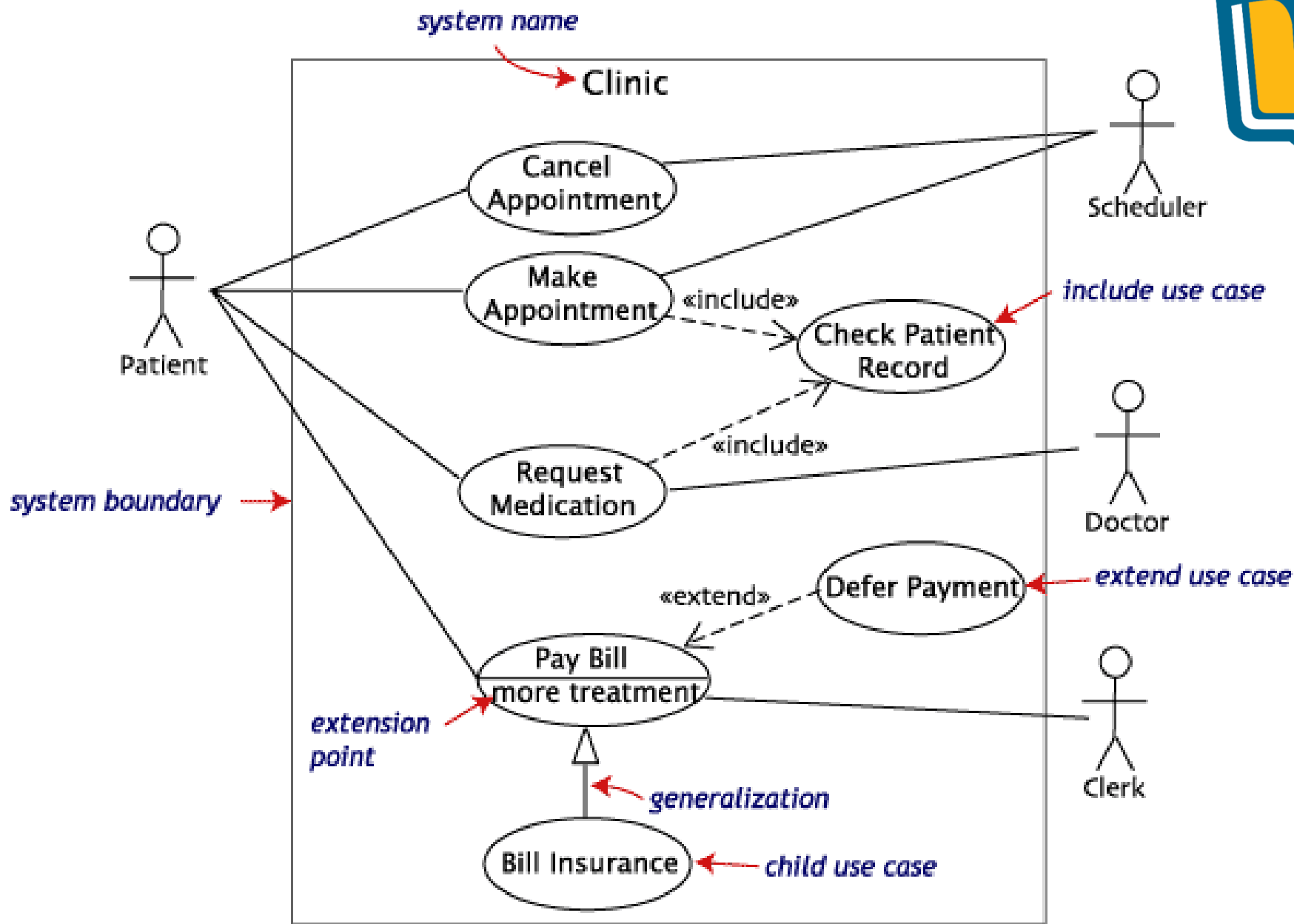
```
.....
```





## Simple ATM Machine System







# References

Bài giảng này có tham khảo:

- Slide bài giảng “Xây dựng phần mềm hướng đối tượng”, **Trần Minh Triết**, ĐH KHTN TpHCM.
- Slide bài giảng “Phân tích Thiết kế HTTT hướng đối tượng”, **Nguyễn Trần Minh Thu**, ĐH KHTN TpHCM.
- Slide bài giảng Kỹ nghệ phần mềm, ĐH Công nghệ, ĐHQG Hà Nội.
- UML, Nguyễn Văn Quý, APTECH, Hưng Yên.



# Câu hỏi và thảo luận





**Thank you!!!**

