

Phân tích thiết kế hệ thống

Soạn bởi: Nguyễn Bá Ngọc

Chương 1

Hà Nội-2022


Chương 1.

Tổng quan về phân tích và thiết kế 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)

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)

Chương trình máy tính và hệ thống thông tin

- Chương trình máy tính - là một chương trình phần mềm chạy trên một thiết bị tính toán 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

- 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

Quy trình

- 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)

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.

SDLC

Có nhiều cách thiết lập SDLC, tuy nhiên hầu hết các SDLC đều bao gồm một số pha cơ bản.

- Các pha cơ bản điển hình trong các SDLC:
 - Lập kế hoạch;
 - Phân tích;
 - Thiết kế;
 - Thực thi.
- Mỗi pha bao gồm một 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 một 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

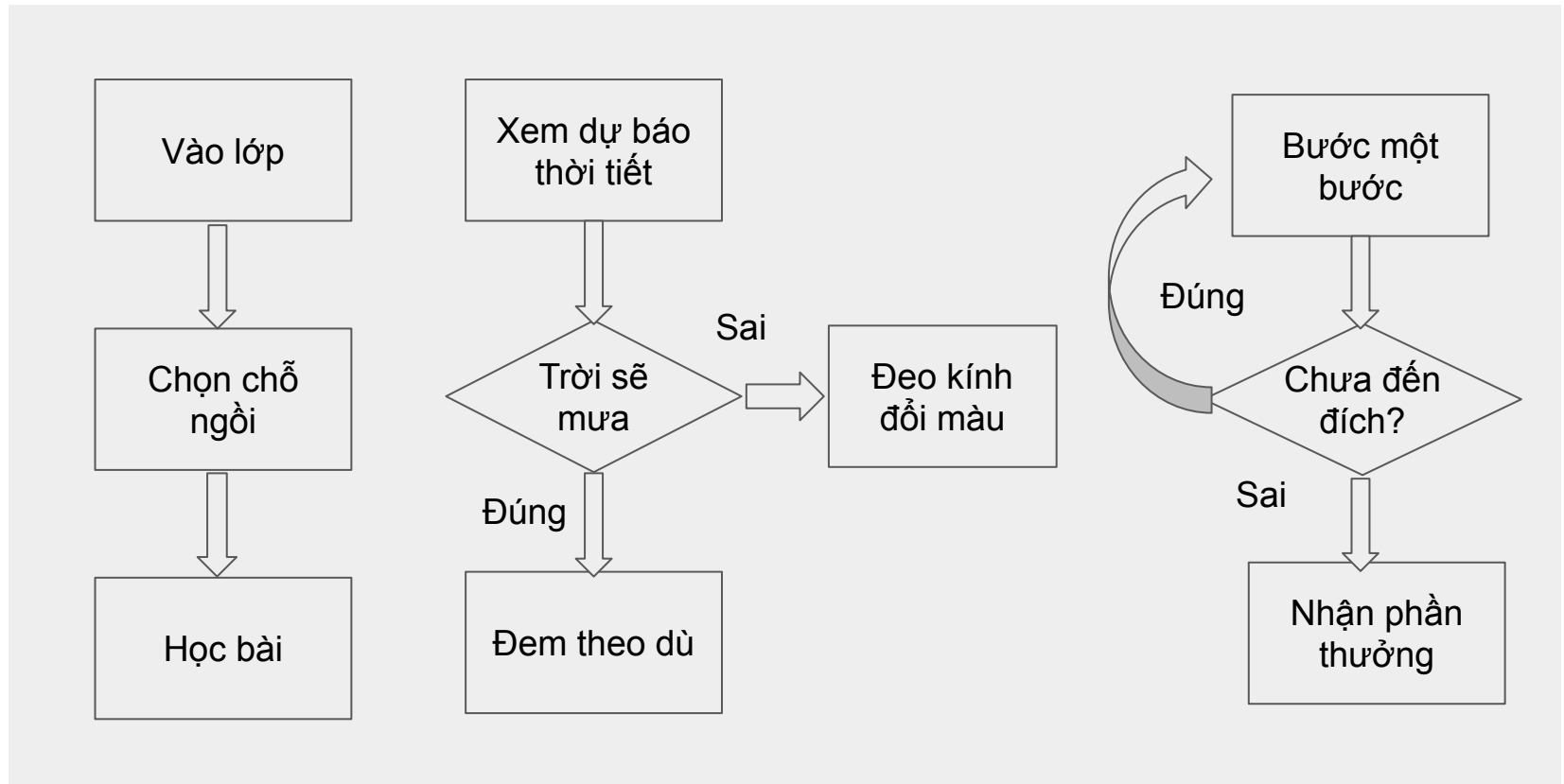
- Một số khái niệm cơ bản
 - Vòng đời phát triển hệ thống (SDLC)
 - Hai cách tiếp cận phát triển hệ thống
 - Ngôn ngữ mô hình hóa hợp nhất (UML)
- 

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

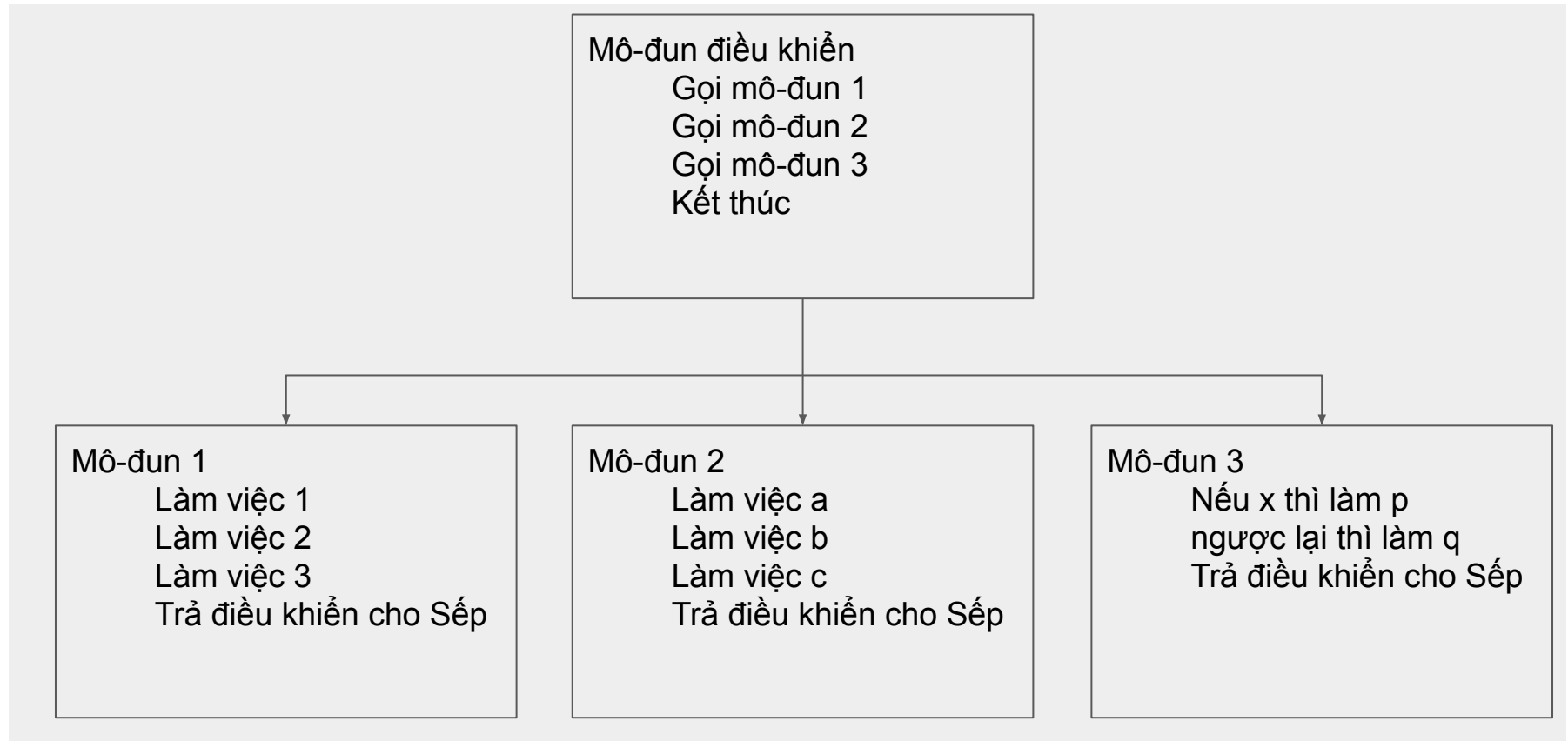
Cách tiếp cận hướng cấu trúc

- Lập trình hướng cấu trúc
 - Biến
 - Xử lý tuần tự, rẽ nhánh, và lặp



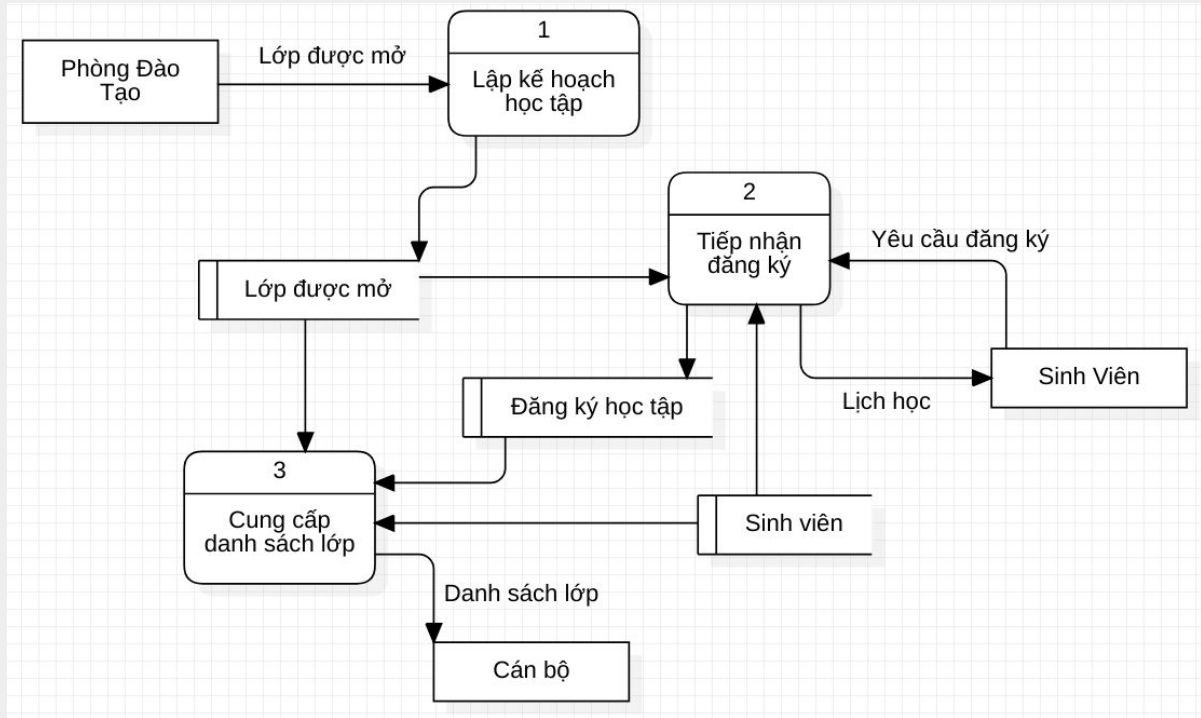
Cách tiếp cận hướng cấu trúc₍₂₎

- Chia nhỏ từ trên xuống



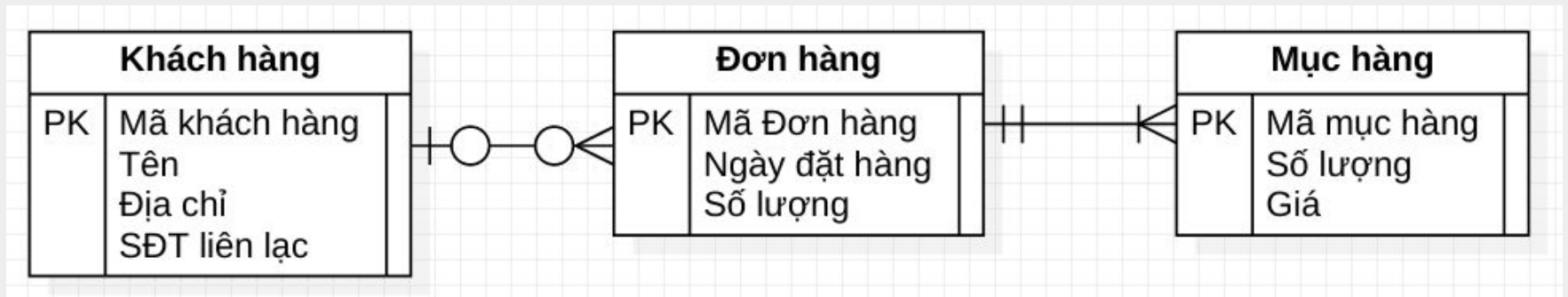
Cách tiếp cận hướng cấu trúc⁽³⁾

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



Cách tiếp cận hướng cấu trúc₍₄₎

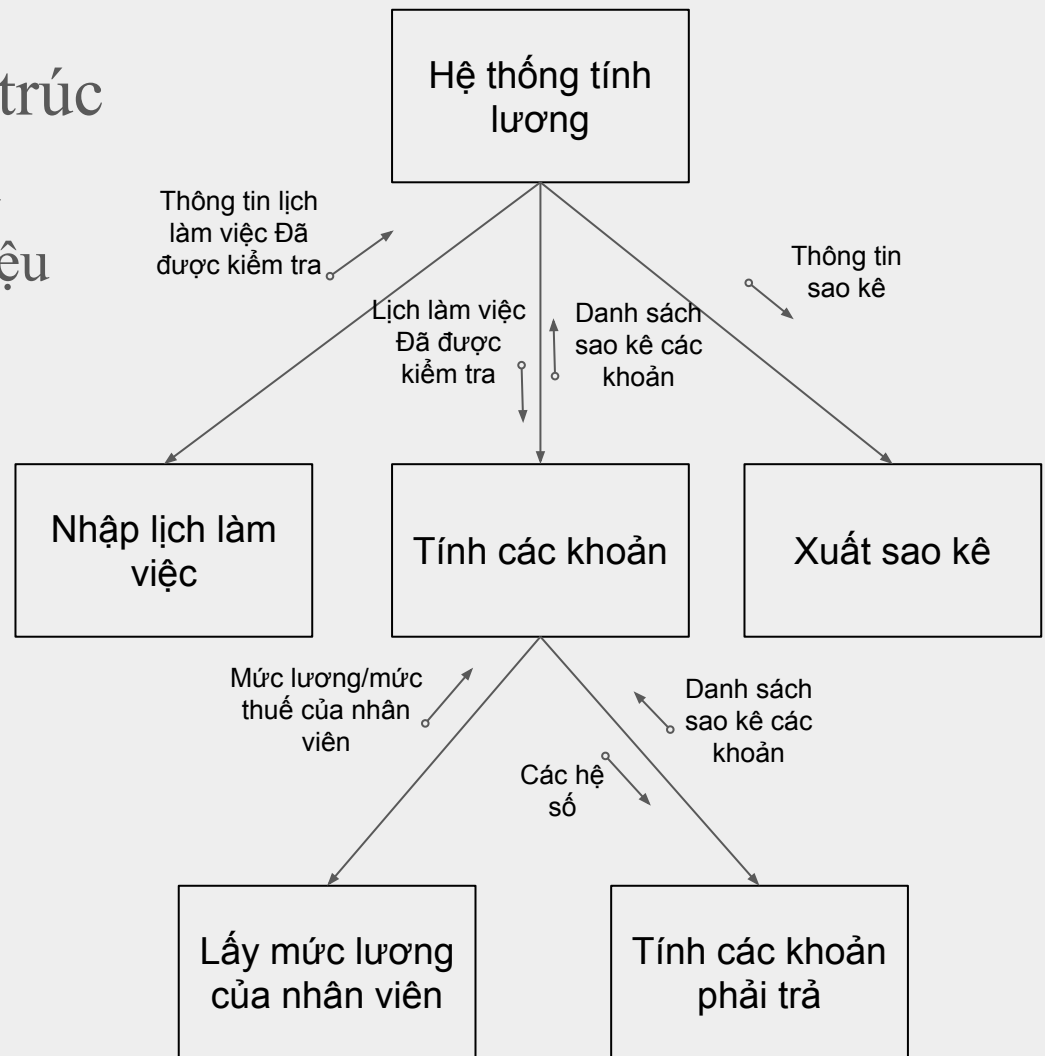
- Mô hình hóa dữ liệu với biểu đồ thực thể-liên kết



Cách tiếp cận hướng cấu trúc⁽⁵⁾

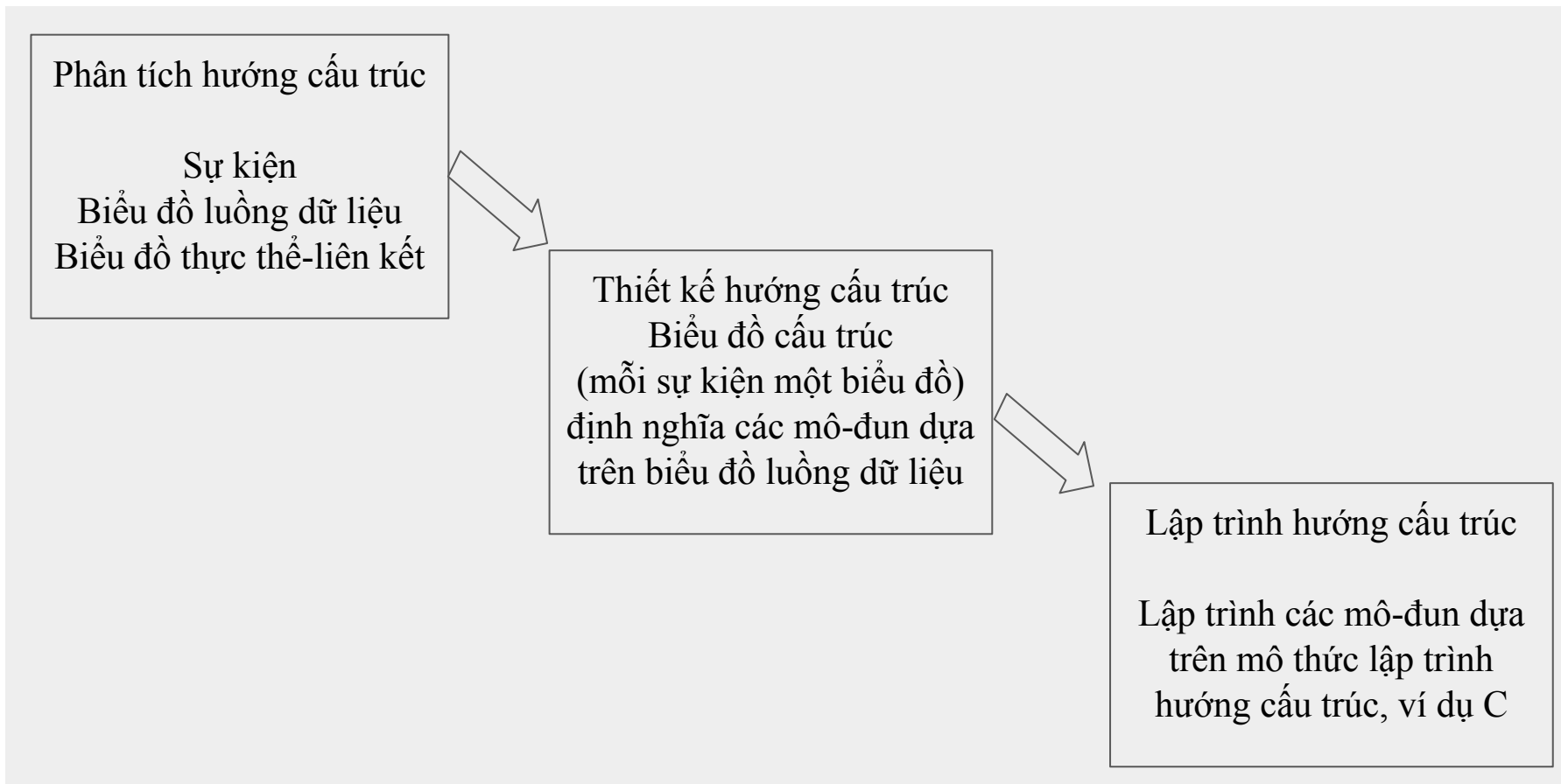
- Thiết kế hướng cấu trúc

- Biểu diễn các thành phần cùng với dữ liệu liên quan.



Cách tiếp cận hướng cấu trúc₍₆₎

- Kết hợp các thành phần

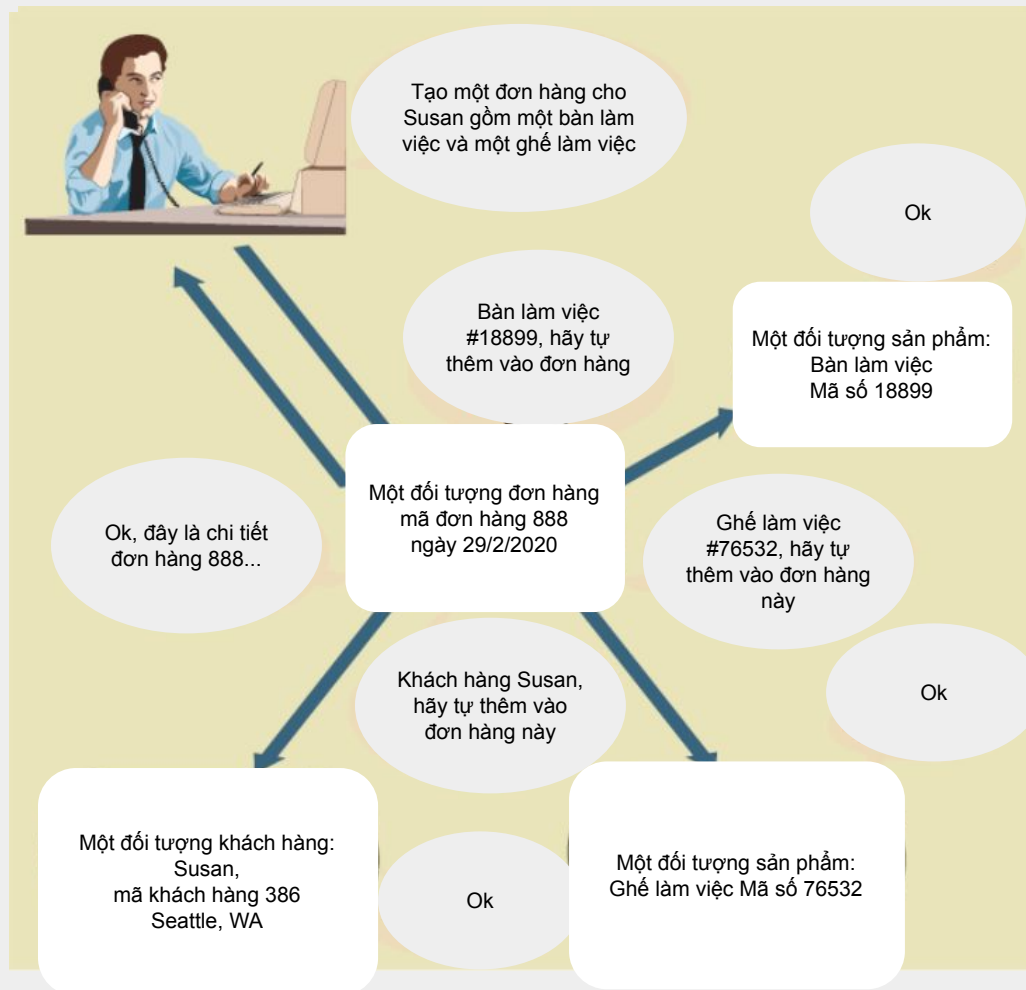


Cách tiếp cận hướng đối tượng

- 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

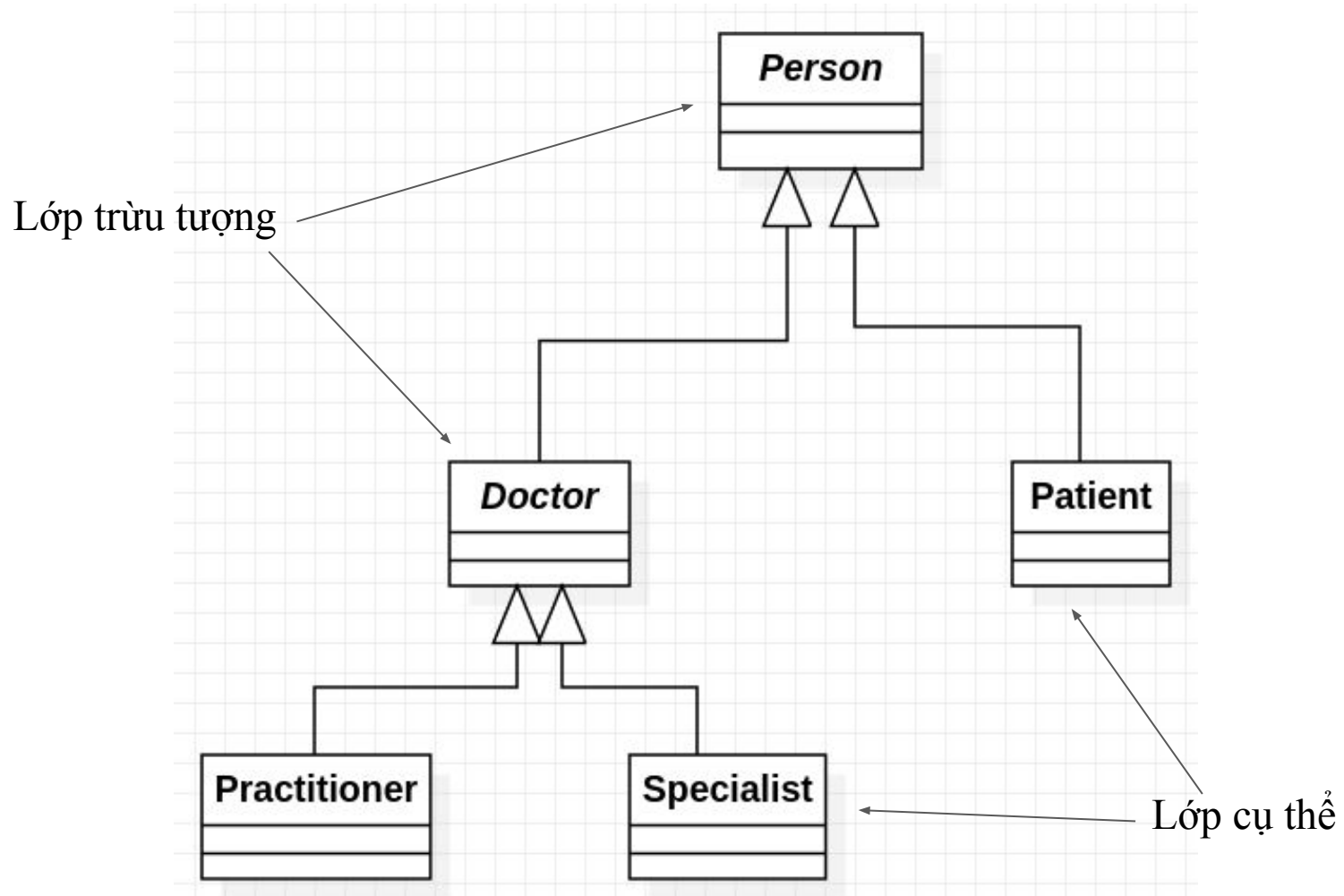


Một số phần tử và tính năng cơ bản

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

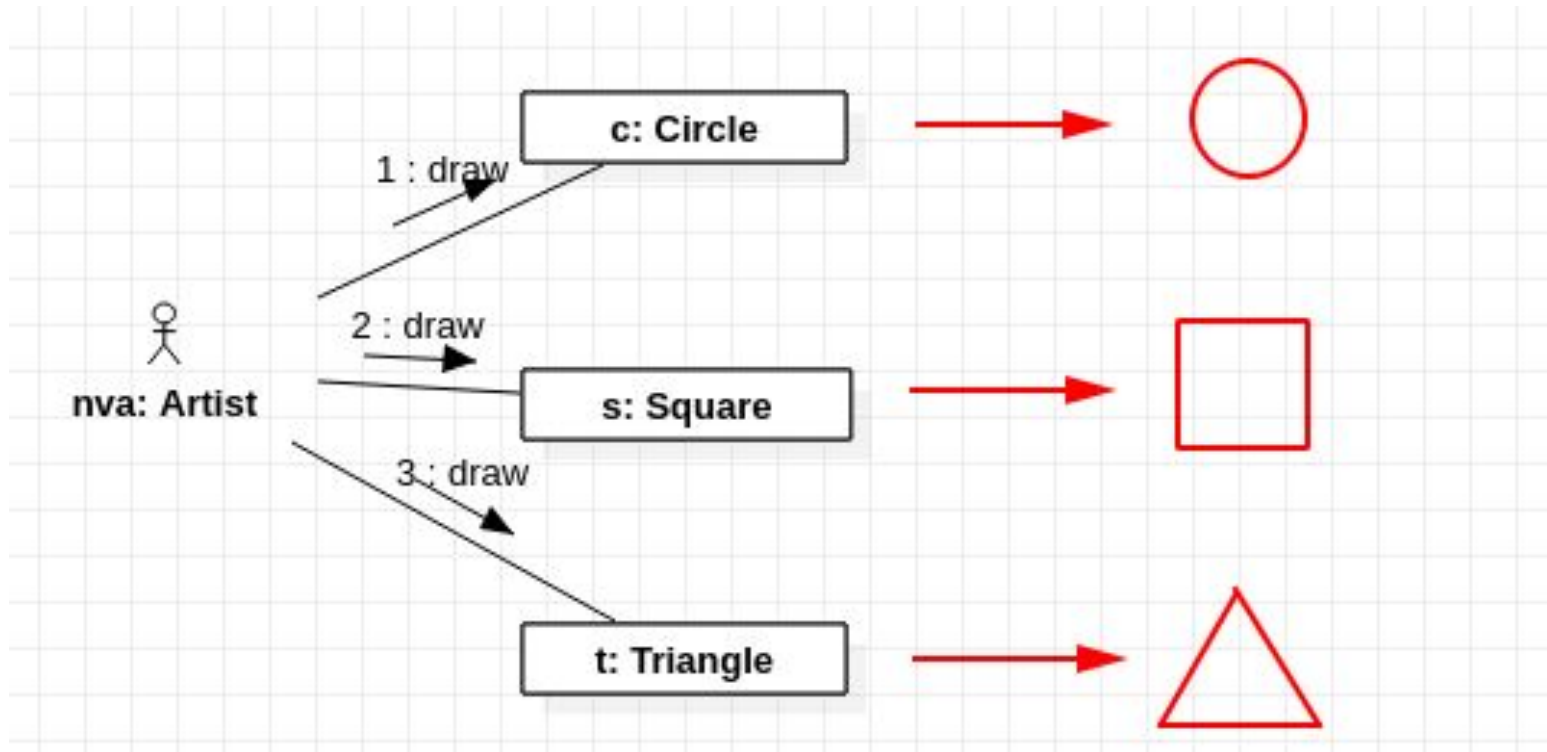
Một số phần tử và tính năng cơ bản₍₂₎

- Kế thừa



Một số phần tử và tính năng cơ bản₍₃₎

- Đa hình



Phân tích & Thiết kế Hệ thống theo phương pháp Hướng Đối Tượng

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

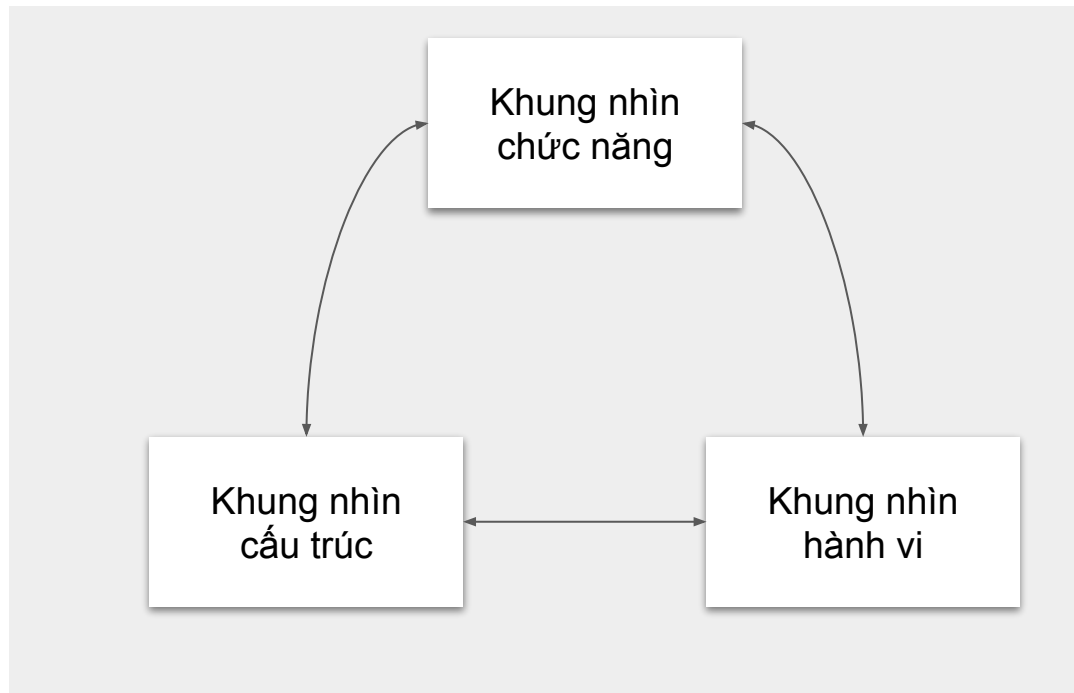
[Grady Booch, Ivar Jacobson, and James Rumbaugh]

Phân tích & Thiết kế 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 một 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

Phân tích & Thiết kế Hệ thống theo phương pháp Hướng Đối Tượng₍₃₎

- Lặp 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 một 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)



Ngôn ngữ mô hình hóa hợp nhất

- Unified Modeling Language (UML)
- Ngôn ngữ trực quan, cung cấp nhiều biểu 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

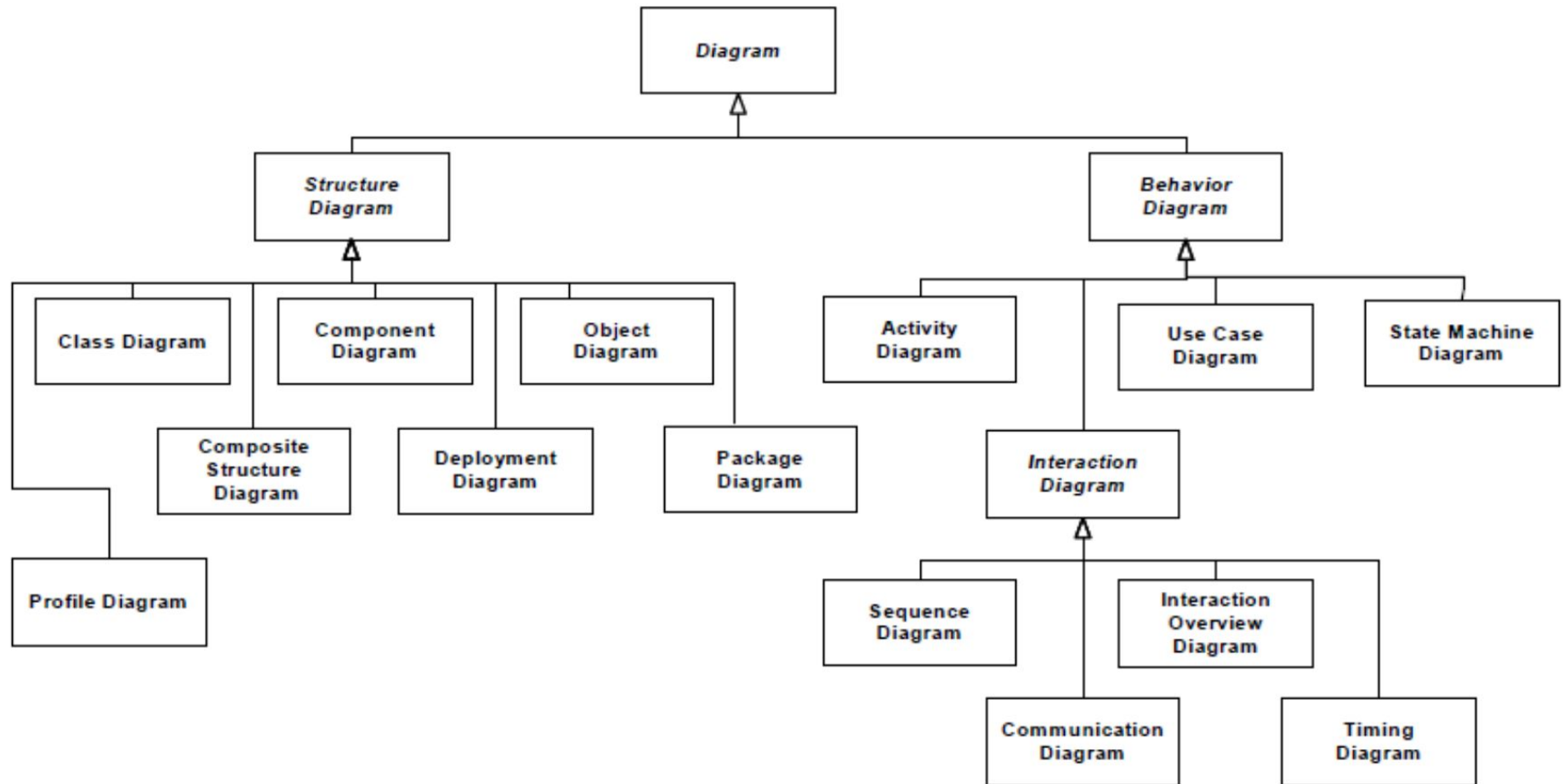
Các biểu đồ UML

Loại biểu 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 biểu 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 biểu đồ UML₍₂₎

Loại biểu 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 biểu 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 biểu đồ tuần tự, giao tiếp, và tính giờ để tìm tương tác quan trọng diễn ra trong hệ thống	UML 2.x

Các biểu đồ UML 2.5



Tham khảo

[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.