



HUST

ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.



ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

PTTK HỆ THỐNG

CHƯƠNG 1: TỔNG QUAN VỀ PTTK HỆ THỐNG

ONE LOVE. ONE FUTURE.

- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu

Nội dung

- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu

Chương trình máy tính và hệ thống thông tin

- Chương trình máy tính - là một chương trình phần mềm chạy trên một hệ thống máy tính để thực hiện một tập chức năng cụ thể.
 - Phạm vi nhỏ hơn hệ thống thông tin
- Hệ thống thông tin - là một tập các thành phần được kết nối và cộng tác để thu thập, xử lý, lưu trữ, và cung cấp thông tin cần thiết để hoàn thành các hoạt động nghiệp vụ.
 - Phạm vi lớn hơn chương trình máy tính
 - Bao gồm cả cơ sở dữ liệu và các quy trình thủ công liên quan

- Dự án - là một chuỗi hoạt động có thời hạn được lập kế hoạch có bắt đầu, có kết thúc và tạo ra các kết quả xác định
 - Có thể được sử dụng để phát triển hệ thống thông tin
 - Các hoạt động cần được thực hiện được xác định, lập kế hoạch, tổ chức và giám sát.
 - Yêu cầu kiến thức về phân tích và thiết kế hệ thống, các công cụ và các kỹ thuật

- Vì sao chúng ta cần một quy trình hình thức?
 - Nhiều dự án bị chậm tiến độ, vượt kinh phí hoặc được bàn giao với ít tính năng hơn kế hoạch
 - Thất bại xảy ra quá thường xuyên
 - Phát triển hệ thống không đơn giản
- Người phân tích hệ thống có vai trò quan trọng
 - Tạo các đặc tả hệ thống đem lại giá trị
 - Công việc nhiều triển vọng nhưng cũng có nhiều thách thức
 - Phải hiểu các quy trình nghiệp vụ và có tập kỹ năng riêng

Nội dung

- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu



Vòng đời phát triển hệ thống (SDLC)

SDLC bao quát toàn bộ tiến trình phát triển hệ thống.

- Tất cả các hoạt động cần cho các giai đoạn lập kế hoạch, phân tích, thiết kế, lập trình, kiểm thử, đào tạo người dùng, v.v.. trong tiến trình phát triển hệ thống
- Tất cả các hoạt động quản lý dự án cần thiết để triển khai thành công dự án.

SDLC từng bước cho thấy cách một hệ thống thông tin được hình thành bắt đầu từ ý tưởng sơ khai cho tới khi được đưa vào ứng dụng, đáp ứng các nhu cầu nghiệp vụ, từ các đặc tả tới sản phẩm thực tế.

Có nhiều cách thiết lập SDLC, tuy nhiên các mô hình SDLC đều bao gồm nhiều pha và thường có thể thiết lập ánh xạ giữa các pha của các mô hình SDLC khác nhau.

- Các pha của 1 mô hình SDLC điển hình:
 - Lập kế hoạch;
 - Phân tích;
 - Thiết kế;
 - Thực thi.
- Mỗi pha bao gồm 1 chuỗi các bước
- Mỗi pha đều tạo ra sản phẩm (có thể chuyển giao)
- Các pha có thể được thực hiện tuần tự, tăng dần, lặp hoặc theo 1 số quy luật khác.

Tập câu hỏi cần được làm rõ trong các pha cơ bản

- Pha lập kế hoạch
 - Vì sao chúng ta nên xây dựng hệ thống này?
 - Nó sẽ đem lại lợi ích gì?
 - Xây dựng nó như thế nào? và ai sẽ xây dựng nó?
 - Cần bao nhiêu thời gian và kinh phí để xây dựng nó?
- Pha phân tích
 - Ai sẽ sử dụng nó? ở đâu? và trong điều kiện nào?
 - Hệ thống cần làm những gì?
- Pha thiết kế
 - Hệ thống hoạt động như thế nào?
 - Hệ thống được xây dựng như thế nào?

Một số hoạt động tiêu biểu

Pha lập kế hoạch

1. Khởi tạo dự án

- Tạo/tiếp nhận các yêu cầu hệ thống
- Thực hiện phân tích tính khả thi

2. Tổ chức quản lý dự án

- Lập kế hoạch làm việc
- Bố trí nhân sự dự án
- Lên kế hoạch theo dõi và điều hành dự án

Một số hoạt động tiêu biểu₍₂₎

Pha phân tích

1. Lập chiến lược phân tích

- Mô hình hóa hệ thống đang có
- Định hình hệ thống mới

2. Thu thập thông tin

- Mô tả ý tưởng hệ thống
- Tạo các mô hình

3. Tạo một đề xuất hệ thống

Một số hoạt động tiêu biểu₍₃₎

Pha thiết kế

1. Xây dựng chiến lược thiết kế
2. Thiết kế kiến trúc và giao diện
3. Thiết kế lưu trữ cố định
4. Thiết kế bố cục hệ thống:
 - Cần viết những chương trình nào?
 - Mỗi chương trình sẽ làm gì?

Một số hoạt động tiêu biểu₍₄₎

Pha thực thi

1. Xây dựng hệ thống

- Lập trình
- Kiểm thử

2. Chuyển giao hệ thống

- Cài đặt
- Đào tạo & hướng dẫn sử dụng

3. Hỗ trợ (bảo trì) hệ thống

Sản phẩm đầu ra tiêu biểu cho các pha

Pha	Sản phẩm tiêu biểu
Lập kế hoạch ⇒	Kế hoạch dự án
Phân tích ⇒	Bản đề xuất hệ thống
Thiết kế ⇒	Tài liệu đặc tả hệ thống
Thực thi ⇒	Hệ thống

Nội dung

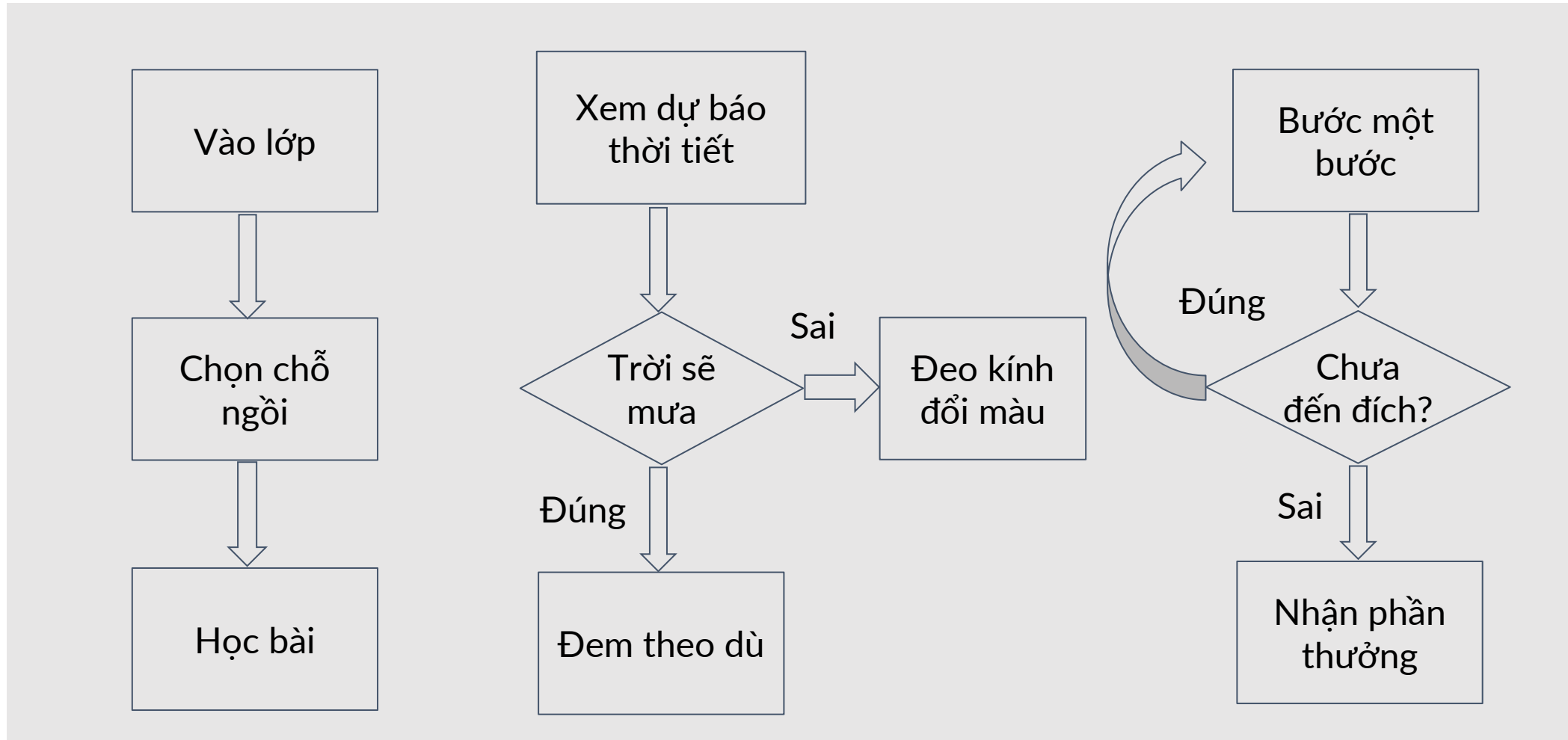
- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu



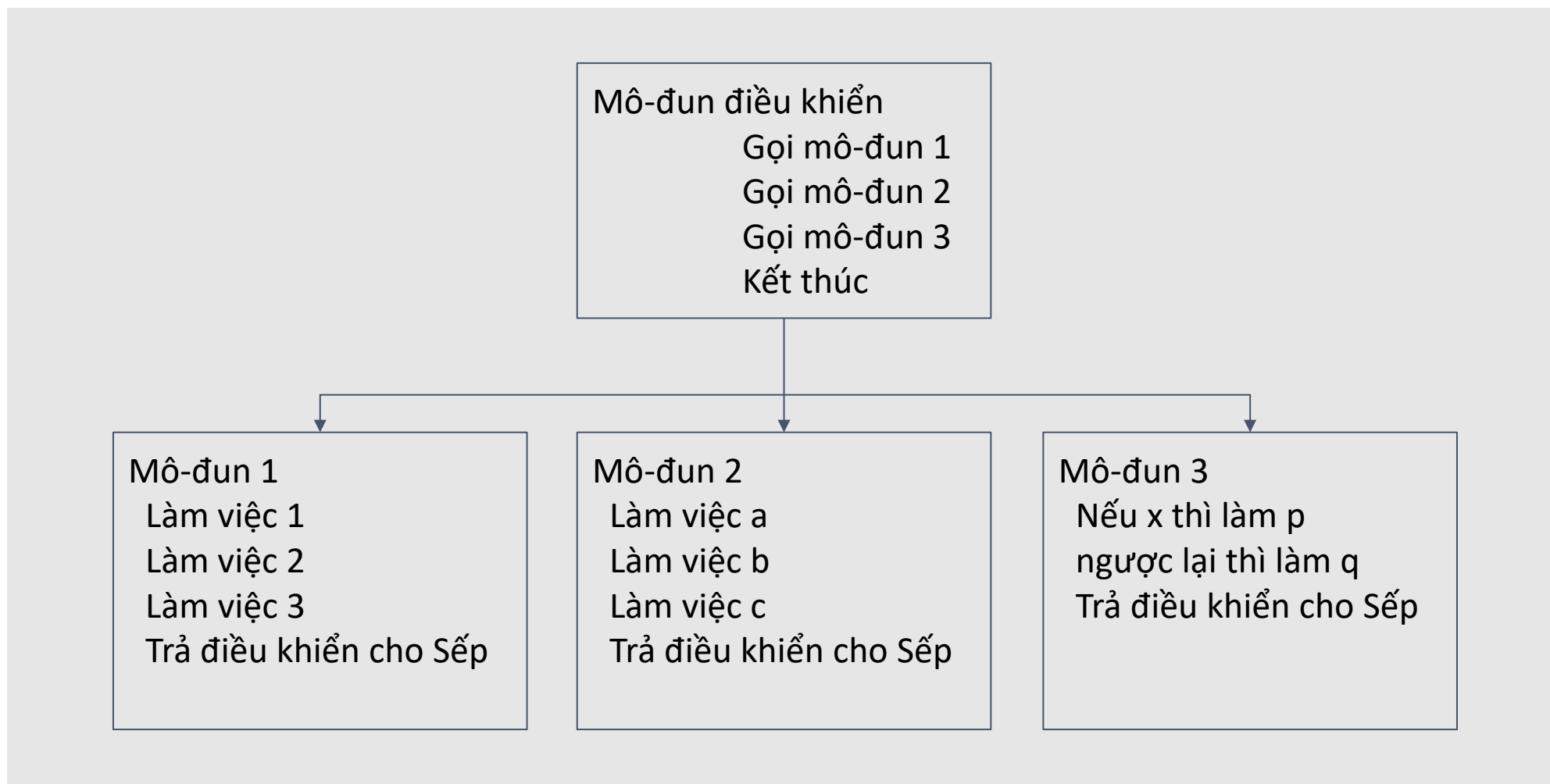
Hai cách tiếp cận phát triển hệ thống

- Cách tiếp cận hướng cấu trúc
 - Cách tiếp cận ra đời sớm hơn. Biểu diễn hệ thống như một tập tiến trình tương tác và xử lý dữ liệu
 - Tập trung vào tiến trình.
 - Phân tích hướng cấu trúc, thiết kế hướng cấu trúc, và lập trình hướng cấu trúc
- Cách tiếp cận hướng đối tượng
 - Cách tiếp cận mới hơn. Biểu diễn hệ thống như một tập đối tượng tương tác với nhau để hoàn thành các nhiệm vụ.
 - Nỗ lực cân bằng sự tập trung vào tiến trình và dữ liệu
 - Phân tích hướng đối tượng, thiết kế hướng đối tượng, và lập trình hướng đối tượng

Cách tiếp cận hướng cấu trúc

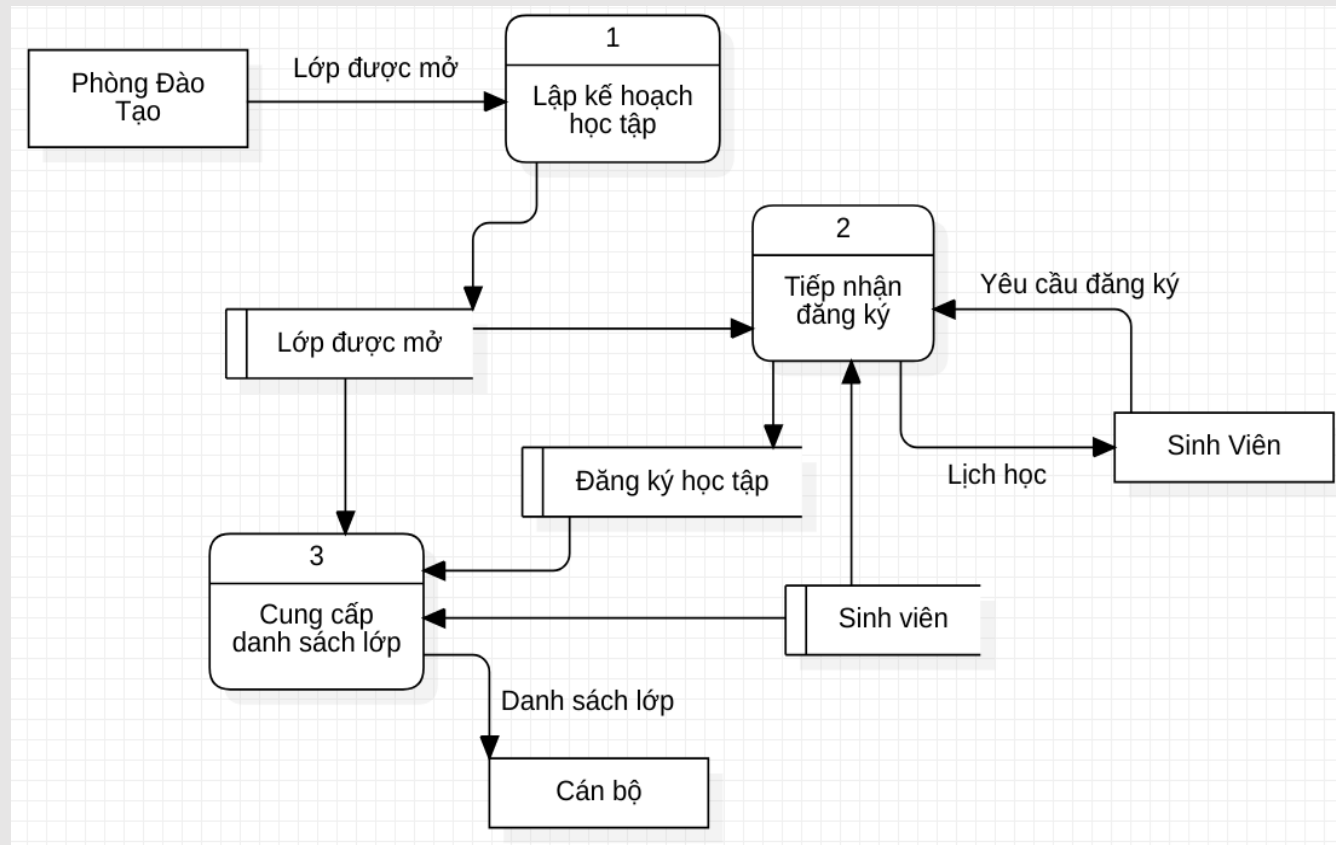


Cách tiếp cận hướng cấu trúc₍₂₎



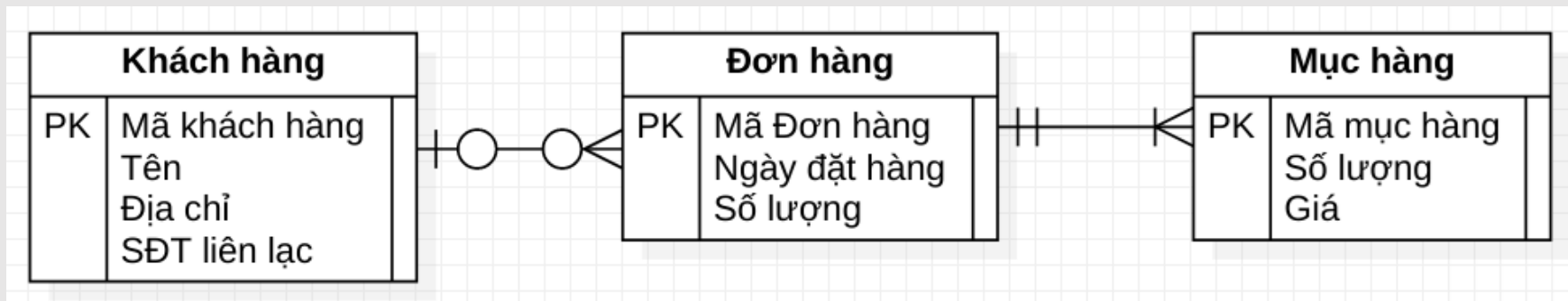
Cách tiếp cận hướng cấu trúc₍₃₎

- Phân tích hướng cấu trúc
 - Sơ đồ luồng dữ liệu (DFD)



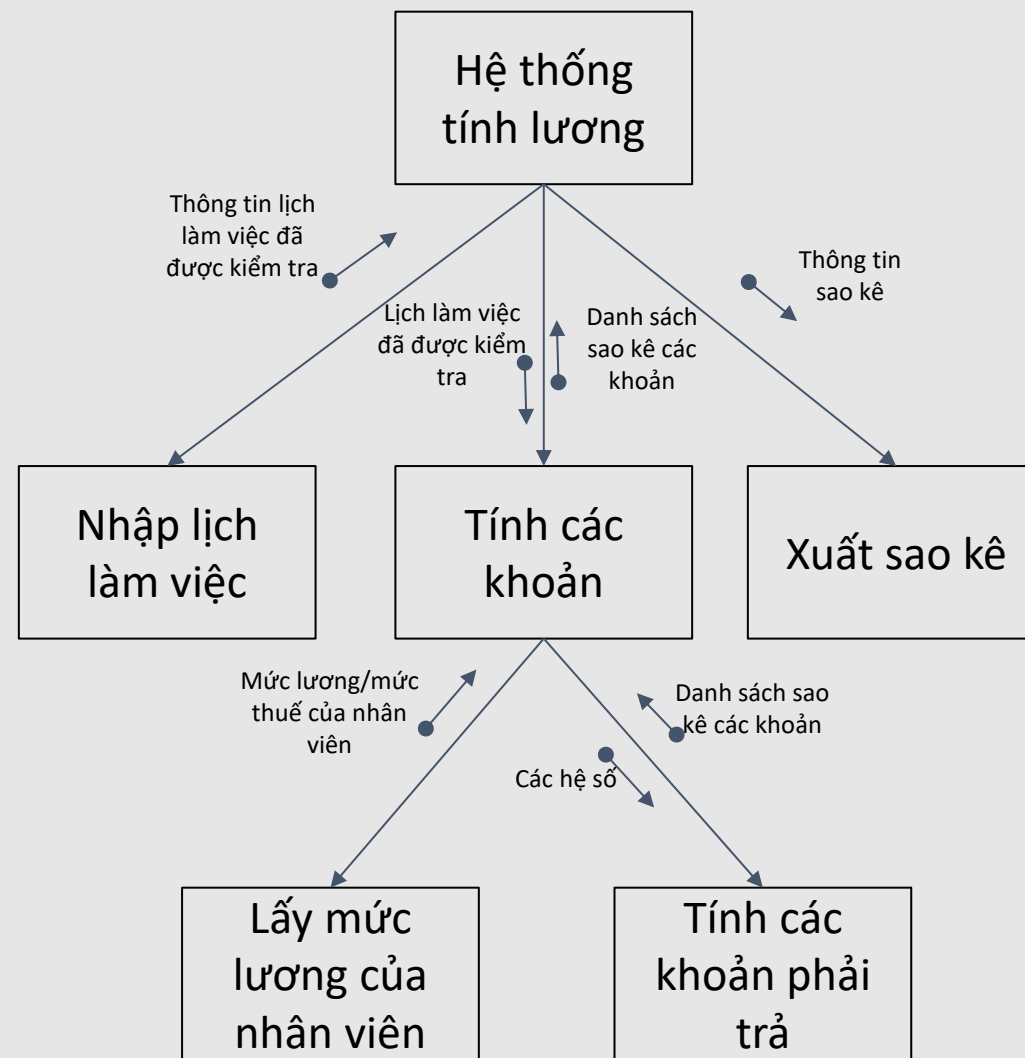
Cách tiếp cận hướng cấu trúc₍₄₎

- Mô hình hóa dữ liệu với sơ đồ thực thể-liên kết



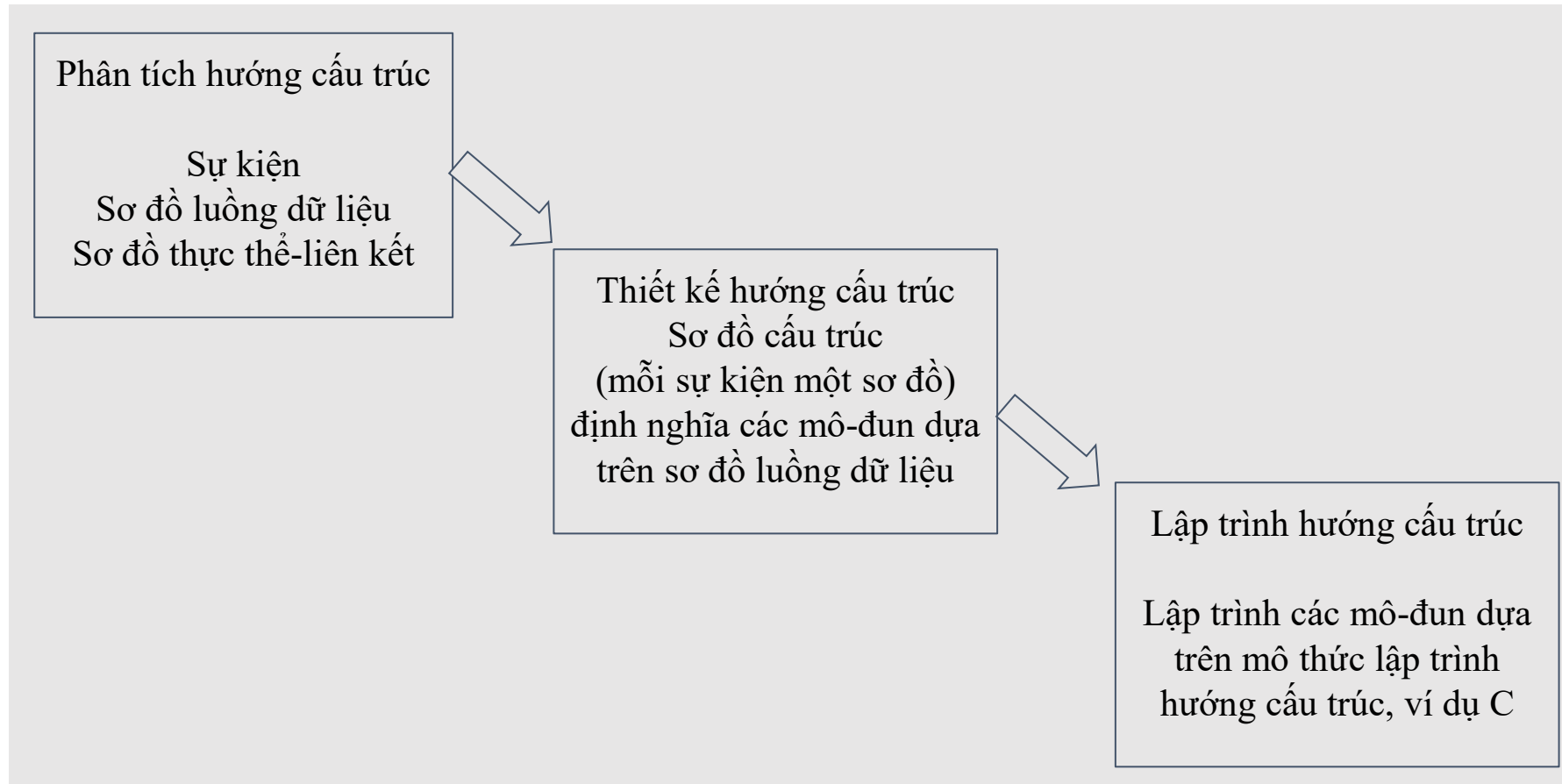
Cách tiếp cận hướng cấu trúc₍₅₎

- Thiết kế hướng cấu trúc
 - Biểu diễn các thành phần cùng với dữ liệu liên quan.



Cách tiếp cận hướng cấu trúc₍₆₎

- Kết hợp các thành phần

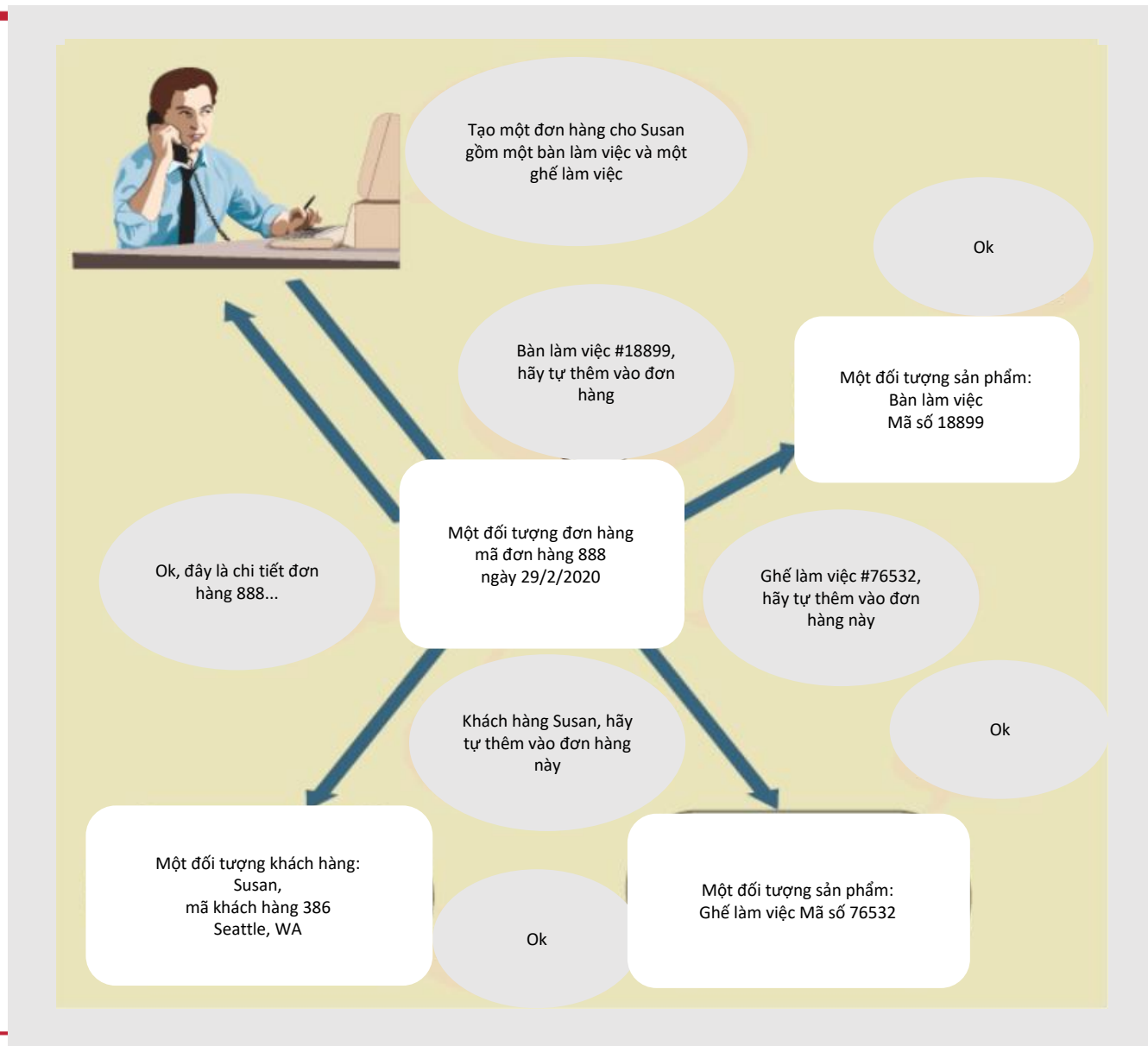


Cách tiếp cận hướng đối tượng

- Phân tích hướng đối tượng (OOA)
 - Xác định và đặc tả ca sử dụng
 - Xác định các đối tượng, các tập đối tượng, lớp
- Thiết kế hướng đối tượng (OOD)
 - Thiết kế lớp,
 - Thiết kế tương tác
- Lập trình hướng đối tượng (OOP)
 - C++, Java
 - Đóng gói và ẩn dữ liệu, kế thừa, đa hình

Cách tiếp cận hướng đối tượng₍₂₎

- Gần với tư duy hợp tác và phân chia công việc
- Các đối tượng tương tác để hoàn thành công việc

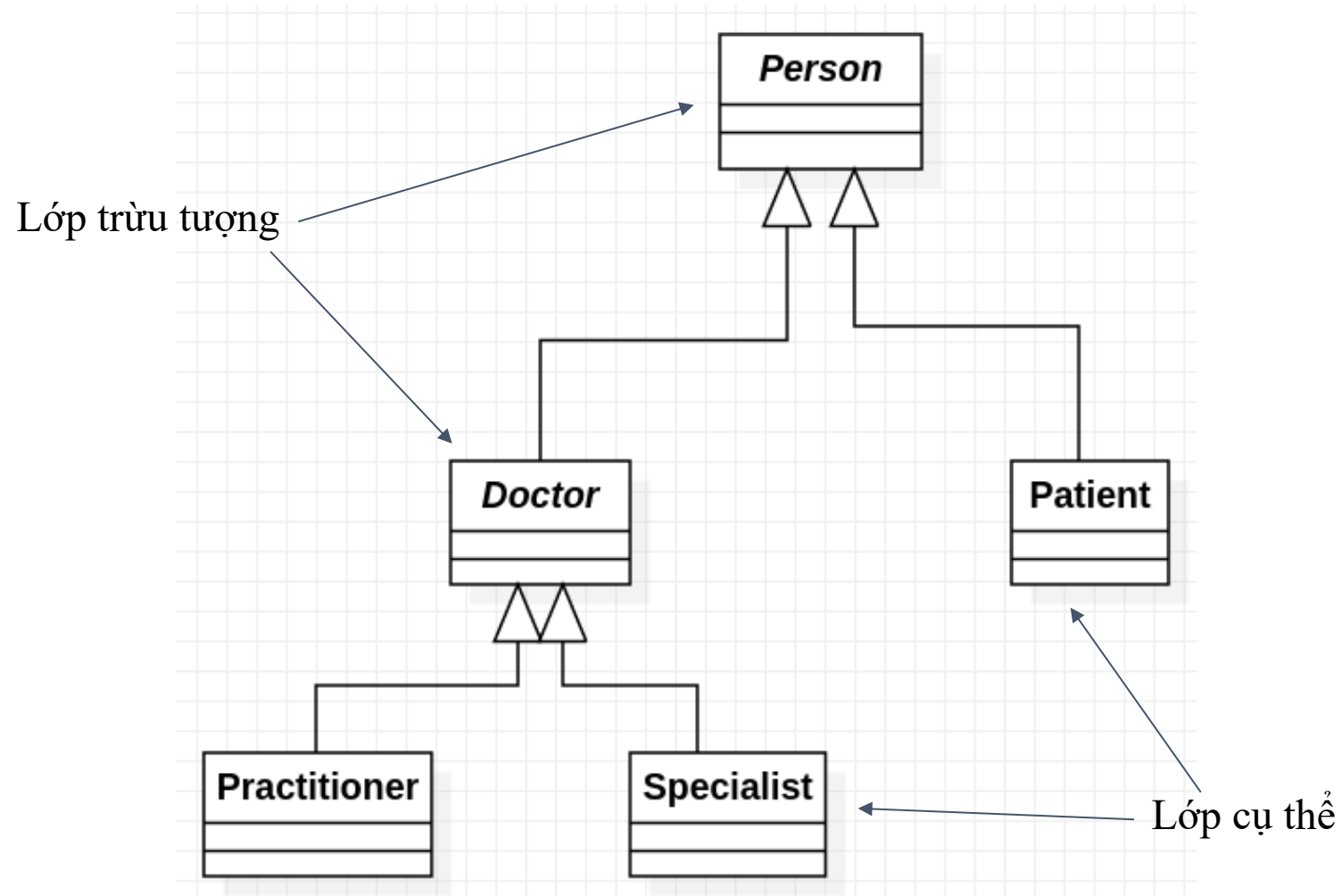


Một số phần tử và tính năng cơ bản

- Đối tượng và lớp
- Thông điệp và phương thức
- Đóng gói và ẩn dữ liệu
- Kế thừa
- Đa hình và phân giải động

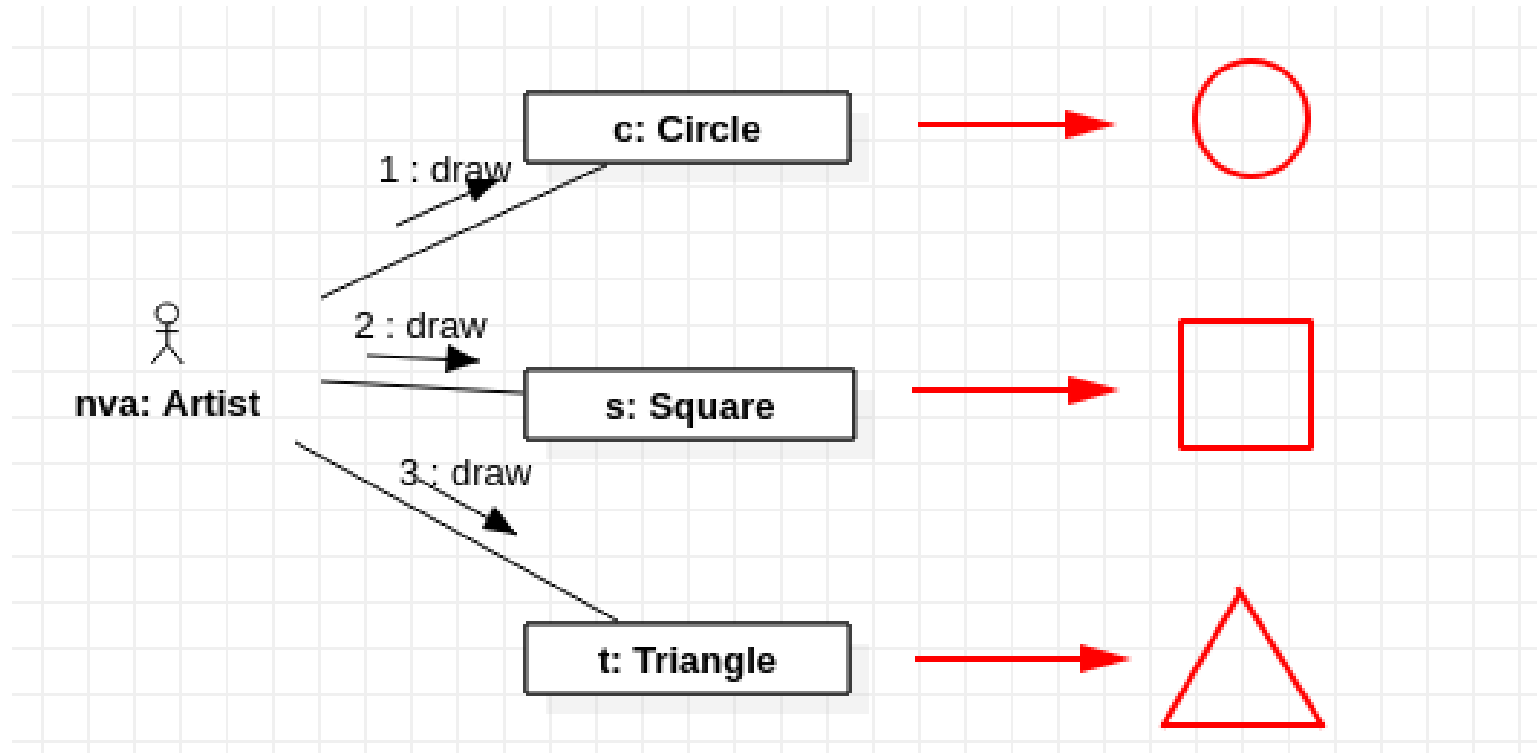
Một số phần tử và tính năng cơ bản⁽²⁾

- Kế thừa



Một số phần tử và tính năng cơ bản₍₃₎

- Đa hình



Object-Oriented Systems Analysis and Design (OOSAD)

- Thường sử dụng
 - Ngôn ngữ mô hình hóa hướng đối tượng (UML - Unified Modeling Language)
 - Và một số ngôn ngữ khác nữa ...
 - và tiến trình thống nhất (Unified Process)
- Một số đặc điểm cơ bản OOSAD
 - Dẫn dắt bởi ca sử dụng
 - Kiến trúc khung
 - Lặp và tăng dần

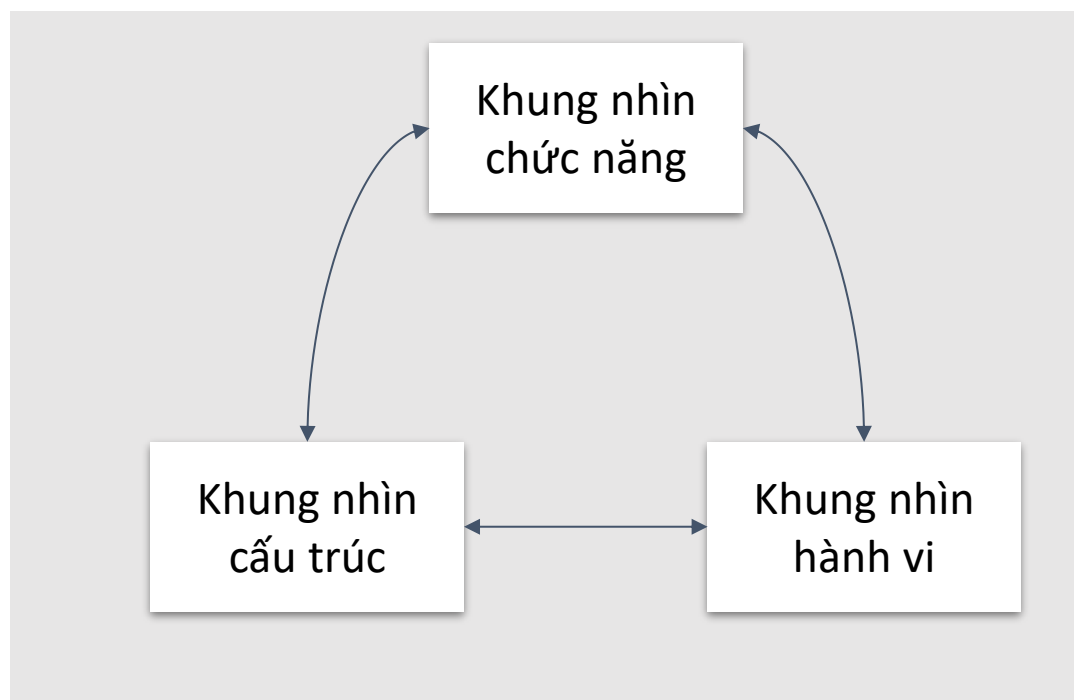
[Grady Booch, Ivar Jacobson, and James Rumbaugh]

PTTK hệ thống theo phương pháp hướng đối tượng₍₂₎

- Dẫn dắt bởi ca sử dụng
 - Các hành vi của hệ thống được đóng gói trong các ca sử dụng
 - Mỗi ca sử dụng tập trung vào 1 quy trình nghiệp vụ
- Kiến trúc khung: Kiến trúc phần mềm được áp dụng cho các đặc tả, xây dựng và tài liệu của hệ thống.
- Cách tiếp cận hướng đối tượng thường hỗ trợ tối thiểu 3 khung nhìn kiến trúc riêng biệt nhưng có quan hệ mật thiết:
 - Khung nhìn chức năng (từ bên ngoài): Tập trung vào góc nhìn người dùng
 - Khung nhìn cấu trúc (tĩnh): Tập trung vào các thuộc tính, phương thức, lớp và các mối quan hệ
 - Khung nhìn hành vi (động): Tập trung vào các thông điệp giữa các lớp và các hành vi phát sinh

PTTK Hệ thống theo phương pháp Hướng Đối Tượng₍₃₎

- Lập và tăng dần
 - Trải qua kiểm thử và điều chỉnh liên tục
 - Sau mỗi bước người phân tích càng hiểu rõ hơn về hệ thống



Các lợi ích của OOSAD

- Chia 1 hệ thống phức tạp thành nhiều phân hệ => dễ quản lý hơn
- Làm việc với từng phân hệ => tập trung hơn
- Sử dụng các vòng lặp => kịp thời thực hiện các thay đổi

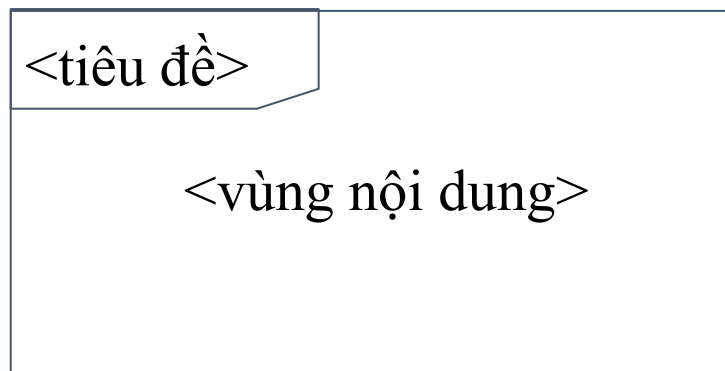
Nội dung

- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu



Ngôn ngữ mô hình hóa hợp nhất

- Unified Modeling Language (UML)
- Ngôn ngữ trực quan, cung cấp nhiều sơ đồ hữu ích cho các hoạt động phân tích & thiết kế hệ thống
 - Nhưng không phải tất cả
 - Cần sử dụng các ngôn ngữ khác (IFML, OCL, v.v..)
- Quy chuẩn OMG (từ 1997) và ISO (từ 2005)
 - omg.org
 - iso.org
 - uml.org
- Có nhiều phiên bản



- Mỗi sơ đồ có thể được đặt trong 1 khung có tiêu đề
 - *(Không bắt buộc phải đặt trong khung)*
- Tiêu đề trong trường hợp sử dụng khung có định dạng:
 - [<phân loại>][<tên>][<tham số>]
 - Có thể mô tả không gian tên của vùng nội dung
 - ... hoặc thành phần của mô hình sở hữu các thành phần được biểu diễn trong vùng nội dung.

Phân loại sơ đồ

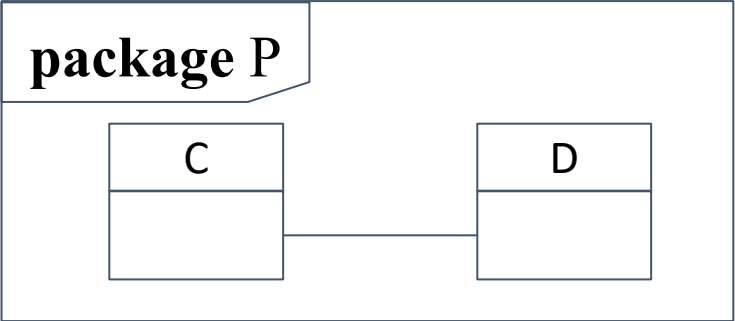
- Các phân loại sơ đồ sau được sử dụng trong tiêu đề khung:

Tên tiếng Anh	Tên rút gọn	Tên tiếng Việt
activity	act	hoạt động
class	--	lớp
component	cmp	thành phần
deployment	dep	triển khai
interaction	sd	tương tác
package	pkg	gói
state machine	stm	máy trạng thái
use case	uc	ca sử dụng

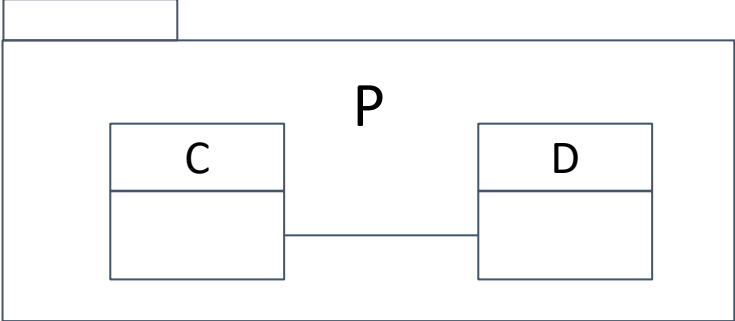
!Lưu ý: Có phần tương đồng nhưng tránh nhầm lẫn với khái niệm loại sơ đồ UML

Ví dụ 1.1: Khung và tiêu đề

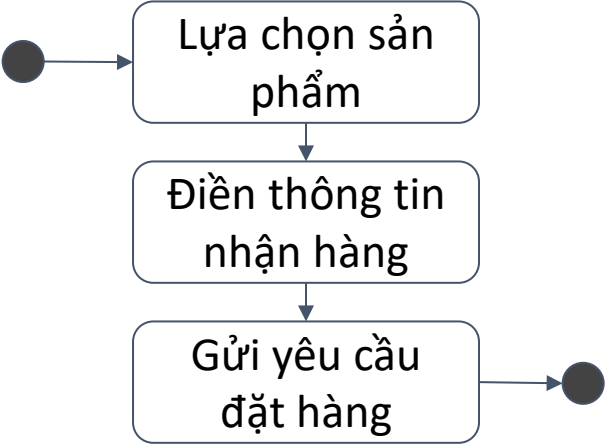
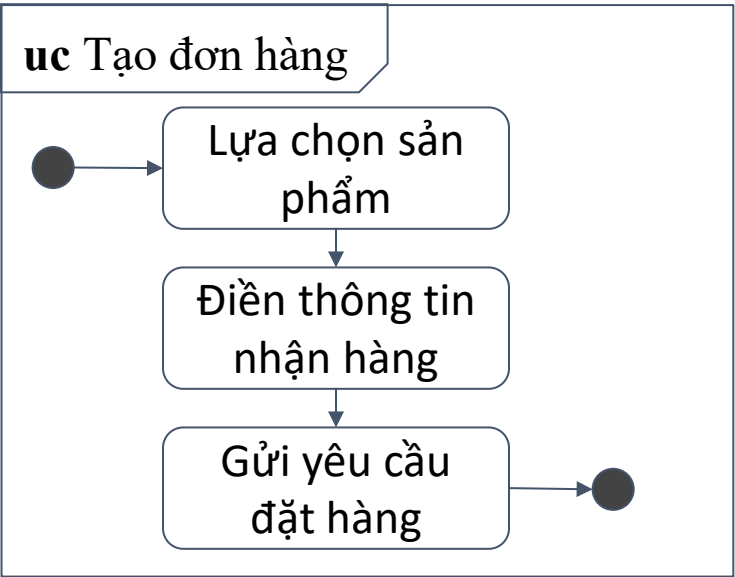
Sử dụng khung



Không sử dụng khung



Sơ đồ lớp với các lớp trong gói P



Sơ đồ hoạt động của 1 kịch bản tương tác trong CSD Tạo đơn hàng

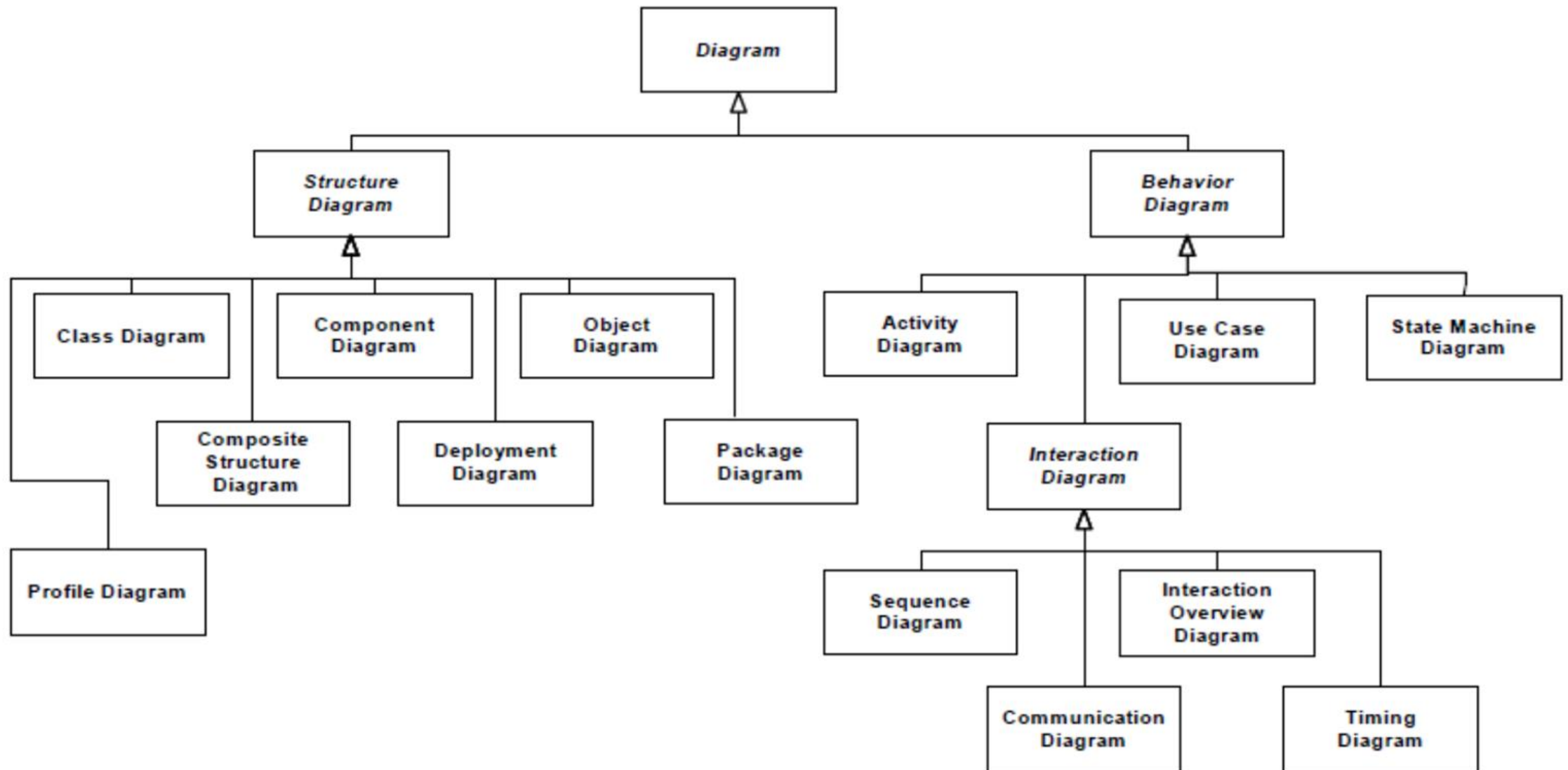
Các sơ đồ UML

Loại sơ đồ	Biểu diễn	Từ phiên bản
Lớp/Class	Lớp, kiểu, giao diện, và các mối quan hệ giữa các thành phần này.	UML 1.x
Đối tượng/Object	Các đối tượng/thực bản của các lớp trong sơ đồ lớp	UML 1.x
Gói/Package	Cấu trúc phân cấp của các nhóm lớp và các thành phần	UML 2.x
Kết cấu/Composite structure	Biểu diễn bên trong của 1 lớp hoặc 1 thành phần, và có thể mô tả các mối quan hệ của lớp trong 1 ngữ cảnh xác định	UML 2.x
Thành phần/Component	Các thành phần quan trọng trong hệ thống và các giao diện được sử dụng để tương tác với các thành phần khác	UML 1.x, được cập nhật trong UML 2.0
Triển khai/Deployment	Cách hệ thống sẽ được triển khai trong môi trường thực tế	UML 1.x
Hồ sơ/Profile	Mở rộng và tùy chỉnh UML với các thành phần và thuộc tính mới	UML 2.x

Các sơ đồ UML₍₂₎

Loại sơ đồ	Biểu diễn	Từ phiên bản
Ca sử dụng/Use Case	Tương tác giữa hệ thống với người dùng và các hệ thống ngoại.	UML 1.x
Hoạt động/Activity	Các hoạt động tuần tự và song song diễn ra trong hệ thống	UML 1.x
Máy trạng thái/State Machine	Trạng thái của đối tượng trong suốt thời gian tồn tại và các sự kiện làm thay đổi trạng thái của đối tượng	UML 1.x
Tuần tự/Sequence	Tương tác giữa các đối tượng trong trường hợp thứ tự tương tác có ý nghĩa quan trọng.	UML 1.x
Giao tiếp/Communication	Cách các đối tượng tương tác và các liên kết cần thiết để thực hiện các tương tác	Đổi tên sơ đồ cộng tác/collaboration, UML 1.x
Tính giờ/Timing	Tương tác giữa các đối tượng trong trường hợp thời gian là yếu tố quan trọng	UML 2.x
Tương tác tổng quan/Interaction Overview	Được sử dụng để tổng hợp các sơ đồ tuần tự, giao tiếp, và tính giờ để tìm tương tác quan trọng diễn ra trong hệ thống	UML 2.x

Các sơ đồ UML 2.5



Nội dung

- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu



Dự án thường được phát sinh từ các nhu cầu nghiệp vụ

- Có thể được xác định bởi người làm nghiệp vụ
- Có thể được xác định bởi người làm CNTT
- Hoặc (tốt hơn nữa) được xác định đồng thời bởi cả hai.

Các dự án phát triển Hệ thống thông tin thường:

- Khai thác 1 tiềm năng
 - Bước đi chiến lược
 - Đem lại lợi thế cạnh tranh
- Giải quyết 1 vấn đề:
 - Hệ thống hiện tại liên tục phát sinh sự cố vận hành
 - Nhu cầu người dùng không được đáp ứng
- Đáp ứng 1 chỉ thị từ bên ngoài
 - Tạo lập các biểu mẫu báo cáo theo quy định của pháp luật

Có thể chia các giá trị mà dự án sẽ tạo ra thành 2 nhóm:

- Giá trị hữu hình
 - Có thể định lượng và đo đếm trực tiếp được
 - Ví dụ: giảm 2% trong chi phí vận hành
- Giá trị vô hình
 - Chúng ta biết nó sẽ tạo ra giá trị và tiết kiệm thời gian, nhưng chúng ta có thể không định lượng được hoặc đo được những lợi ích của nó
 - Ví dụ: Làm khách hàng hài lòng hơn; nâng cao trình độ chuyên môn trong bộ phận IT; v.v.

Đánh giá tính khả thi

- Dự án này có khả thi hay không?
 - Các rủi ro là gì?
 - Có tránh được những rủi ro đó hay không?
- Các thành phần chính:
 - Tính khả thi kỹ thuật (Có đủ khả năng xây dựng nó?)
 - Tính khả thi kinh tế (Có lợi khi xây dựng nó?)
 - Tính khả thi tổ chức (Mọi người sẽ sử dụng nó?)

Xác định rủi ro trong những mảng sau:

- Mảng chức năng: Những người phân tích có kinh nghiệm trong lĩnh vực nghiệp vụ này không?
- Công nghệ: Càng ít kinh nghiệm với công nghệ được lựa chọn thì rủi ro càng cao
- Quy mô dự án: Dự án càng lớn thì càng có nhiều rủi ro
- Tính tương thích: Càng khó tích hợp thì rủi ro càng cao

Tính khả thi kinh tế (Phân tích Chi phí-Lợi ích)

- Xác định các khoản chi phí các lợi ích
- Gán giá trị cho các khoản chi phí và các lợi ích
- Xác định dòng tiền
- Xác định giá trị bằng một hoặc nhiều phương pháp
 - Giá trị hiện tại thuần (NPV)
 - Tỷ lệ hoàn vốn đầu tư (ROI)
 - Điểm hòa vốn

Tính khả thi tổ chức

- Người dùng sẽ đón nhận hệ thống?
- Dự án có thuận theo chiến lược chung?
- Phân tích các bên liên quan:
 - Người ủng hộ dự án
 - Ban quản lý
 - Người dùng hệ thống
 - v.v..

Lựa chọn dự án

- Các dự án được phê duyệt, từ chối hoặc tạm hoãn dựa trên các kết quả đánh giá thực trạng
- Các dự án được lựa chọn được đưa vào quy trình quản lý dự án

- Một số khái niệm cơ bản
- Vòng đời phát triển hệ thống (SDLC)
- Hai cách tiếp cận phát triển hệ thống
- Ngôn ngữ mô hình hóa hợp nhất (UML)
- Môi trường & Nhu cầu
- Xác định yêu cầu



Tổng quan về xác định yêu cầu

- Mục đích: Chuyển đổi các nhu cầu nghiệp vụ thành các yêu cầu chi tiết hơn.
- Một bước rất quan trọng có tầm ảnh hưởng trên toàn bộ SDLC
- Thay đổi liên quan đến yêu cầu có thể dễ dàng được thực hiện ở giai đoạn bắt đầu dự án nhưng càng về sau càng khó thực hiện
- Hầu hết (>50%) dự án thất bại có nguyên nhân liên quan đến yêu cầu
- Tính lặp và tăng dần của PT&TK HĐT hiệu quả bởi vì:
 - Mỗi vòng lặp xác định và triển khai một gói yêu cầu nhỏ
 - Hệ thống phát triển tăng dần theo thời gian

Khái niệm yêu cầu

- Yêu cầu là gì?
 - Yêu cầu mô tả những gì hệ thống phải làm hoặc đặc điểm mà nó phải có,
 - cung cấp thông tin cần thiết cho các chuỗi công việc tiếp theo,
 - là cơ sở của những đặc tả kỹ thuật về cách triển khai hệ thống,
 - xác định phạm vi hệ thống.
- Các loại yêu cầu:
 - Chức năng: Liên quan đến xử lý hoặc dữ liệu
 - Phi chức năng: Liên quan đến hiệu năng hoặc đặc điểm sử dụng
- Các yêu cầu thường được liệt kê theo danh mục
- Có thể được đánh giá mức ưu tiên

Khái niệm yêu cầu₍₂₎

- Yêu cầu chức năng có liên quan chặt chẽ với các quy trình nghiệp vụ và các quy định của tổ chức, và được tập trung nghiên cứu ở pha phân tích
 - Được mô tả chủ yếu bằng các ca sử dụng: Sơ đồ và đặc tả chi tiết.
- Yêu cầu phi chức năng được tập trung nghiên cứu ở pha thiết kế và có thể tiếp tục được phân loại thành:
 - (Các yêu cầu) vận hành: Các đặc điểm liên quan đến cách sử dụng
 - Độ tin cậy: - các lỗi có thể phát sinh và khả năng khắc phục
 - Hiệu năng: - tải và thời gian phản hồi
 - Bảo mật: - kiểm soát truy cập và bảo vệ dữ liệu.

- Yêu cầu chức năng: **F**unctional requirements
- Yêu cầu phi chức năng: Non-functional
 - Vận hành/sử dụng: **U**sability
 - Độ tin cậy: **R**eliability
 - Hiệu năng: **P**erformance
 - Bảo mật: **S**ecurity
- Phân loại FURPS (**F**unctions, **U**sability, **R**eliability, **P**erformance, **S**ecurity Requirements).

Ví dụ 2.1. Các yêu cầu hệ thống

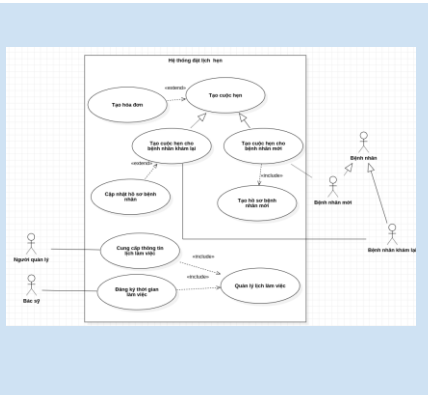
- Yêu cầu chức năng:
 - Tạo phiếu mượn sách (trong thư viện)
 - Tra cứu ghế trống (trong rạp chiếu phim)
- Yêu cầu phi chức năng:
 - Vận hành:
 - Sử dụng màu đỏ làm màu chủ đạo trong thiết kế giao diện.
 - Nhân viên sử dụng máy tính bảng để phục vụ gọi món tại bàn.
 - Độ tin cậy:
 - Thông tin được bảo toàn trong trường hợp phát sinh sự cố đột ngột.
 - Hiệu năng:
 - Có khả năng đáp ứng yêu cầu của 1000 người dùng sử dụng đồng thời
 - Thời gian phản hồi cho mỗi thao tác không quá 0.5 s.
 - Bảo mật
 - Dữ liệu được gửi từ người dùng phải được mã hóa.
 - Sử dụng giao thức HTTPS cho kết nối giữa người dùng và hệ thống.
 - ...

- Mỗi mô hình biểu diễn một vài khía cạnh của hệ thống
 - Người phân tích tạo các mô hình và sử dụng như công cụ trao đổi hiểu biết hiện tại về hệ thống với khách hàng và những người khác cùng phát triển hệ thống.
 - Để xác nhận tính đúng đắn và sử dụng cho những hoạt động tiếp theo trong SDLC
 - Có thể được tạo bằng nhiều vòng lặp
- Các mô hình trong phân tích & thiết kế có thể được chia thành các lớp:
 - Văn bản: Ví dụ, các đặc tả ca sử dụng.
 - Hình vẽ: Ví dụ, các sơ đồ UML
 - Toán học: Ví dụ, công thức tính giá khuyến mãi cho đơn hàng

Ví dụ 1.2. Các mô hình

- 1. Mua ô-tô mới
- 2. Bán ô-tô
- 3. Bảo dưỡng ô-tô
- 4. Thanh toán
- 5. Mượn ô-tô

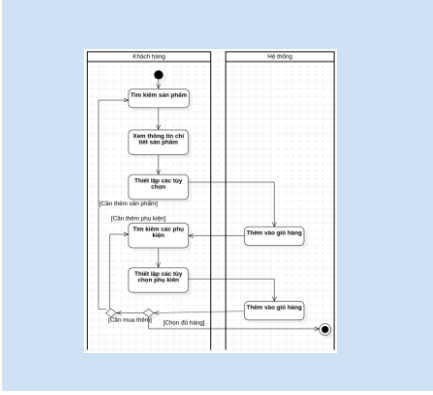
Danh sách sự kiện



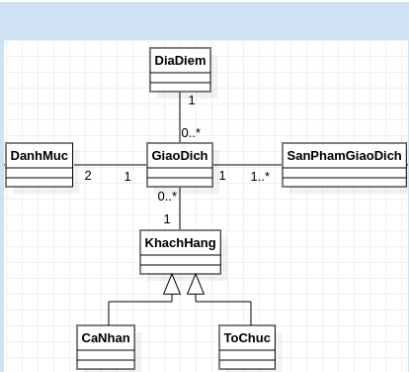
Sơ đồ ca sử dụng



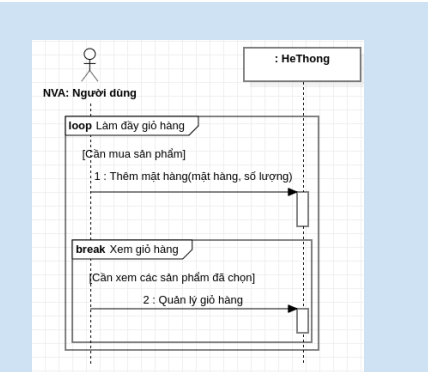
Đặc tả ca sử dụng



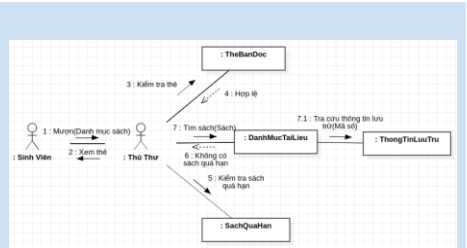
Sơ đồ hoạt động



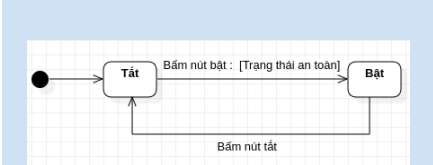
Sơ đồ lớp



Sơ đồ tuần tự



Sơ đồ giao tiếp



Sơ đồ máy trạng thái

Kỹ năng xác định yêu cầu

- Cần kết hợp người làm CNTT & người làm nghiệp vụ
- Các yêu cầu được xác định tốt nhất bởi người phân tích hệ thống và người làm nghiệp vụ làm việc cùng nhau
- Cần các kỹ năng phân tích vấn đề và thu thập thông tin.

Các khó khăn trong xác định yêu cầu

- Người phân tích có thể không tiếp cận được đúng người dùng
- Các đặc tả yêu cầu có thể không đầy đủ
- Yêu cầu có thể không được phát hiện từ ban đầu
 - Cần cố gắng phát hiện tất cả các yêu cầu (các yêu cầu được phát hiện sau trong tiến trình phát triển thường khó kết hợp)
- Có thể gặp khó khăn trong kiểm tra và đánh giá các yêu cầu.

Tổng quan về các chiến lược phân tích vấn đề

- Phân tích nguyên nhân gốc - Tập trung vào nguyên nhân của 1 vấn đề - không phải cách giải quyết nó
 - Tìm hiểu nguyên nhân của các vấn đề và xây dựng giải pháp sau khi các nguyên nhân đã được làm rõ
- Phân tích thời lượng - Xác định thời gian cần thiết để hoàn thành mỗi bước trong 1 quy trình nghiệp vụ
 - So sánh với tổng thời gian cần cho toàn bộ tiến trình
 - Khác biệt lớn cho thấy rằng các vấn đề có thể được giải quyết bằng cách:
 - Kết hợp một số bước
 - Thực hiện một số bước đồng thời (song song)

Tổng quan về các chiến lược phân tích vấn đề₍₂₎

- Chi phí dựa trên hoạt động - Tương tự như thời lượng nhưng áp dụng cho các chi phí
- Đánh giá không chính thức - Tham khảo quy trình tương tự ở môi trường khác.
- Phân tích đầu ra - Kết quả cuối cùng mà khách hàng muốn có
- Phân tích công nghệ - Giả định áp dụng công nghệ mới vào quy trình nghiệp vụ và xác định các lợi ích
- Loại bỏ hoạt động - Thử loại bỏ từng hoạt động trong quy trình nghiệp vụ, và quan sát tiến trình trong điều kiện đó, tìm hiểu các hiệu ứng mà nó tạo ra.

Tổng quan về các kỹ thuật thu thập thông tin

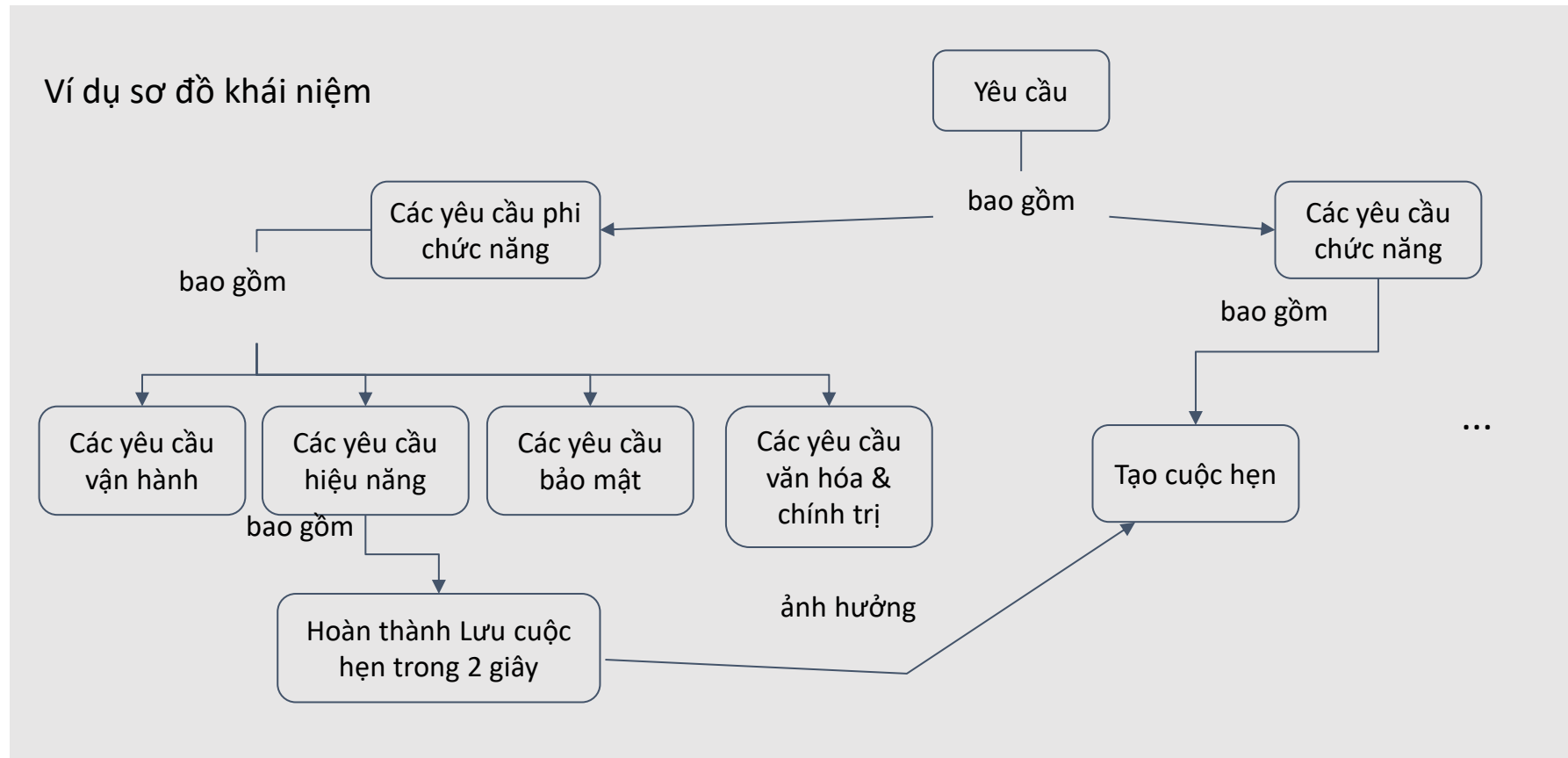
- Phỏng vấn - Kỹ thuật thông dụng nhất - hỏi ai đó khi cần biết về điều gì đó
- Hội thảo liên kết (JAD) - Cuộc họp bao gồm cả người dùng và người phân tích để thảo luận về các vấn đề phát triển hệ thống.
- Khảo sát - Sử dụng 1 tập câu hỏi để thu thập thông tin từ nhiều người.
- Phân tích tài liệu - Tìm hiểu các tài liệu kỹ thuật, các biểu mẫu, báo cáo, các quy định và điều khoản. v.v..
- Quan sát - Quan sát diễn biến thực tế của các hoạt động.

So sánh các kỹ thuật thu thập thông tin

	Phỏng vấn	Hội thảo liên kết	Khảo sát	Phân tích tài liệu	Quan sát
Loại thông tin	Đang có, cải tiến, sẽ có	Đang có, cải tiến, sẽ có	Đang có, cải tiến	Đang có	Đang có
Độ sâu thông tin	Cao	Cao	Trung bình	Thấp	Thấp
Độ rộng thông tin	Thấp	Trung bình	Cao	Cao	Thấp
Tích hợp thông tin	Thấp	Cao	Thấp	Thấp	Thấp
Người dùng tham gia	Trung bình	Cao	Thấp	Thấp	Thấp
Chi phí	Trung bình	Thấp-Trung bình	Thấp	Thấp	Thấp tới Trung bình

Các kỹ thuật khác

- Sơ đồ khái niệm
 - Biểu diễn trực quan mối quan hệ giữa các khái niệm
 - Tập trung các cá nhân vào một lượng nhỏ những ý tưởng chính



Các kỹ thuật khác₍₂₎

- Các thẻ câu chuyện & các danh sách nhiệm vụ
 - Mô tả yêu cầu người dùng theo cách tự nhiên nhất

1. Là một thư ký, tôi mong muốn có khả năng lập lịch hẹn cho những bệnh nhân để có thể đáp ứng tốt nhất các nhu cầu khám chữa bệnh.

2. Tôi mong muốn hệ thống có kết nối không dây với máy in để nhiều máy tính có thể dùng chung 1 máy in mà không cần đến những dây cáp kết nối phức tạp.

V.V...

- [1] Satzinger, Jackson, Burd. Systems Analysis and Design in a changing world, 7th edition.
- [2] Dennis, Wixon, Tegarden. Systems Analysis & Design: An object-oriented approach with UML, 5th edition.

A large graphic on the left side of the slide. It features a dark blue background with a circular pattern of red dots of varying sizes, creating a sense of depth and movement. The word "HUST" is centered within this graphic in a white, bold, sans-serif font.

HUST

THANK YOU !