MÔ HÌNH HOÁ HỆ THỐNG

ThS. Dương Hữu Thành,
Khoa CNTT, Đại học Mở TP.HCM,
thanh.dh@ou.edu.vn.

Nội dung chính

1. Mô hình hoá hệ thống
2. Sơ đồ lớp
3. Sơ đồ hoạt động
4. Sơ đồ tuần tự
5. Sơ đồ trạng thái
6. Một số sơ đồ khác

Mô hình hoá hệ thống

- Mô hình hoá hệ thống là quá trình phát triển mô hình trừu tượng cho hệ thống, mỗi mô hình thể hiện khung nhìn (view) khác nhau về hệ thống.
- Mô hình hoá hệ thống thường sử dụng các ký hiệu đồ hoạ UML (Unified Modeling Language).

Mô hình hoá hệ thống

- Mô hình hoá hệ thống có thể sử dụng trong các quá trình
  - Phân tích yêu cầu giúp hiểu rõ yêu cầu hệ thống.
  - Thiết kế để mô tả hệ thống giúp các kỹ sư hiện thực hệ thống.
  - Tài liệu hoá các hoạt động và cấu trúc hệ thống sau giai đoạn hiện thực hệ thống.

Mô hình hoá hệ thống

- Mục đích mô hình hoá
  - Làm sáng tỏ vấn đề.
  - Mô phỏng các hình ảnh hệ thống.
  - Đơn giản hoá hệ thống.
  - Tăng khả năng bảo trì hệ thống.

Mô hình hoá hệ thống

- UML là ngôn ngữ mô hình hoá hợp nhất để xác định (specifying), trực quan hoá (visualizing), xây dựng (constructing), tài liệu hoá (documenting) cho các artifacts của quy trình hiện thực phần mềm.
  - Artifacts mô tả kết quả được tạo ra sau mỗi bước lặp nào đó của quy trình phần mềm.

Mô hình hoá hệ thống

- Mô hình tĩnh (static model) mô tả các thông số, cấu trúc và khía cạnh tĩnh của hệ thống.
- Mô hình động (dynamic model) mô tả hành vi, thủ tục của hệ thống. Các mô hình động biểu diễn sự tương tác các đối tượng để thực hiện một công việc hệ thống.

Mô hình hoá hệ thống

- Các tiếp cận mô hình hoá hệ thống
  - Bên ngoài (external): mô hình ngữ cảnh hoặc môi trường hệ thống.
  - Tương tác (interaction): mô hình tương tác giữa hệ thống và môi trường của nó hoặc giữa các thành phần trong hệ thống.
  - Cấu trúc (structural): mô hình tổ chức hệ thống thành component và mối quan hệ giữa chúng.
  - Hành vi (behavioral): mô hình hành vi động của hệ thống khi nó thực thi hoặc cách thức phản hồi của hệ thống với một sự kiện nào đó.

Các sơ đồ UML

- Structural Diagram
  - Class Diagram
  - Package Diagram
  - Object Diagram
  - Composite Structure Diagram
  - Component Diagram
  - Deployment Diagram
- Behavioral Diagram
  - Use case Diagram
  - Activity Diagram
  - State Diagram
  - Sequence Diagram
  - Communication Diagram

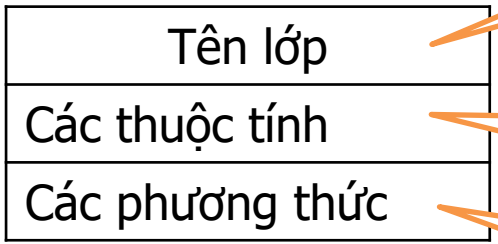
Sơ đồ lớp

- Sơ đồ lớp (Class Diagram) là một mô hình cấu trúc tĩnh của hệ thống, dùng biểu diễn các lớp đối tượng và mối quan hệ giữa chúng.
- Ký hiệu lớp

Tên lớp
tên-thuộc-tính1: kiểu-dữ-liệu1 = giá-trị-mặc-định1 tên-thuộc-tính2: kiểu-dữ-liệu2 = giá-trị-mặc-định2
tên-phương-thức1(ds-tham-số1): kiểu-trả-về1 tên-phương-thức2(ds-tham-số2): kiểu-trả-về2

Sơ đồ lớp

+: public
–: private
#: protected
~: package
/: thuộc tính suy diễn



Bình thường: lớp bình thường
In nghiêng: lớp trừu tượng
Gạch chân: đối tượng

Bình thường: thuộc tính bình thường
Gạch chân: thuộc tính tĩnh

Bình thường: phương thức bình thường
In nghiêng: phương thức trừu tượng
Gạch chân: phương thức tĩnh

Sơ đồ lớp

- Quan hệ Association

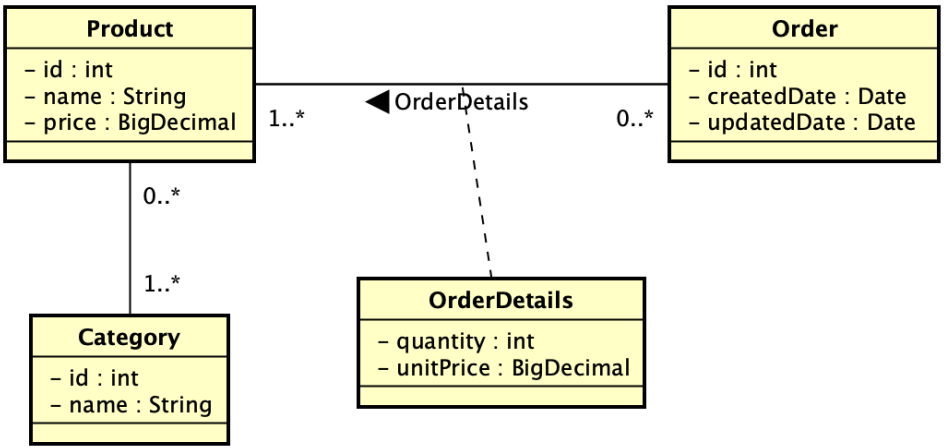
- Bản số (multiplicity): xác định số đối tượng của hai lớp có thể liên kết với nhau, chẳng hạn:
  - *: nhiều
  - 0..*: từ 0 đến nhiều
  - 1..*: từ 1 đến nhiều
  - 1..3: từ 1 đến 3
  - 1: một

Sơ đồ lớp

- Chú ý khi giữa hai lớp có quan hệ nhiều-nhiều, UML mô hình hoá các thuộc tính quan hệ giữa chúng trong lớp kết hợp (association class).
- Lớp kết hợp kết nối vào mối quan hệ hai lớp bằng đường nét đứt (dotted line).

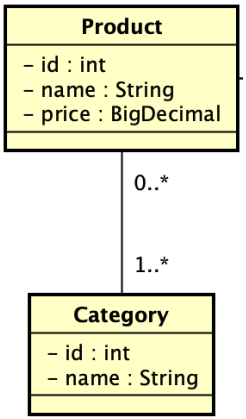
Sơ đồ lớp

- Lớp Product và Order có quan hệ n-n, và các thuộc tính quan hệ giữa chúng là quantity và unitPrice được biểu diễn trong lớp kết hợp orderDetails



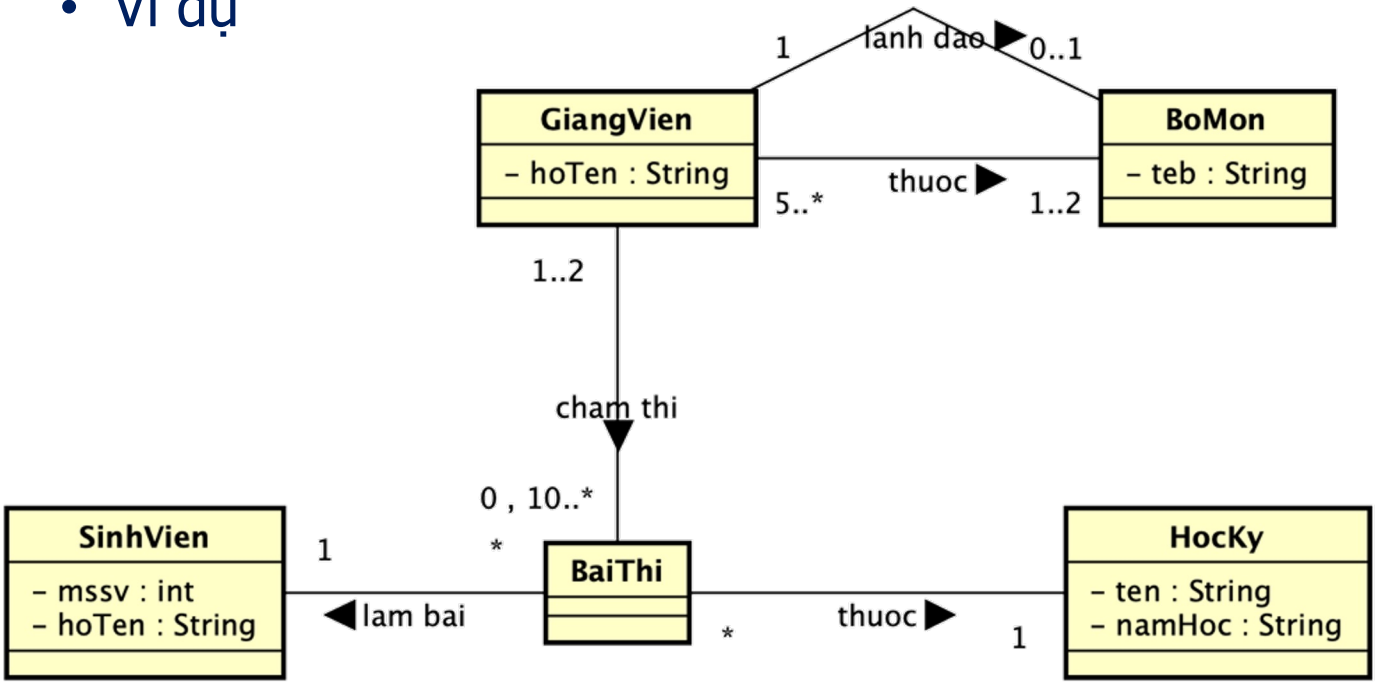
Sơ đồ lớp

- Nếu giữa hai lớp trong quan hệ n-n không có thêm các thuộc tính quan hệ nào thì UML sẽ bỏ qua lớp kết hợp.
- Trong ví dụ trên giữa Product và Category có quan hệ n-n và không có thuộc tính quan hệ nào khác.



Sơ đồ lớp

- Ví dụ



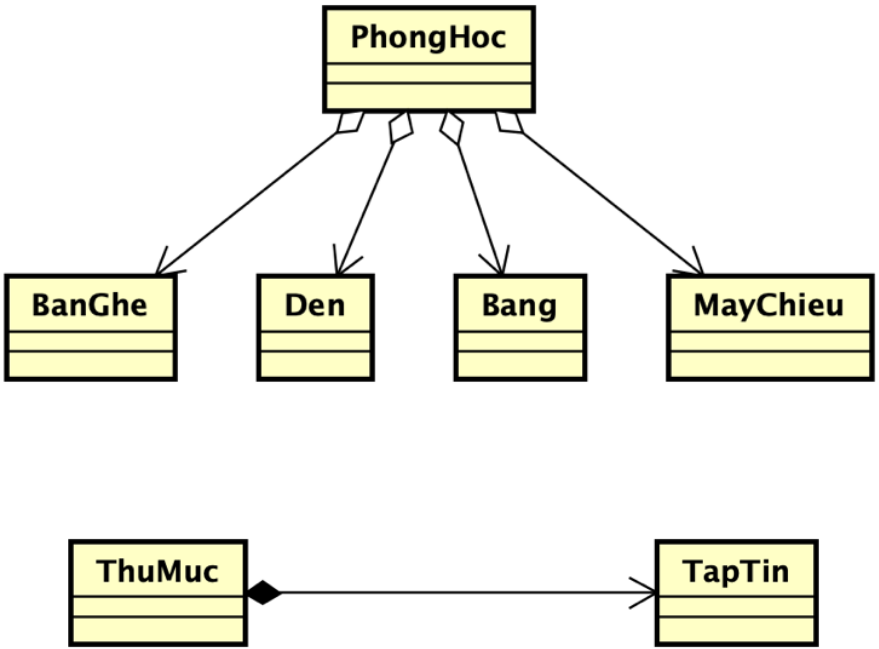
Sơ đồ lớp

- Quan hệ Aggregation (whole-part)

- Quan hệ Composition (has-a)

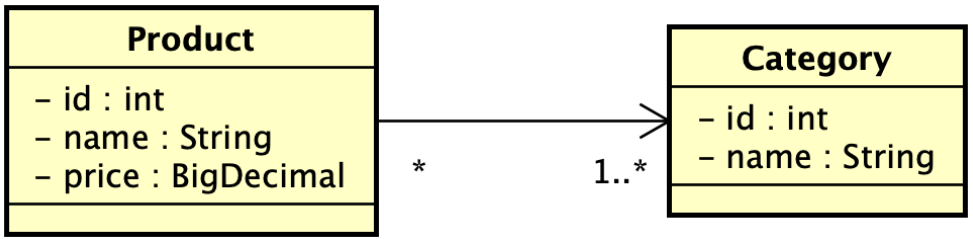
- Quan hệ Dependency (uses-a)

Sơ đồ lớp



Sơ đồ lớp

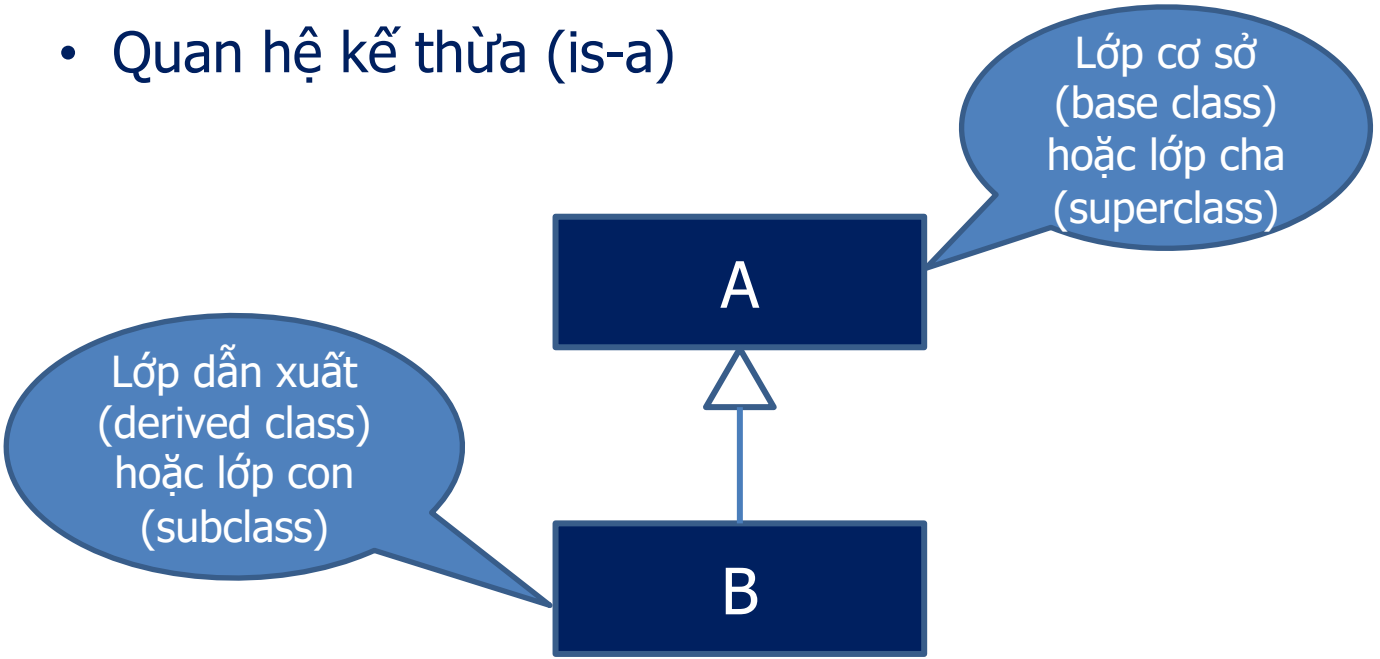
- Dấu mũi tên chỉ định quan hệ một chiều, thể hiện lời gọi hàm theo chiều đó.



- Một đối tượng của Product yêu cầu phải biết các đối tượng của Category mà nó kết hợp, nhưng ngược lại thì không cần thiết.

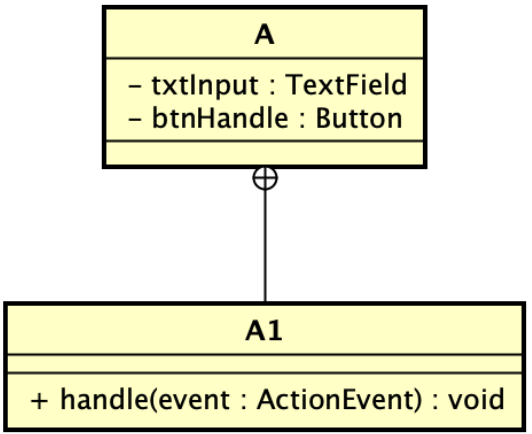
Sơ đồ lớp

- Quan hệ kế thừa (is-a)



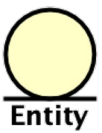
Sơ đồ lớp

- Lớp A1 là lớp trong (nested) của lớp A ký hiệu như sau:



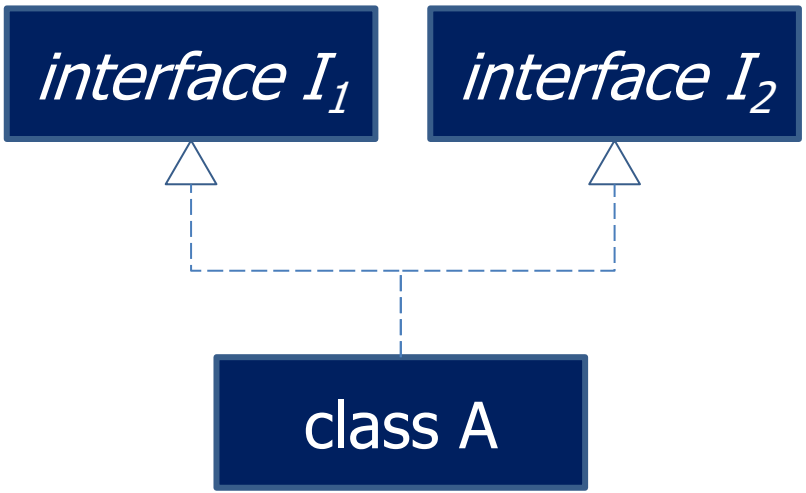
Sơ đồ lớp

- Boundary Class: dùng mô hình hoá sự tương tác giữa người dùng và hệ thống, chẳng hạn như giao diện chương trình.
- Entity Class: dùng mô hình hoá thông tin của thành phần tồn tại lâu dài. Nó đại diện cho dữ liệu hệ thống như Product, Transaction, v.v.
- Controller Class: biểu diễn các đối tượng trung gian giữa boundary và entity. Nó thực thi các lệnh từ boundary bằng cách tương tác với các đối tượng entity và boundary.



Sơ đồ lớp

- Sự hiện thực hoá (Realization)



Bài tập vẽ sơ đồ lớp

- Thư mục chứa thông tin tên thư mục, ngày tạo. Tập tin cũng là một loại thư mục có thêm thông tin loại tập tin và dung lượng. Thư mục có thể chứa các thư mục con và nhiều tập tin khác. Trong đó:
  - Thư mục bắt buộc phải có tên, còn tập tin phải có tên và dung lượng.
  - Một thư mục bị xoá thì tất cả tập tin, thư mục con của nó cũng bị xoá theo.
- Thiết kế các lớp **`TapTin`**, **`ThuMuc`** và quan hệ giữa chúng theo yêu cầu trên.

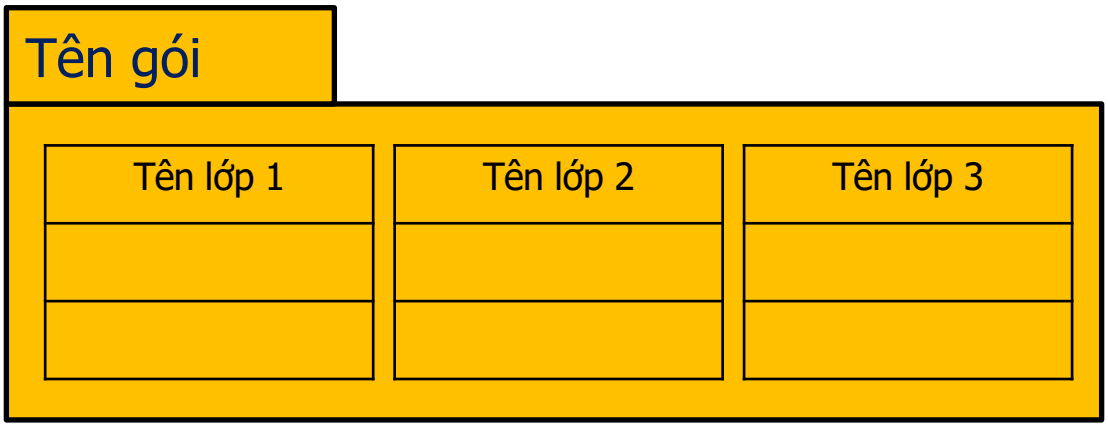
Bài tập vẽ sơ đồ lớp

Một trường đại học xây dựng hệ thống quản lý minh chứng phục vụ kiểm định chương trình đào tạo theo các bộ kiểm định khác nhau (như MOET, AUN, ABET, …). Một bộ kiểm định gồm tối thiểu là 5, tối đa 20 tiêu chuẩn, mỗi tiêu chuẩn có tối thiểu là 3 tiêu chí, mỗi tiêu chí phải thuộc vào một tiêu chuẩn nào đó. Khi tiêu chuẩn bị xoá thì các tiêu chí của nó cũng sẽ bị xoá theo. Cả tiêu chuẩn và tiêu chí gồm thông tin mã, tên, nội dung của tiêu chuẩn/tiêu chí.

Một tiêu chí có thể có nhiều minh chứng khác nhau, mỗi minh chứng gồm thông tin mã minh chứng, tên minh chứng, nơi ban hành, ngày ban hành. Một minh chứng có thể thuộc vào nhiều tiêu chí. Hệ thống yêu cầu các tính năng thêm/cập nhật/xoá tiêu chuẩn mới vào bộ kiểm định, thêm/cập nhật/xoá tiêu chí vào tiêu chuẩn, thêm/cập nhật/xoá minh chứng vào tiêu chí, tìm kiếm tiêu chuẩn/tiêu chí theo tên, xem danh sách minh chứng tiêu chí, xuất một báo cáo hoàn chỉnh cho một bộ tiêu chuẩn kiểm định.

Sơ đồ gói

- Sơ đồ gói (Package Diagram) để nhóm các thành phần trong UML theo một ngữ cảnh nhất định để đơn giản hoá việc tổ chức, quản lý các thành phần UML



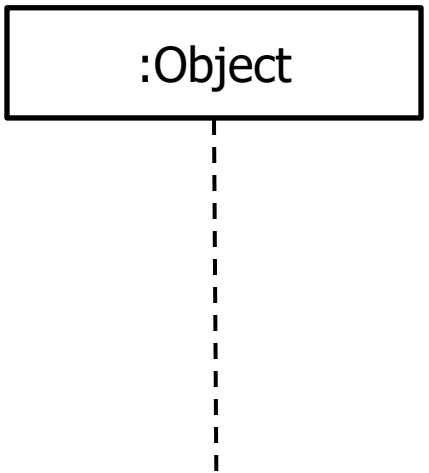
Sơ đồ tuần tự

- Sơ đồ tuần tự (Sequence Diagram) để biểu diễn sự tương tác giữa actor và các đối tượng trong hệ thống hoặc giữa các đối tượng với nhau.

- Sơ đồ tuần tự thể hiện dãy các tương tác xảy ra trong một use case cụ thể.

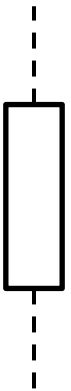
Sơ đồ tuần tự

- Các actor hoặc object nằm ở đỉnh (top) của sơ đồ và các đường thẳng nét đứt xuất phát từ đó xuống dưới sơ đồ.



Sơ đồ tuần tự

- Thời gian sống của đối tượng thể hiện bằng được nét đứt (lifelines) vẽ thẳng từ đối xuống.
- Hình chữ nhật trên đường nét đứt (activations) thể hiện thời gian sống của đối tượng khi thực hiện một service nào đó.



Sơ đồ tuần tự

- Sự tương tác giữa các đối tượng biểu diễn bằng đường mũi tên và các ghi chú tương ứng.
  - Gửi message đồng bộ (synchronous) yêu cầu phản hồi trước khi thực hiện tiếp.

  - Gửi message bất đồng bộ (asynchronous) không yêu cầu phản hồi để thực hiện các tương tác tiếp theo.

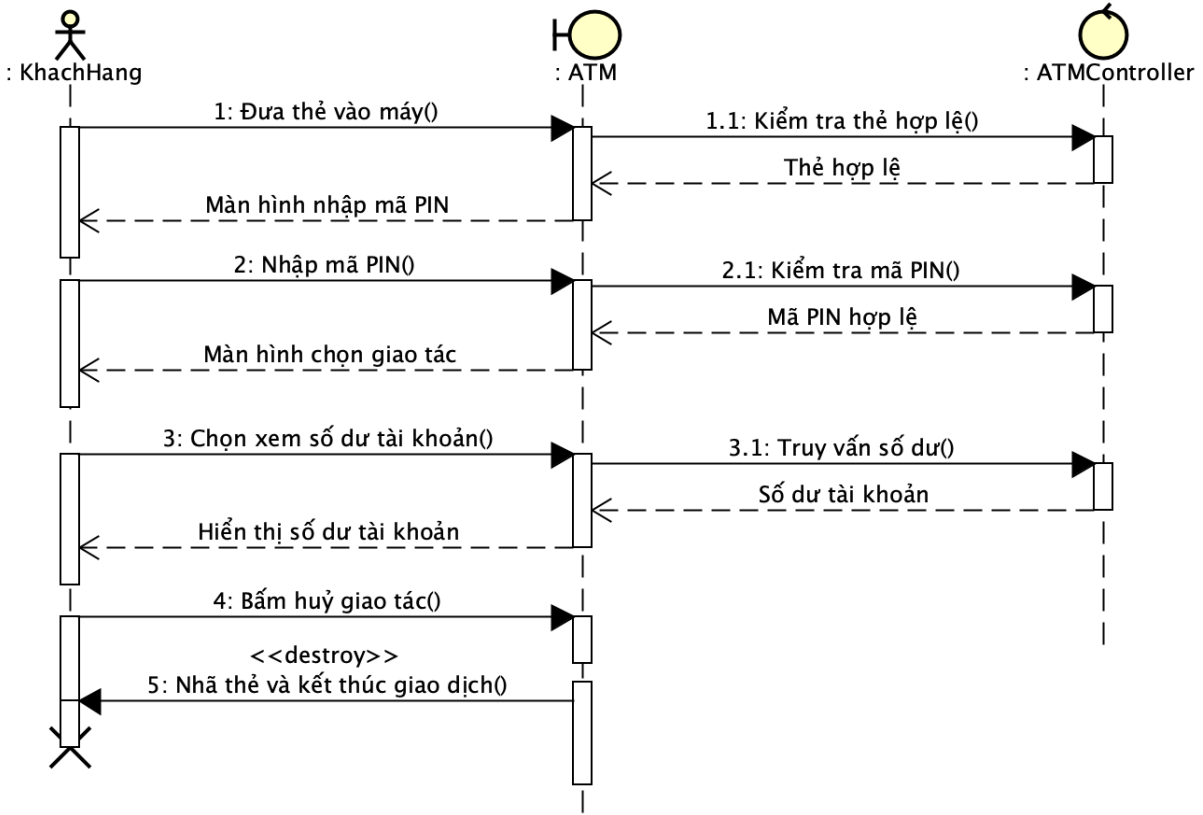
Sơ đồ tuần tự

- Tạo message: dùng tạo đối tượng mới

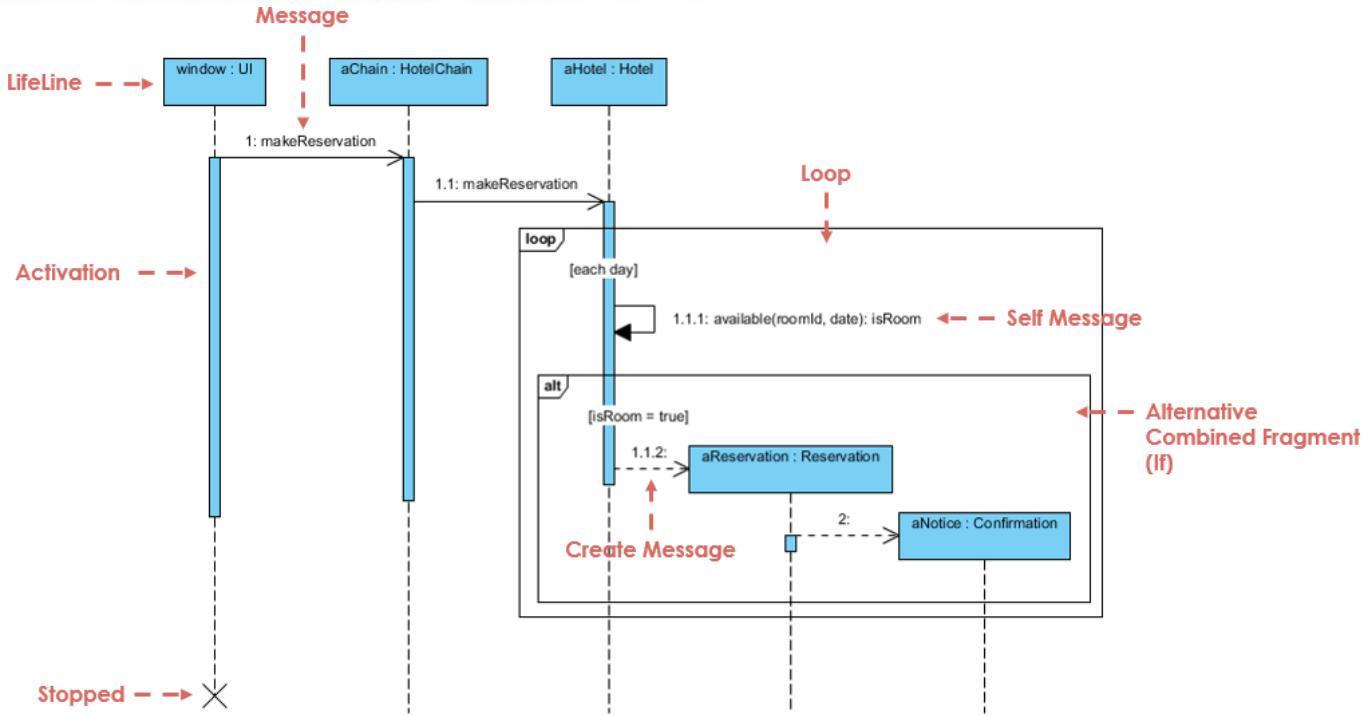
- Gửi message huỷ đối tượng

- Gửi message phản hồi (return/reply message)

Sơ đồ tuần tự


: KhachHang
: ATM
: ATMController
1: Đưa thẻ vào máy()
1.1: Kiểm tra thẻ hợp lệ()
Thẻ hợp lệ
Màn hình nhập mã PIN
2: Nhập mã PIN()
2.1: Kiểm tra mã PIN()
Mã PIN hợp lệ
Màn hình chọn giao tác
3: Chọn xem số dư tài khoản()
3.1: Truy vấn số dư()
Số dư tài khoản
Hiển thị số dư tài khoản
4: Bấm huỷ giao tác()
<<destroy>>
5: Nhã thẻ và kết thúc giao dịch()

Sơ đồ tuần tự



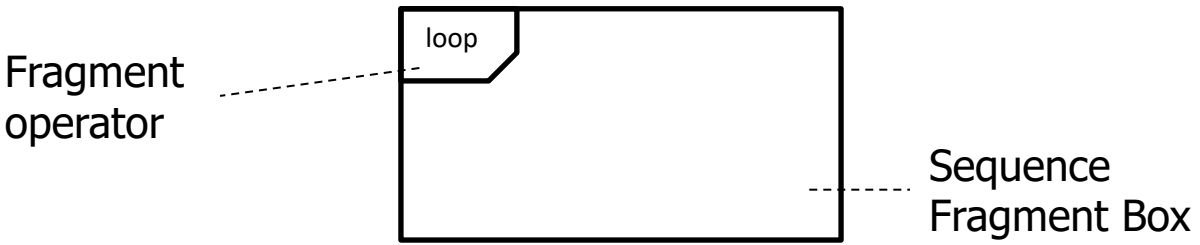
Nguồn: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/

Bài tập vẽ sơ đồ tuần tự

- **Chức năng thanh toán** bắt đầu khi người dùng truy cập vào trang giỏ hàng và click vào nút thanh toán.
- Hệ thống tiến hành kiểm tra số lượng hàng trong kho hàng có đáp ứng được hoá đơn hay không.
- Nếu đủ hàng thì sẽ tiến hành tính tổng hoá đơn hiển thị cho người dùng và hỏi xác nhận thanh toán.
- Nếu người dùng xác nhận, hệ thống liên lạc với paypal tiến hành thanh toán đơn hàng, nếu đơn hàng được thanh toán thành công thì hệ số sẽ ghi nhận đơn hàng và cập nhật số lượng hàng trong kho.

Sơ đồ tuần tự

- UML 2.0 giới thiệu thêm sequence fragment, ký hiệu như sau:



- Chức năng của một sequence fragment được quy định tên ở góc trên trái bao gồm: ref, assert, loop, break, alt, opt, neg.

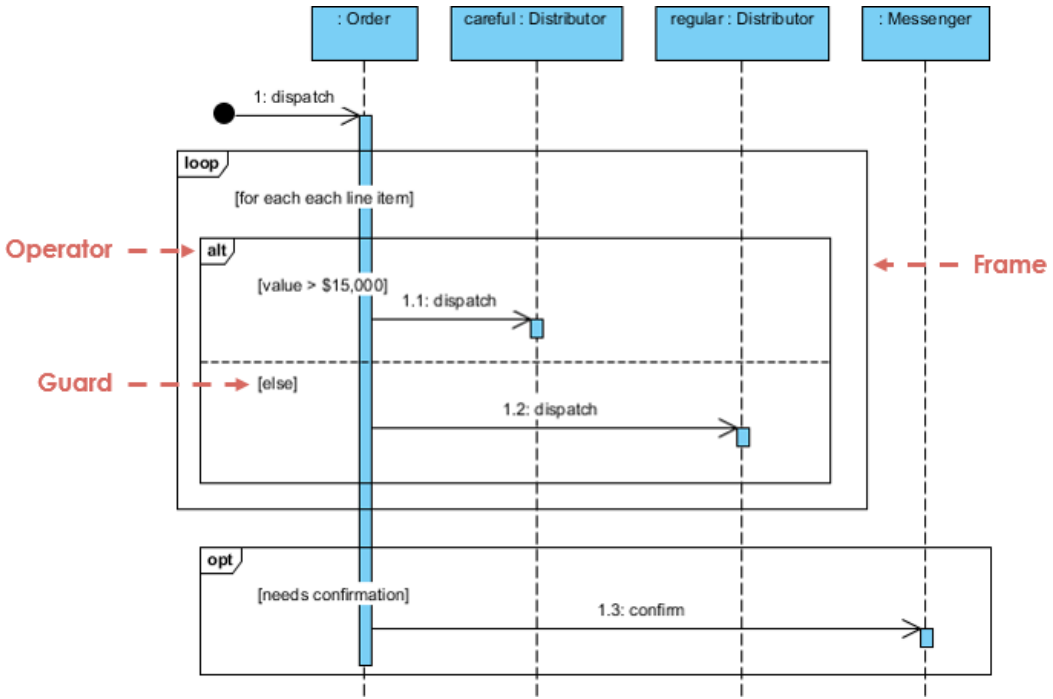
Sơ đồ tuần tự

- alt (alternative mutiple fragments): thực thi khi một điều kiện nào đó đúng.
- opt (optional): fragment được thực thi nếu điều kiện nào đó đúng.
- par (parallel): mỗi fragment được chạy song song.
- loop: fragment được thực thi nhiều lần, guard chỉ định số lần lặp.

Sơ đồ tuần tự

- region (critical region): fragmennt chỉ một thread được thi thi một lần.
- neg (negative): fragment chỉ định các tương tác không hợp lệ.
- ref (reference): chỉ định một tương tác định nghĩa trong lược đồ khác.
- sd (sequence diagram): sử dụng để bọc toàn bộ sơ đồ tuần tự.

Sơ đồ tuần tự



Nguồn: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-sequence-diagram/

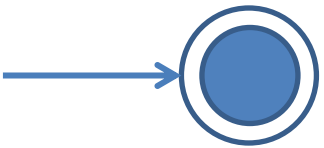
Bài tập sơ đồ tuần tự

- Chức năng nhân viên bán hàng tại siêu thị bán lẻ
  - Khách hàng chọn hàng tại các kệ xong gặp nhân viên quầy thu ngân để thanh toán tiền. Nhân viên dùng máy quét mã vạch để đọc mã mặt hàng, khi đó trên màn hình sẽ hiển thị thông tin mã, tên, xuất xứ mặt hàng, đơn giá mặt hàng, số lượng mặc định là 1 (nhân viên được nhập cập nhật số lượng này). Đối với các sản phẩm chưa định lượng được như rau củ, thịt, trái cây thì nhân viên dùng cân điện tử để cân ký và nhập vào hệ thống.
  - Sau khi nhận được tiền từ khách hàng, nhân viên nhập số tiền đó vào hệ thống để lưu trữ, hệ thống sẽ tính số tiền dư thối lại cho khách hàng, ghi nhận đơn hàng và in ra hoá đơn.

Sơ đồ hoạt động

- Sơ đồ hoạt động (Activity Diagram) biểu diễn các hoạt động tạo nên quy trình hệ thống và luồng điều khiển từ hoạt động này đến hoạt động khác.
- Trạng thái bắt đầu

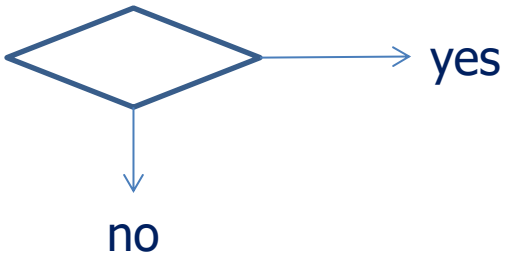
- Trạng thái kết thúc



Sơ đồ hoạt động

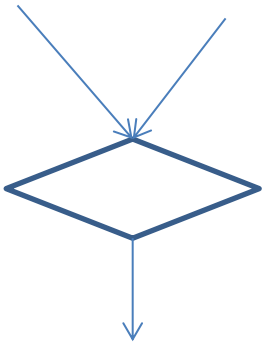
- Mỗi hoạt động đại diện bằng hình chữ nhật bo tròn 4 góc, luồng hoạt động thể hiện bằng mũi tên.

- Điều kiện thể hiện bằng hình thoi



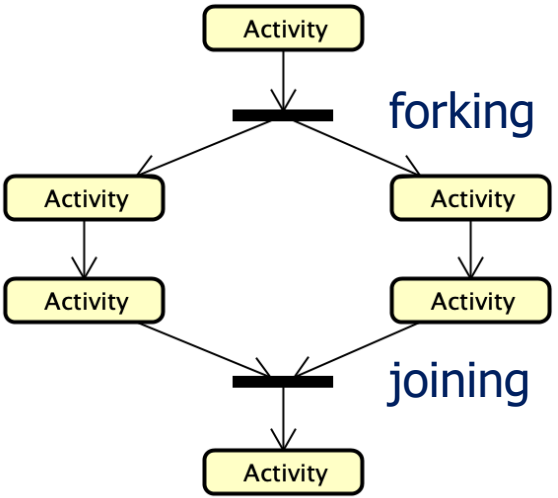
Sơ đồ hoạt động

- Các hoạt động không thực thi đồng thời cần được hợp lại, ta có thể sử dụng ký hiệu sau

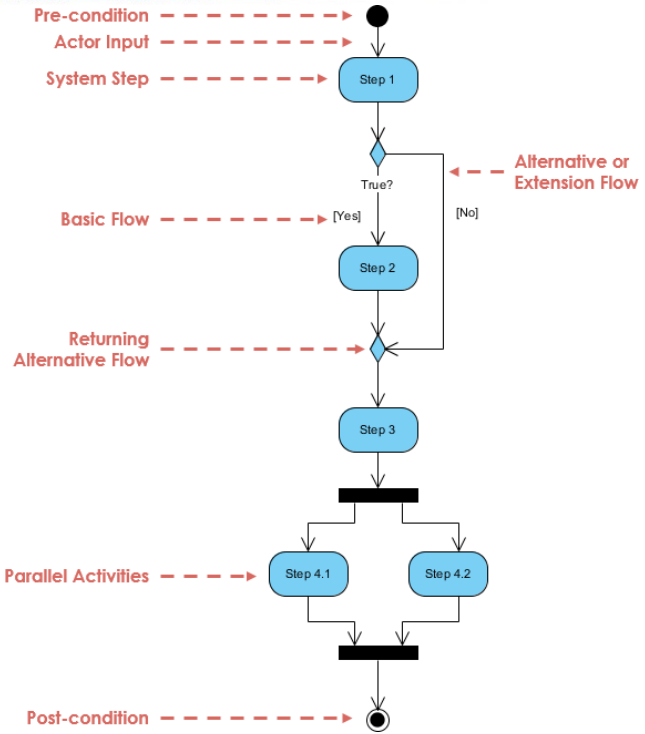


Sơ đồ hoạt động

- Nút forking dùng chia luồng điều khiển thành nhiều luồng xử lý đồng thời.
- Nút joining dùng kết hợp nhiều luồng xử lý đồng thời thành một luồng duy nhất.



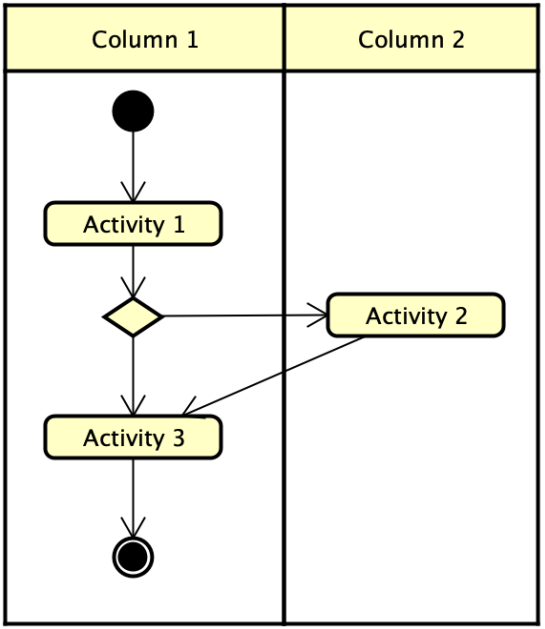
Sơ đồ hoạt động



Nguồn: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/

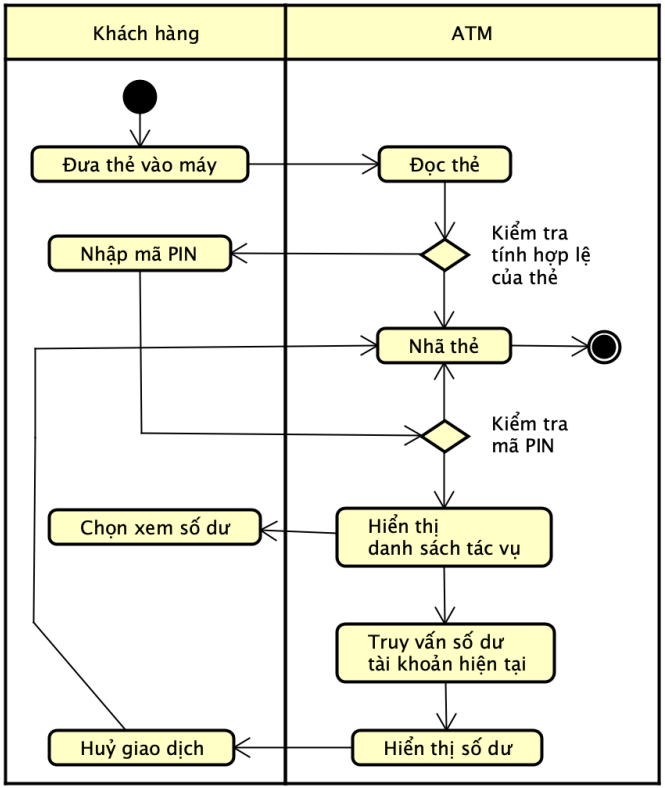
Sơ đồ hoạt động

- Nhóm các hoạt động liên quan thành cột



Sơ đồ hoạt động

- Ví dụ



Bài tập vẽ sơ đồ hoạt động

- Người dùng truy cập vào trang giỏ hàng:
  - Có thể cập nhật số lượng từng món hàng trong giỏ bằng cách nhập vào số lượng mới vào ô nhập liệu số lượng của từng món hàng, rồi bấm nút cập nhật.
  - Hoặc người dùng có thể bấm vào nút thanh toán giỏ hàng, khi đó hệ thống yêu cầu người dùng phải đăng nhập. Sau khi đăng nhập thành công, hệ thống sẽ liên lạc đến stripe để tiến hành thanh toán trực tuyến cho người dùng. Nếu stripe trả về thanh toán thành công thì đơn hàng được ghi nhận vào hệ thống, ngược lại sẽ báo lỗi thanh toán.

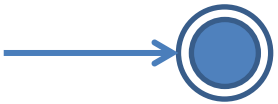
Sơ đồ trạng thái

- Sơ đồ trạng thái (State Diagram) biểu diễn các trạng thái của một hệ thống hoặc một đối tượng duy nhất của lớp và các chuyển đổi từ trạng thái này sang trạng thái khác..
  - states các trạng thái.
  - transitions các chuyển đổi từ trạng thái này sang trạng thái khác.
  - events các sự kiện dẫn đến sự dịch chuyển trạng thái.
  - actions các hành động là kết quả của việc chuyển đổi trạng thái.

Sơ đồ trạng thái

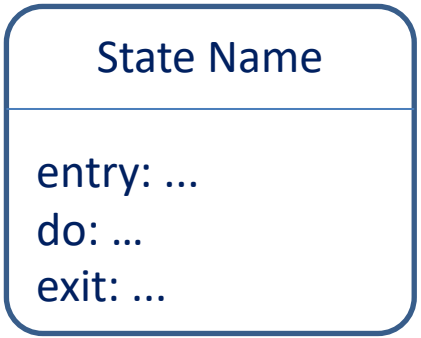
- Mỗi lược đồ trạng thái có duy nhất một trạng thái bắt đầu và có thể có nhiều trạng thái kết thúc.
- Trạng thái bắt đầu

- Trạng thái kết thúc



Sơ đồ trạng thái

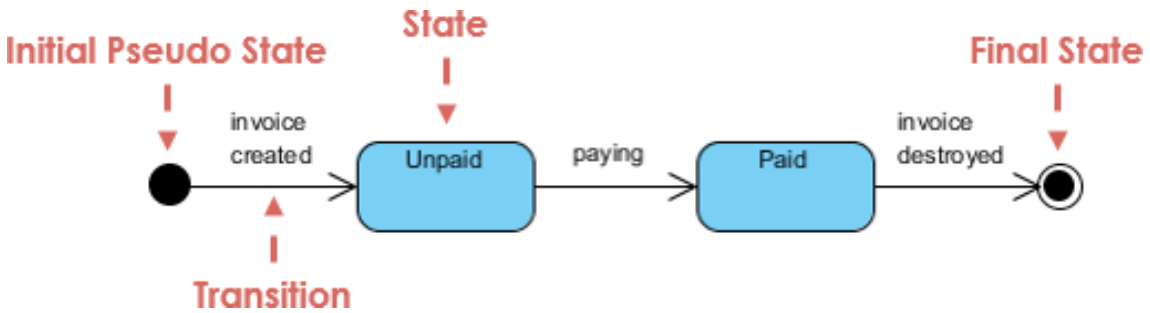
- Mỗi trạng thái được biểu diễn bằng hình chữ nhật bo tròn 4 góc giống như sau:



Sơ đồ trạng thái

- Dùng mũi tên để chuyển đổi (transition) từ trạng thái này sang trạng thái khác.

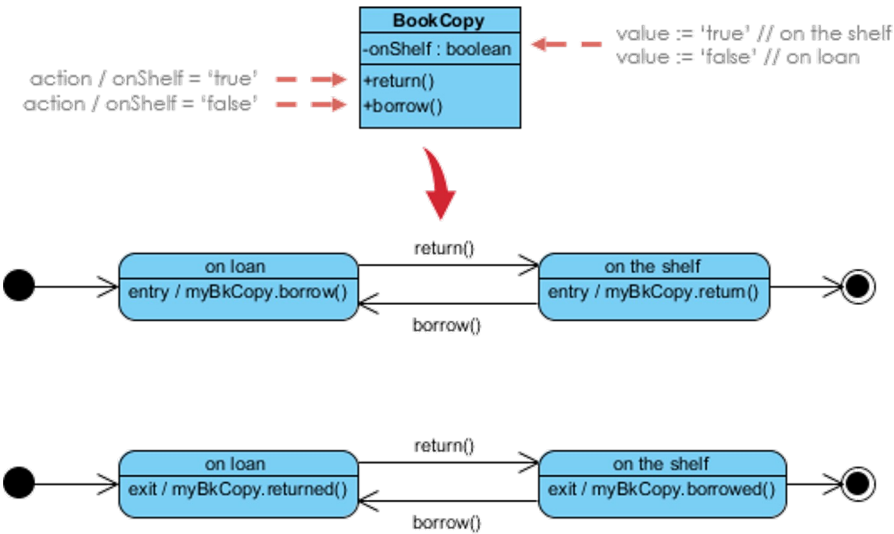
Sơ đồ trạng thái



Nguồn: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-state-machine-diagram/

Sơ đồ trạng thái

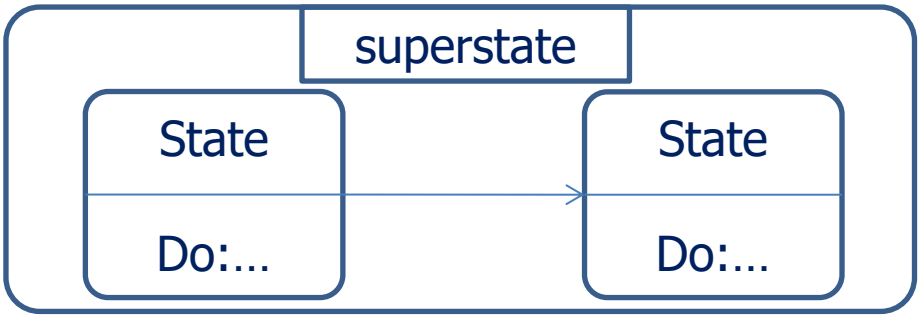
- Một sơ đồ trạng thái được sử dụng mô hình hoá các hành vi động của lớp (class).



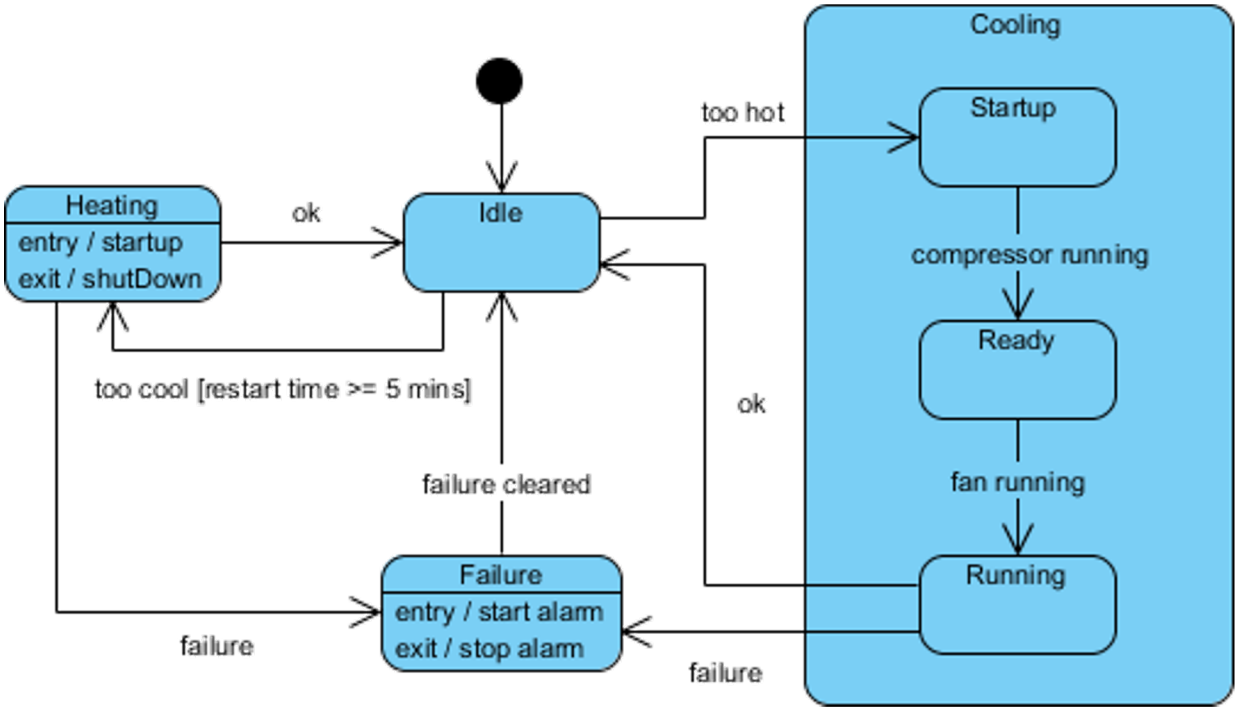
Nguồn: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-state-machine-diagram/

Sơ đồ trạng thái

- Các hệ thống lớn có rất nhiều trạng thái khác nhau, ta cần ẩn đi chi tiết trong mô hình bằng cách sử dụng ký hiệu superstate để đóng gói nhiều trạng thái.
- superstate cũng được xem là một trạng thái, nhưng có thể mở rộng để xem chi tiết trạng thái con bên trong nó.



Sơ đồ trạng thái



Nguồn: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-state-machine-diagram/

Bài tập sơ đồ trạng thái

Một hệ thống quản lý học tập có chức năng cho phép đăng bài viết, một bài viết khi đăng mới chỉ được phép cập nhật hoặc xóa trong vòng 15 phút kể từ lúc submit đăng bài, sau khoảng thời gian này bài viết không được phép chính sửa hay xóa nửa và bài viết sẽ tự động được xuất bản trên hệ thống để người khác có thể đọc.

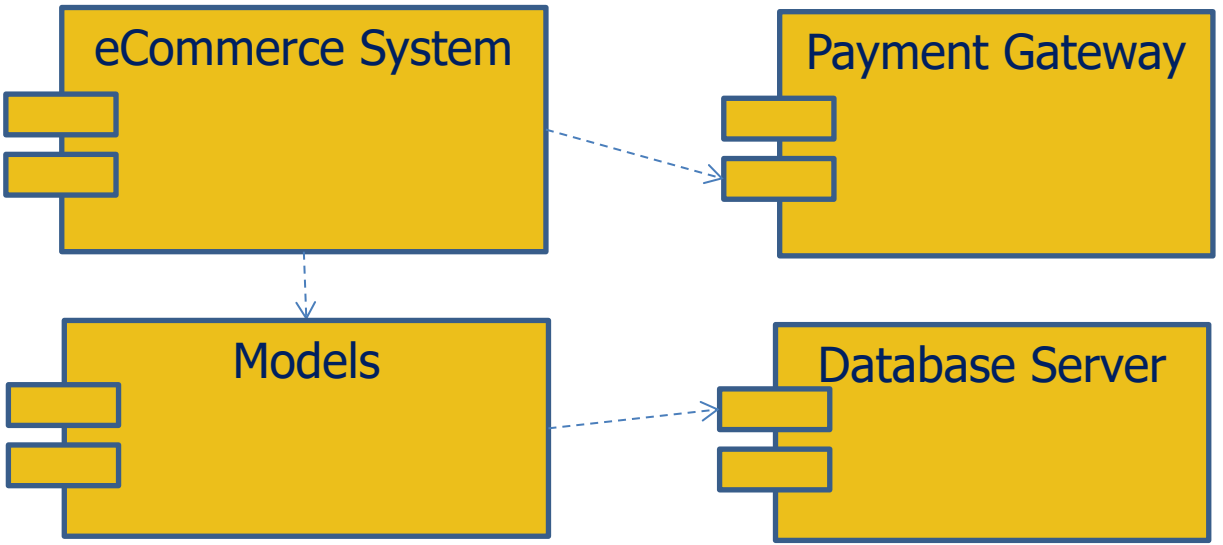
Ngoài ra, khi vừa soạn xong bài viết hoặc trong vòng 15 phút từ lúc submit bài viết, tác giả bài viết có quyền bấm nút "Publish" để xuất bản bài viết, và tất nhiên không được xóa hoặc cập nhật bài viết sau khi đã xuất bản. Sau khi một bài viết được xuất bản, tác giả bài viết muốn xóa hoặc cập nhật bài viết cần phải liên hệ với admin thực hiện. Chú ý sau khi admin chỉnh sửa bài viết đã xuất bản, bài viết đó vẫn ở trạng thái xuất bản để người khác đọc.

Sơ đồ thành phần

- Sơ đồ thành phần (Component Diagram) biểu diễn các khía cạnh vật lý của hệ thống hướng đối tượng.
- Sơ đồ phân rã hệ thống thành các thành phần đảm nhiệm một nhiệm vụ nào đó trong hệ thống và biểu diễn sự phụ thuộc giữa chúng trong hệ thống.

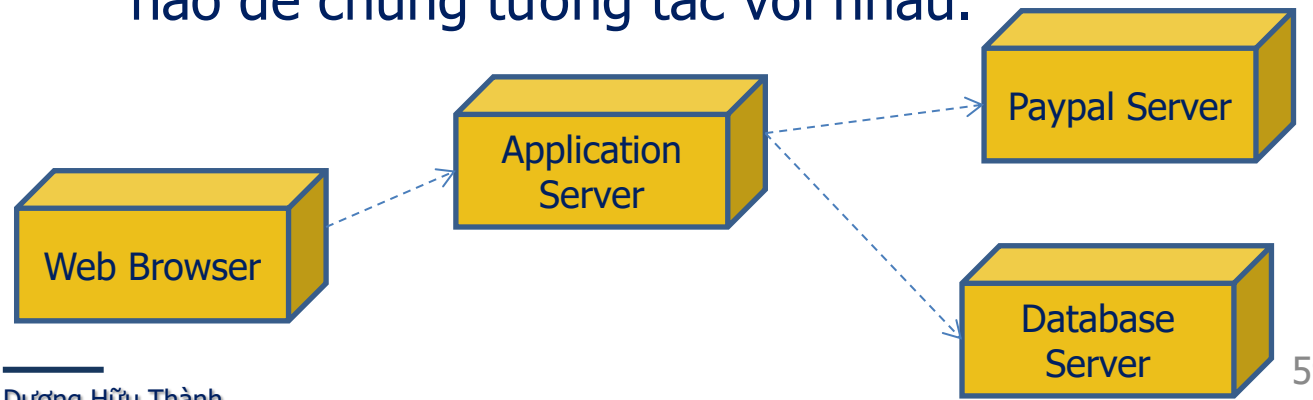
Sơ đồ thành phần

- Ví dụ



Sơ đồ triển khai

- Sơ đồ triển khai (Deployment Diagram) biểu diễn cách một hệ thống được triển khai cụ thể trong môi trường phần cứng.
- Mục đích hiển thị các thành phần khác nhau của hệ thống cụ thể sẽ chạy ở đâu và làm như thế nào để chúng tương tác với nhau.




Q&A