

Phân tích

cuu duong than cong . com

cuu duong than cong . com



KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN



Tham khảo

“Mastering Object-Oriented Analysis and Design with UML 2.0”

IBM Software Group

cuu duong than cong . com

cuu duong than cong . com

Hiện thực hóa Use-Case Analysis Class

cuu duong than cong . com

cuu duong than cong . com



KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

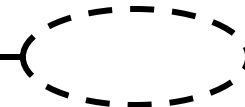
Hiện thực hóa Use-Case

Mô hình Use-Case

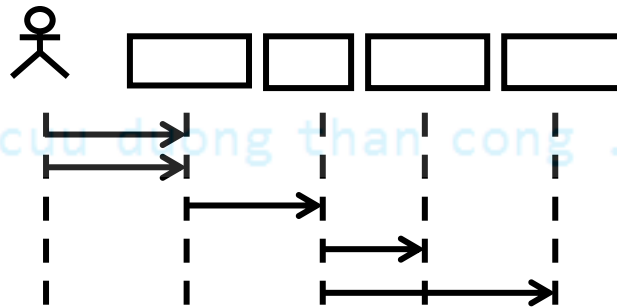


Use Case

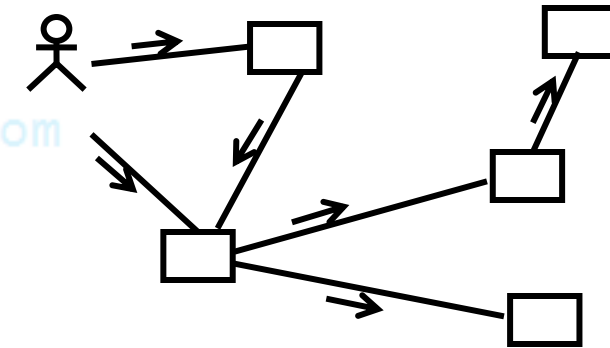
Mô hình thiết kế



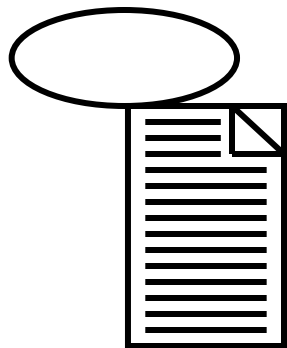
Hiện thực hóa Use-Case



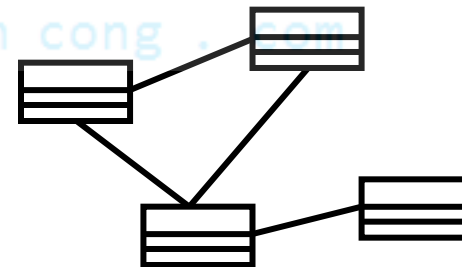
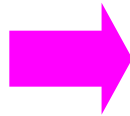
Sơ đồ Sequence



Sơ đồ Communication



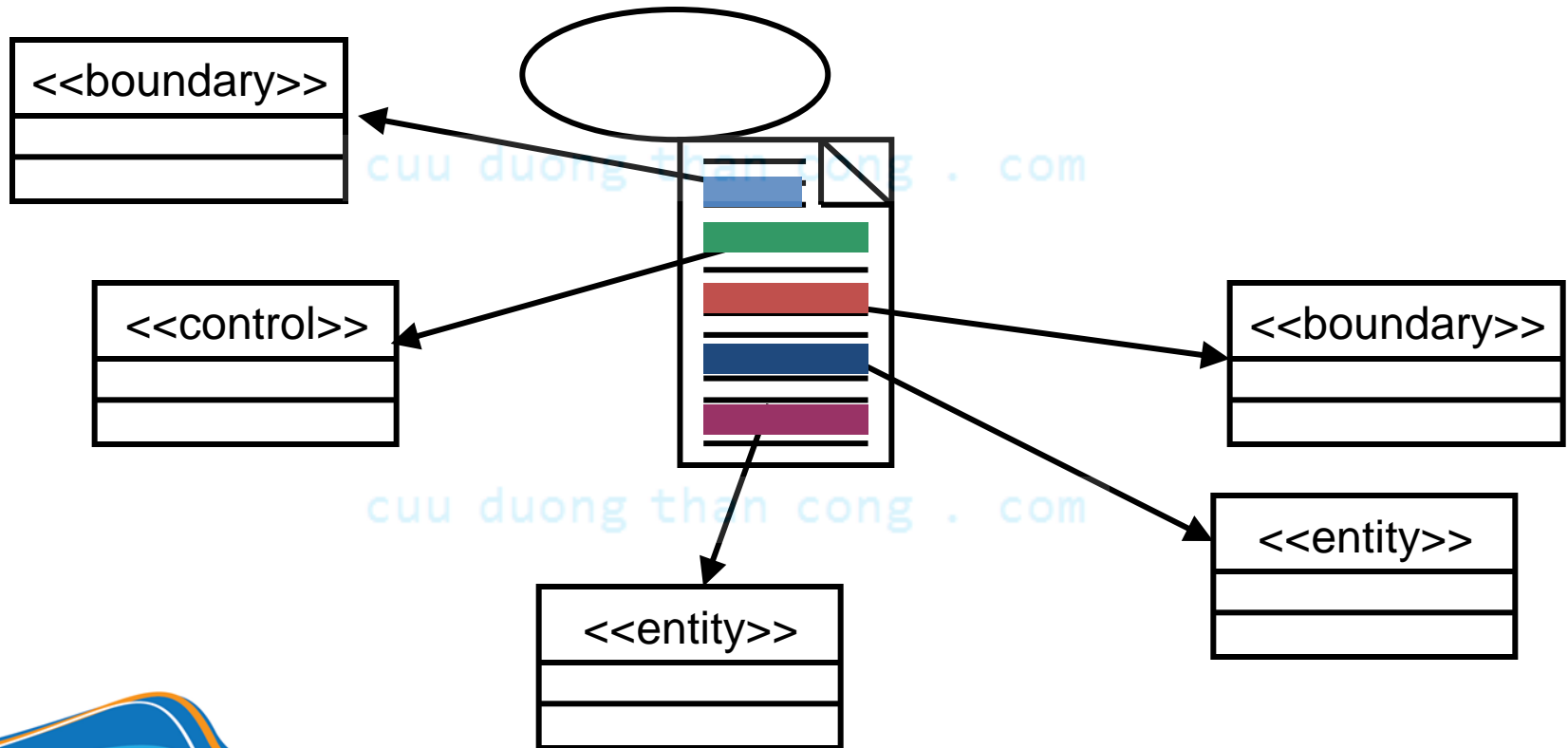
Use Case



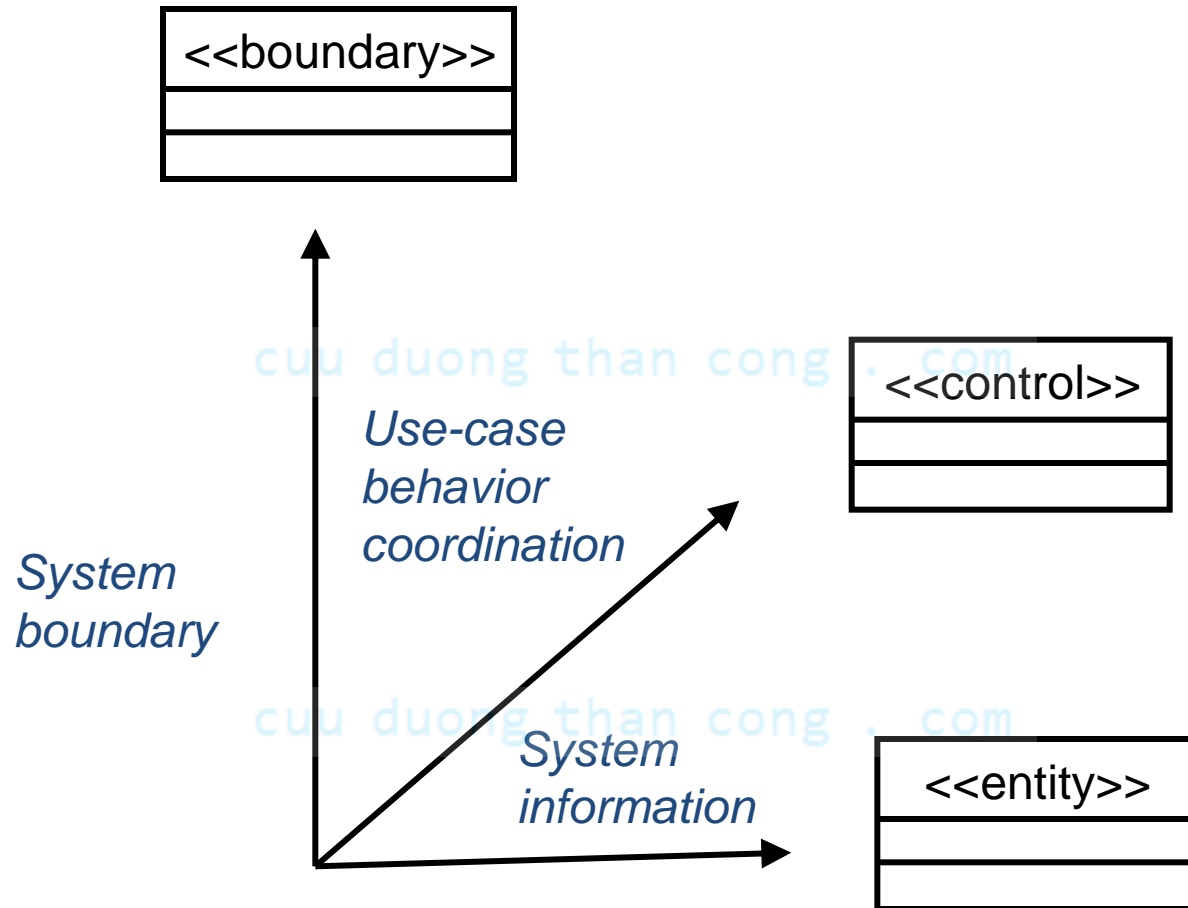
Sơ đồ lớp

Tìm các Class từ Use-Case Behavior

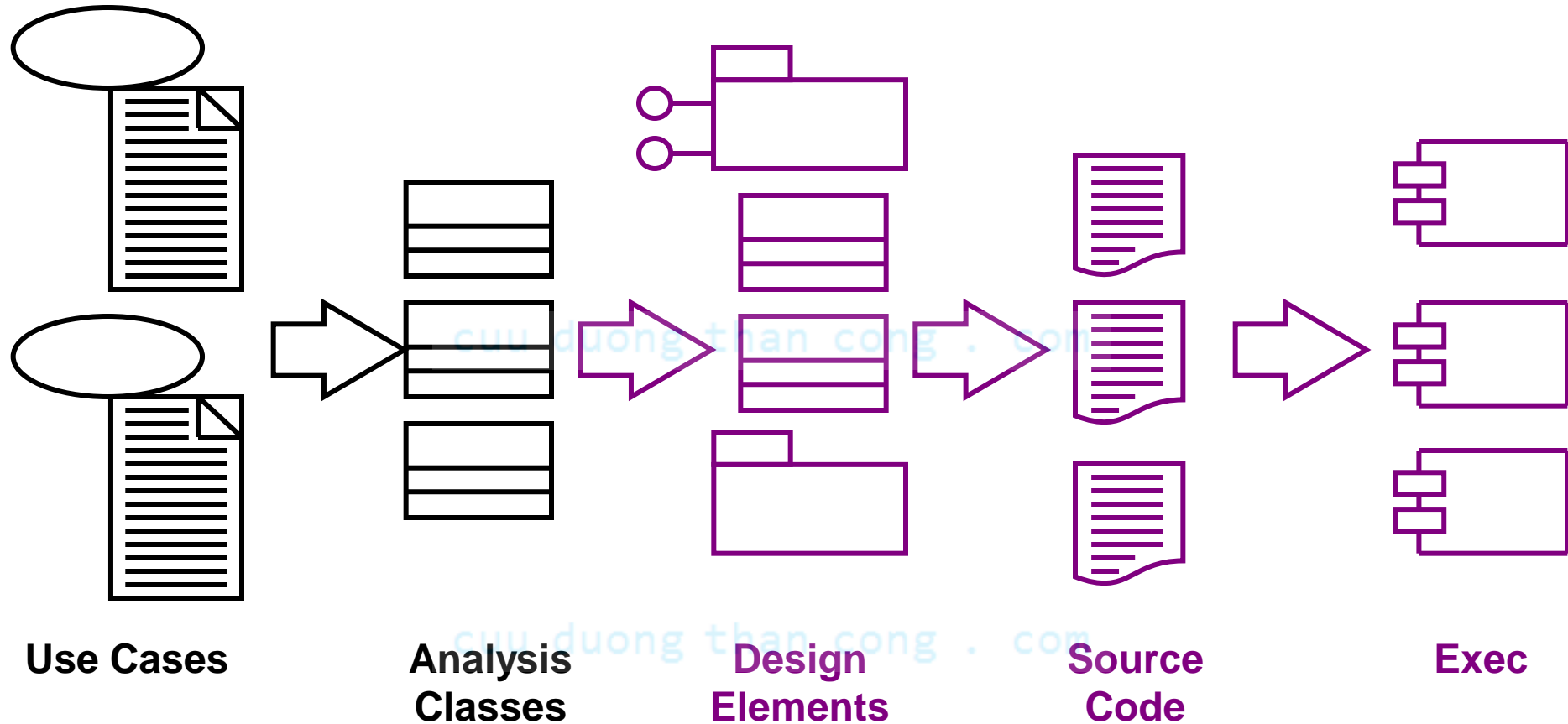
- Toàn bộ hành vi của một use case phải được phân bổ về cho các analysis class



Thế nào là một Analysis Class?



Analysis Class: Bước đầu tiên tiến đến cài đặt

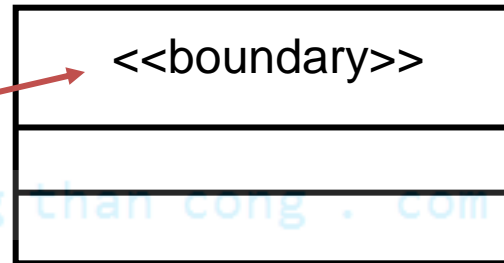


Use-Case Analysis

Thế nào là một Boundary Class?

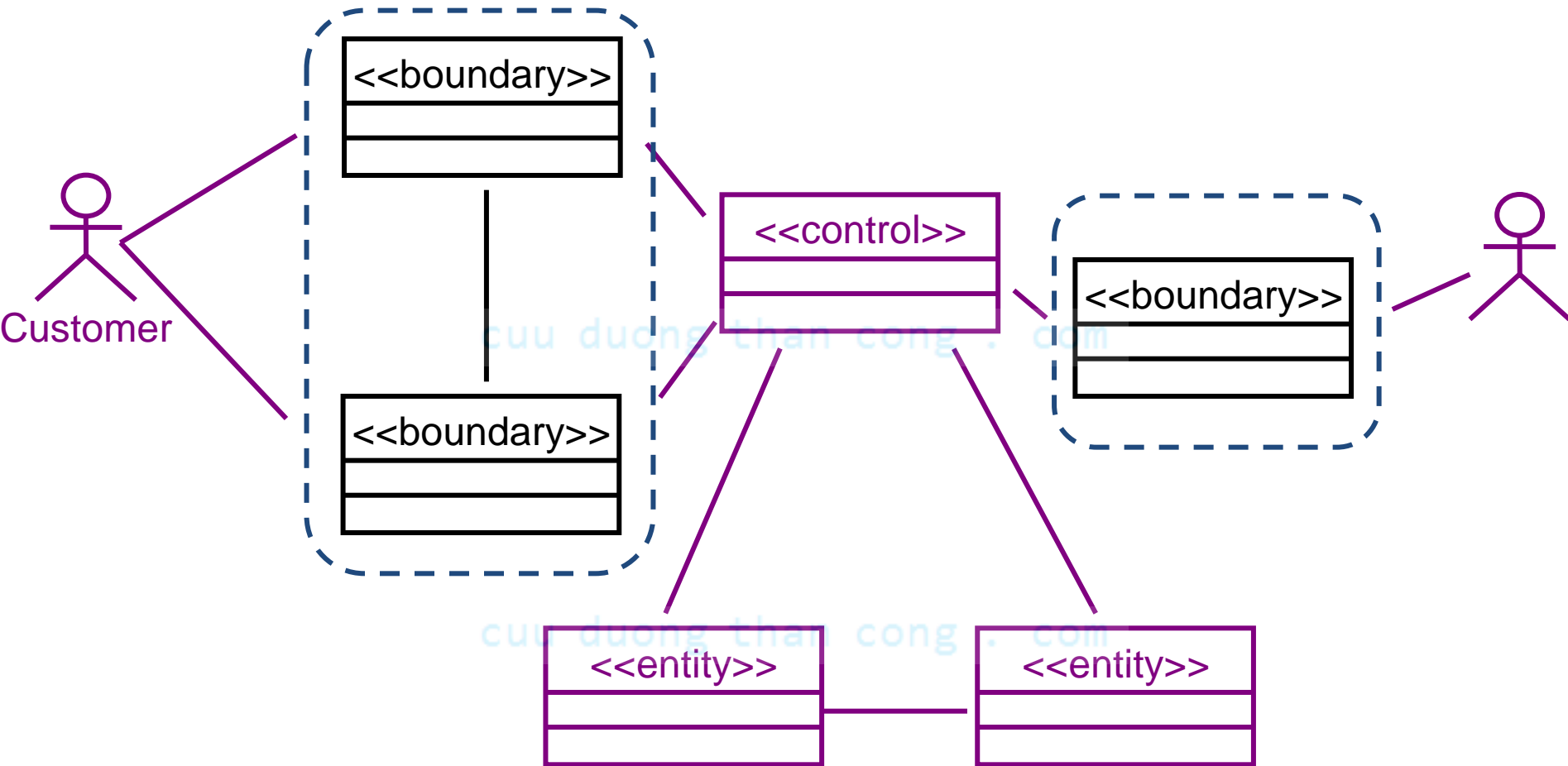
- ☐ Làm trung gian giao tiếp với những gì nằm ngoài hệ thống
- ☐ Một số kiểu
 - ☐ Các User interface class
 - ☐ Các System interface class
 - ☐ Các Device interface class
- ☐ *Một boundary class cho 1 cặp actor/use case*

*Analysis class
stereotype*



Phụ thuộc môi trường

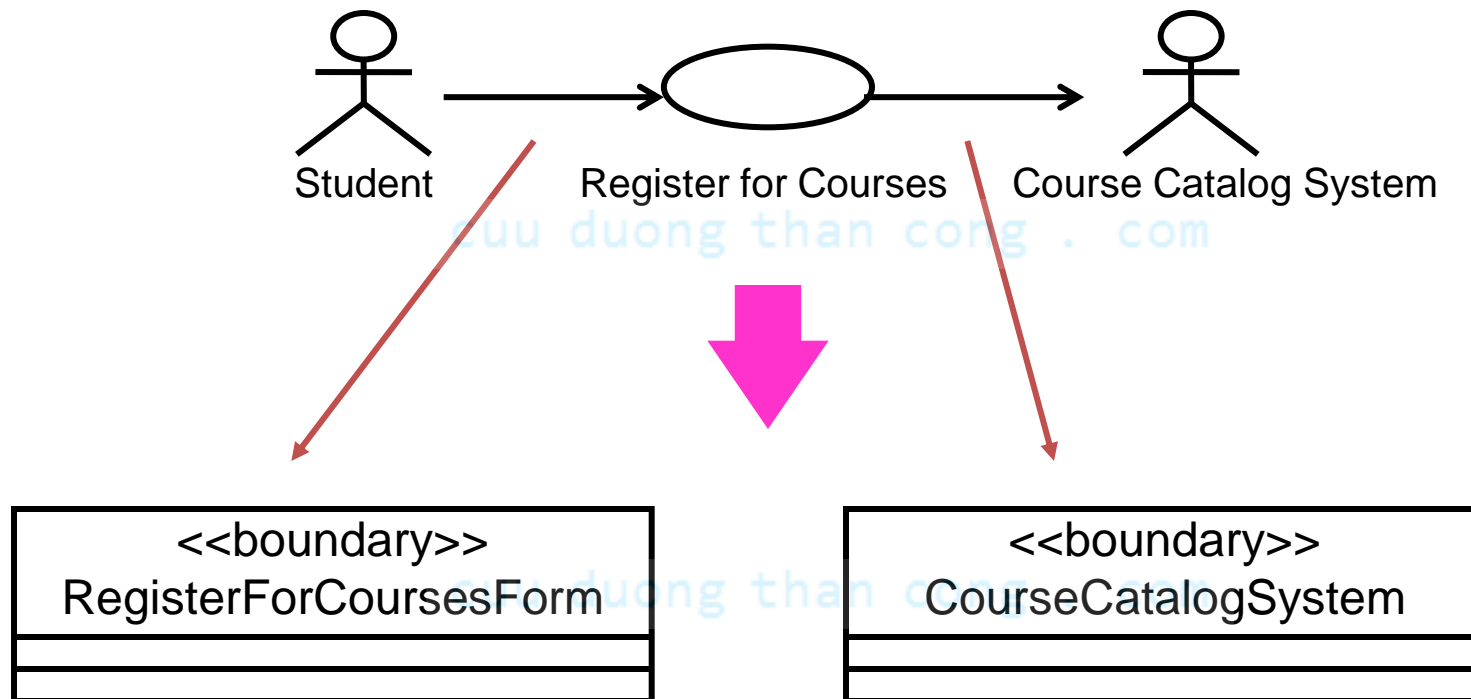
Vai trò của Boundary Class



Mô hình hóa sự tương tác giữa system và môi trường

Ví dụ: Tìm các Boundary Class

- Một boundary class cho 1 cặp actor/use case



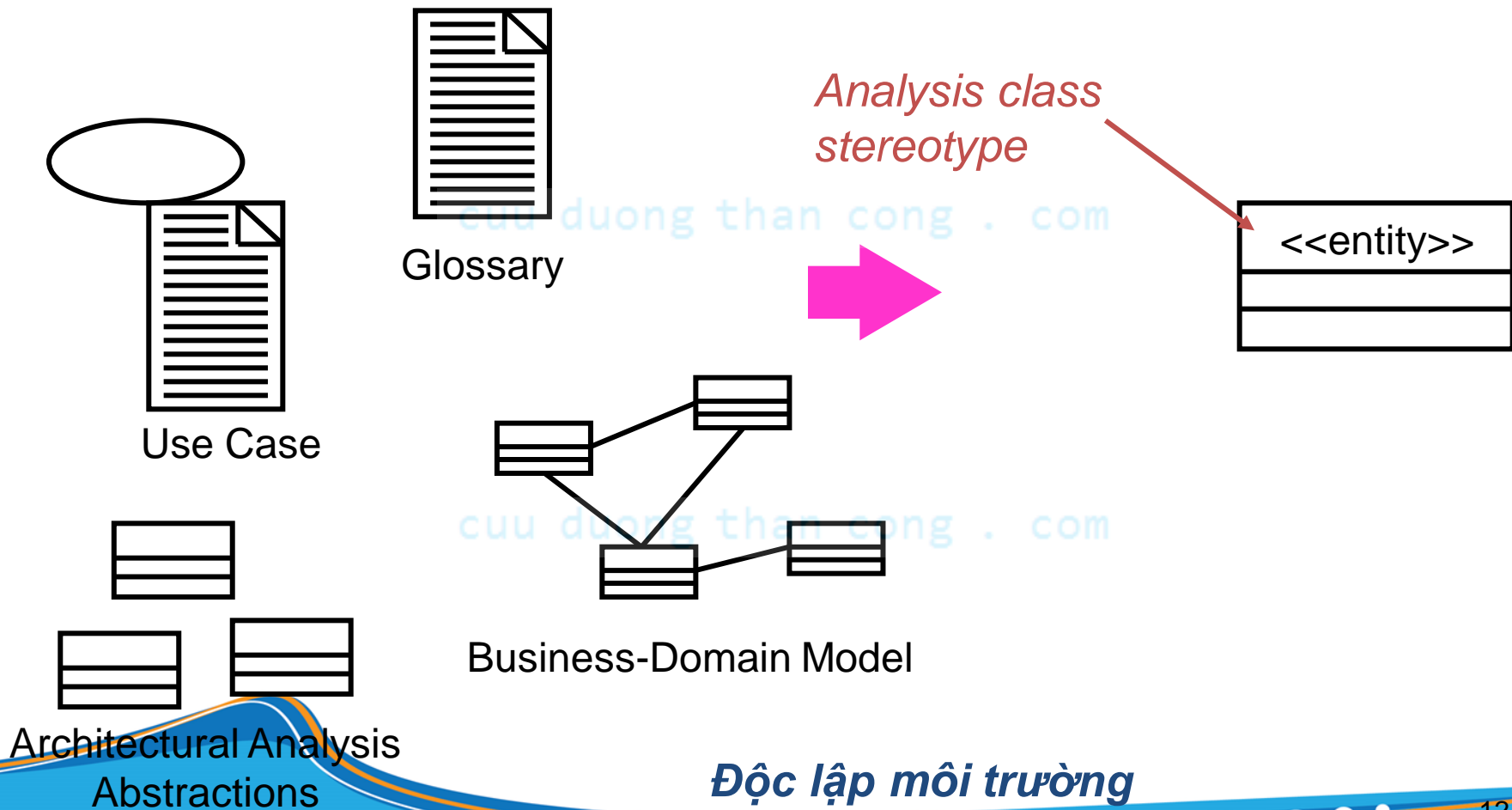
Hướng dẫn: Boundary Class

- Các User Interface Class
 - Tập trung vào những thông tin gì được thể hiện cho người dùng
 - **KHÔNG** tập trung vào các chi tiết UI
- Các System và Device Interface Class
 - Tập trung vào những protocols nào phải định nghĩa
 - **KHÔNG** tập trung vào cách mà các protocol sẽ được cài đặt

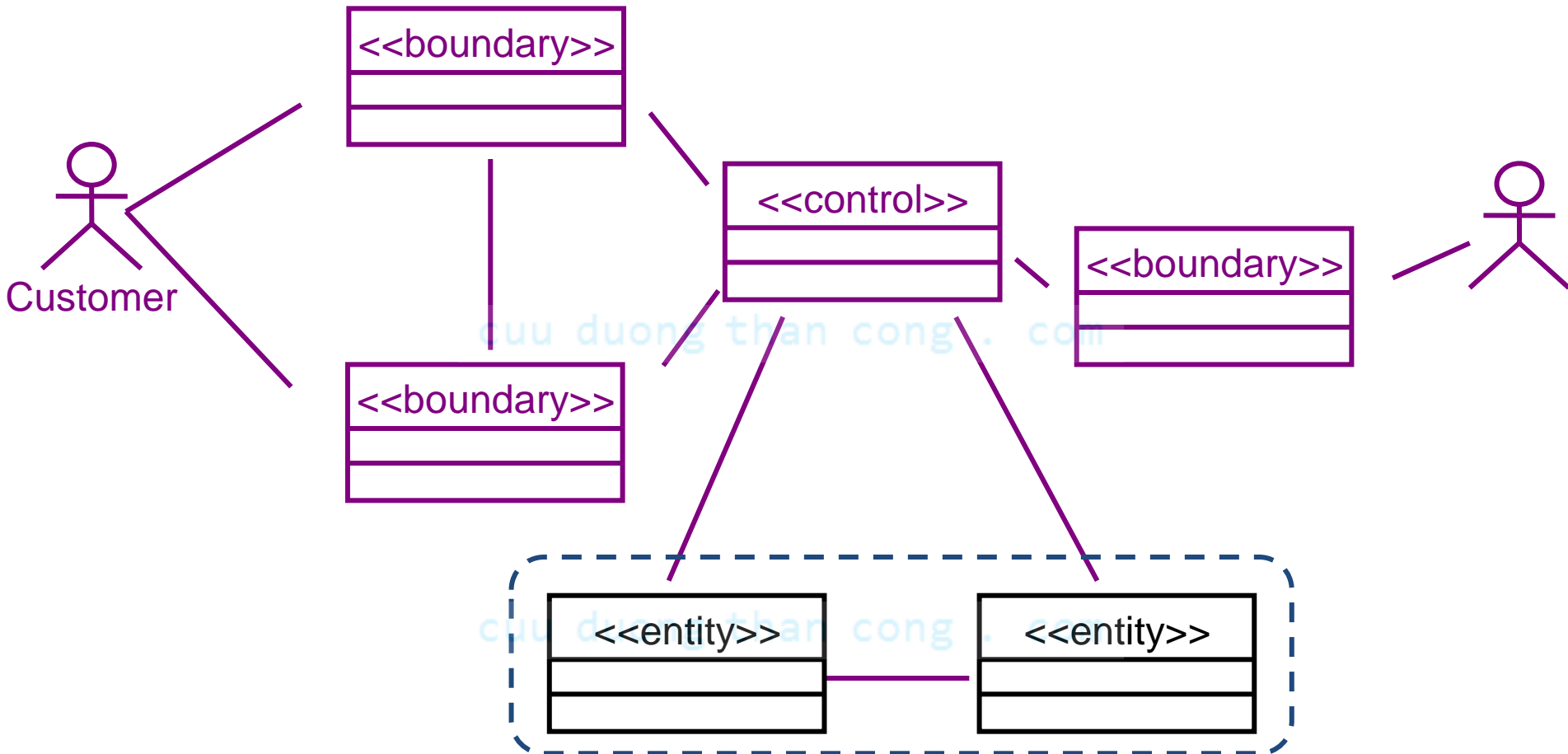
Tập trung vào các nhiệm vụ, chứ không phải chi tiết!

Thế nào là một Entity Class?

- Các trừu tượng hóa then chốt của system



Vai trò của Entity Class



Lưu trữ và quản trị các thông tin trong system

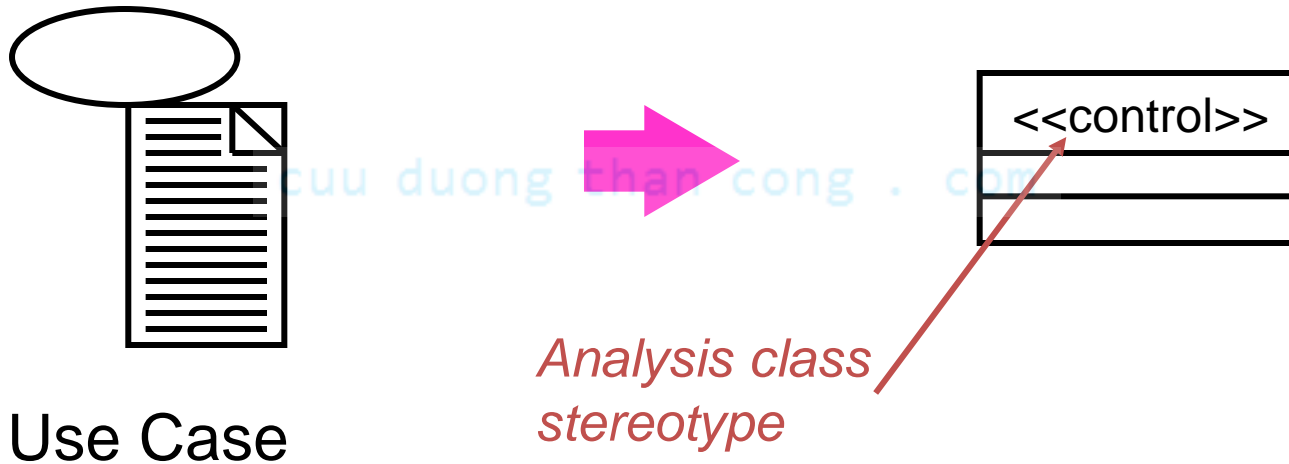
Ví dụ: Tìm các Entity Class

- ☐ Dùng use-case flow of events như input
- ☐ Các trừu tượng hóa then chốt của use case
- ☐ Hướng tiếp cận truyền thống (nouns filtering)
 - ☐ Gạch dưới các cụm danh từ trong flow of events
 - ☐ Loại bỏ các ứng viên dư thừa
 - ☐ Loại bỏ các ứng viên mơ hồ, không rõ ràng
 - ☐ Loại bỏ các actor (ngoài phạm vi)
 - ☐ Loại bỏ các kiến trúc cài đặt
 - ☐ Loại bỏ các attribute (để lại dùng sau)
 - ☐ Loại bỏ các operation

[cuduongthancong . com](http://cuduongthancong.com)

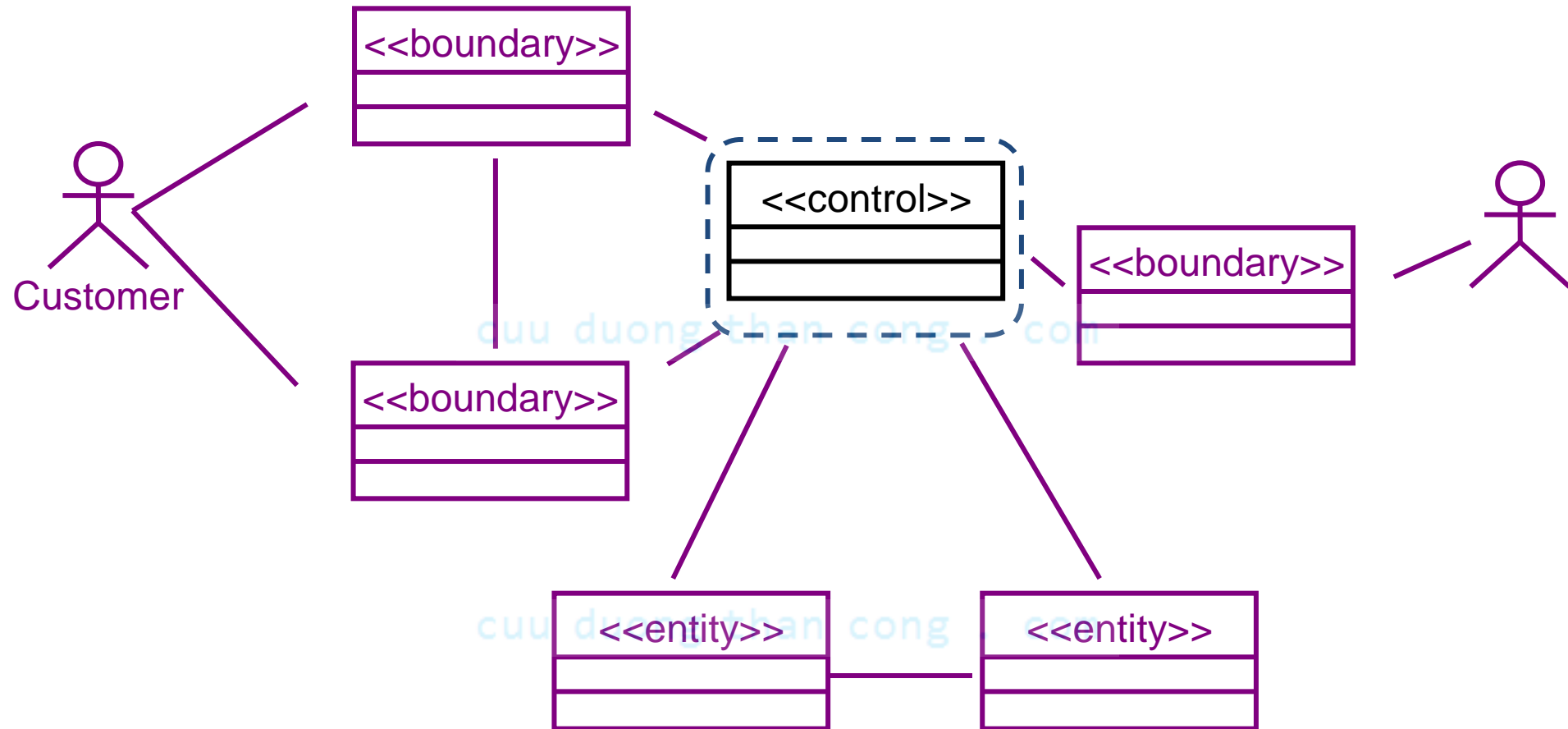
Thế nào là một Control Class?

- ☐ Nhà điều phối các hành vi của Use-case
- ☐ *Chỉ một control class cho một use case*



Phụ thuộc use-case, độc lập môi trường

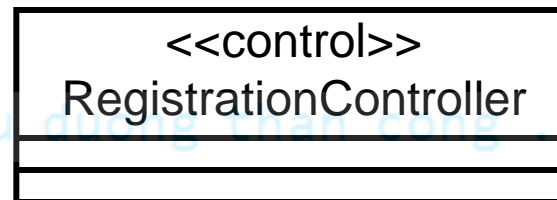
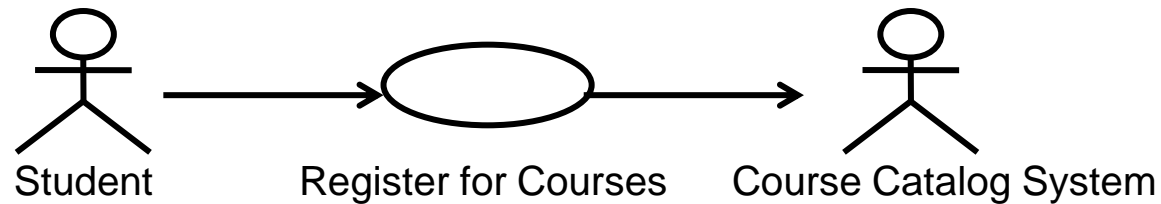
Vai trò của Control Class



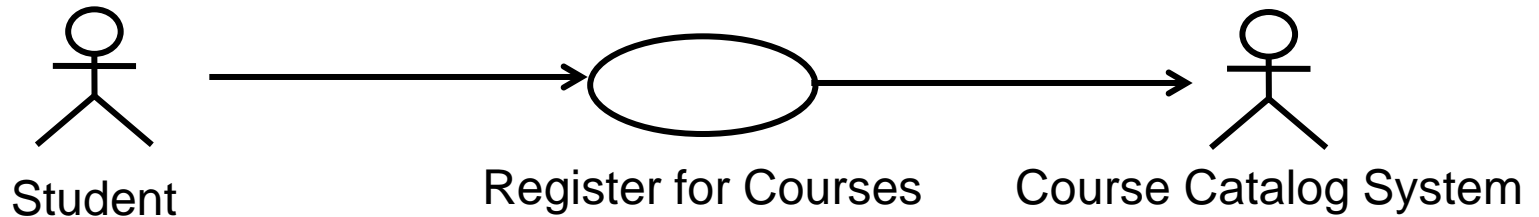
Điều phối các hành vi của use-case

Ví dụ: Tìm các Control Class

- Một control class cho một use case

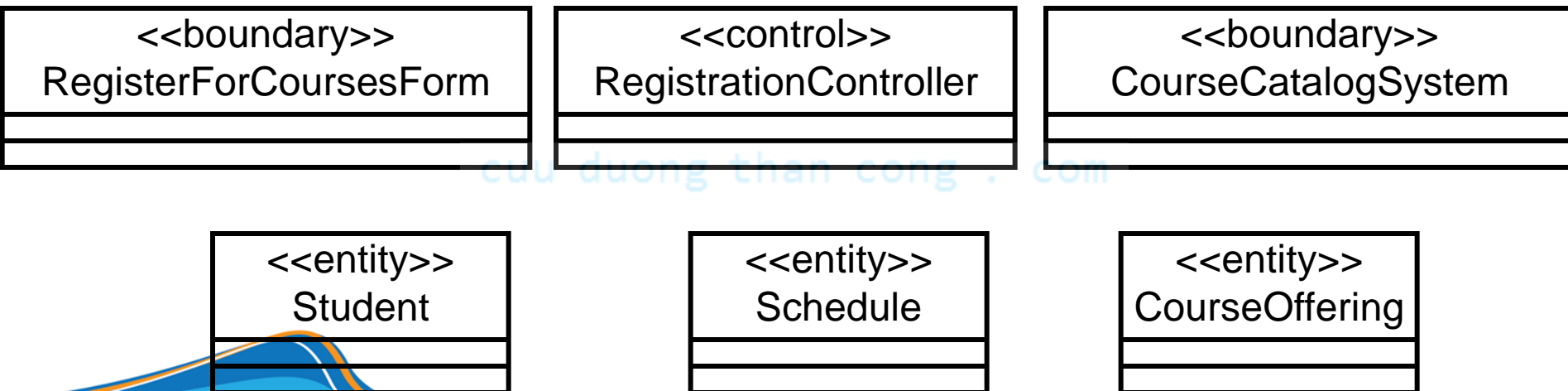


Ví dụ: Tổng kết Analysis Classes



Use-Case Model

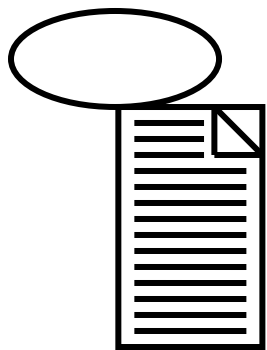
Design Model



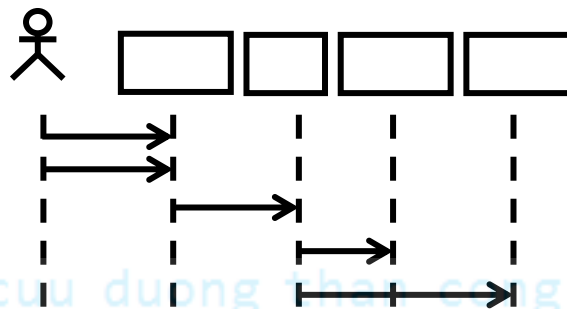
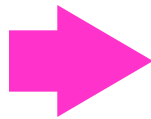
Phân bổ trách nhiệm về các Class

- Đối với mỗi use-case flow of events:
 - Xác định các analysis class
 - Gắn kết các trách nhiệm của use-case cho các analysis class
 - Mô hình hóa tương tác của các analysis class trong interaction diagram

cuu duong than cong . com

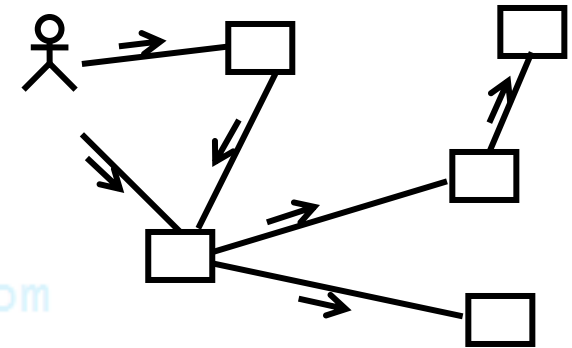


Use Case



sơ đồ Sequence

Use-Case Realization



sơ đồ Communication

Phân bổ trách nhiệm về các Class

- Dùng các A.Class stereotype làm guide:
 - Các Boundary Class
 - Các hành vi liên quan đến sự giao tiếp với actor
 - Các Entity Class
 - Các hành vi liên quan đến dữ liệu được gói trong một abstraction
 - Các Control Class
 - Các hành vi đặc thù cho một use case hoặc một phần rất quan trọng của flow of events

[cuu duong than cong . com](http://cuuduongthanhcong.com)

(còn tiếp)

Phân bổ trách nhiệm về các Class

- Ai có dữ liệu cần cho việc thực hiện nhiệm vụ?
 - Một class có dữ liệu, hãy để nhiệm vụ cùng với dữ liệu
 - Nhiều class có dữ liệu :
 - Hãy để nhiệm vụ trong 1 class và thêm quan hệ với các class khác.
 - Tạo một class mới, để nhiệm vụ trong class mới này, và thêm quan hệ với các class cũ
 - Hãy để nhiệm vụ trong control class, và thêm quan hệ với các class cần để thực hiện nhiệm vụ

cuu duong than cong . com

Sơ đồ Sequence và sơ đồ Communication

cuu duong than cong . com

cuu duong than cong . com



KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Sơ đồ Sequence

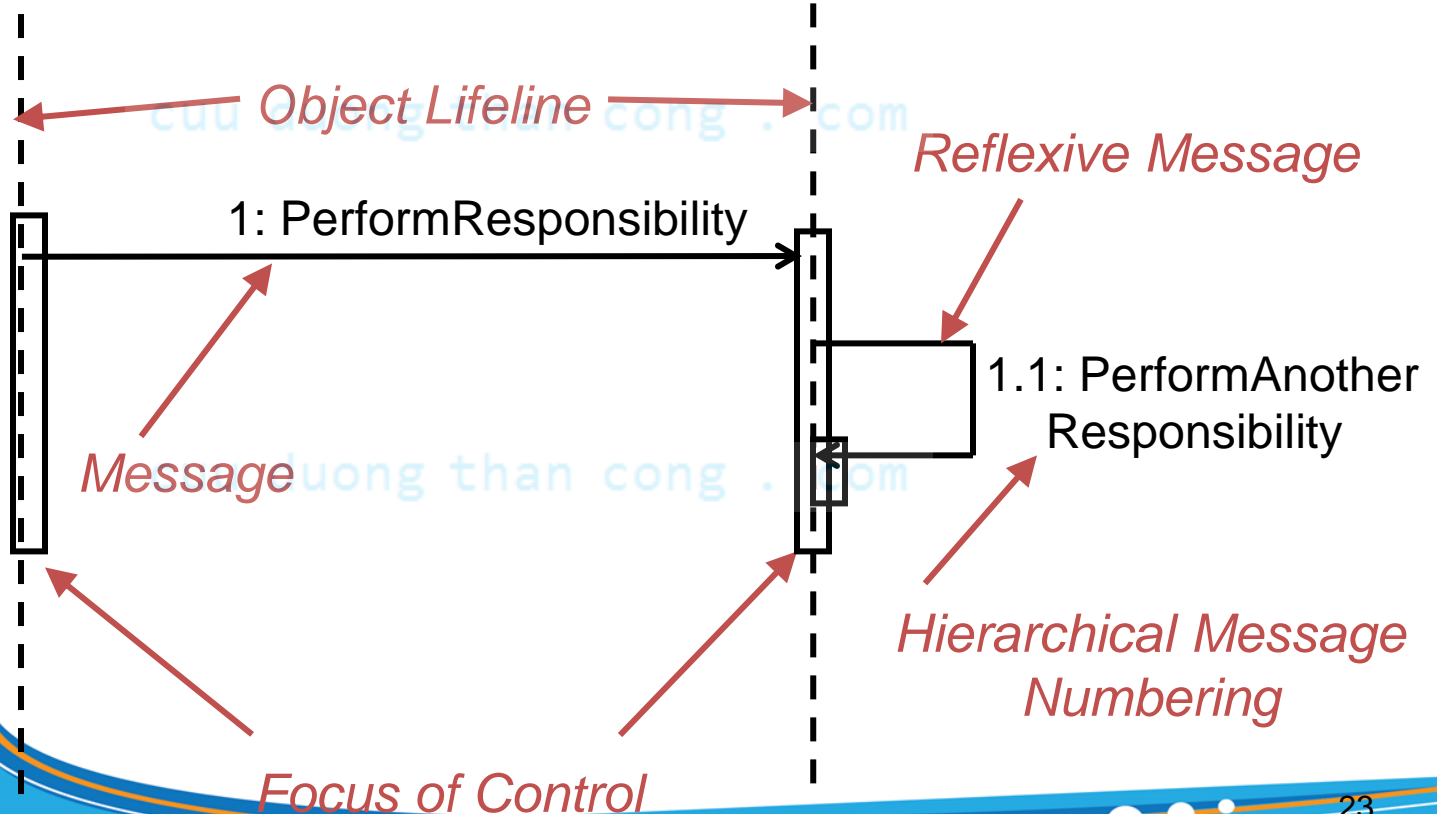
Client Object

Supplier Object

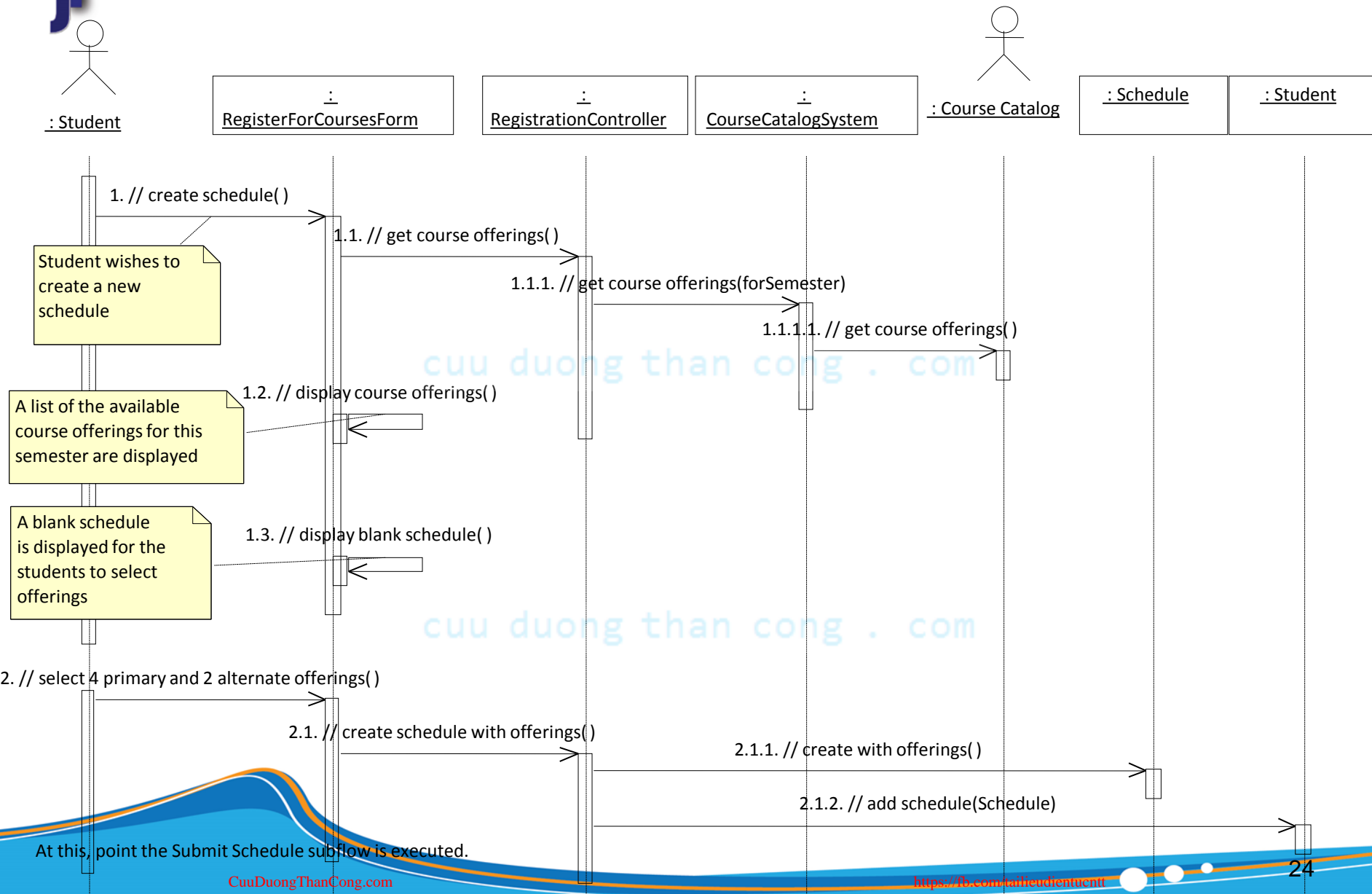
:Client

:Supplier

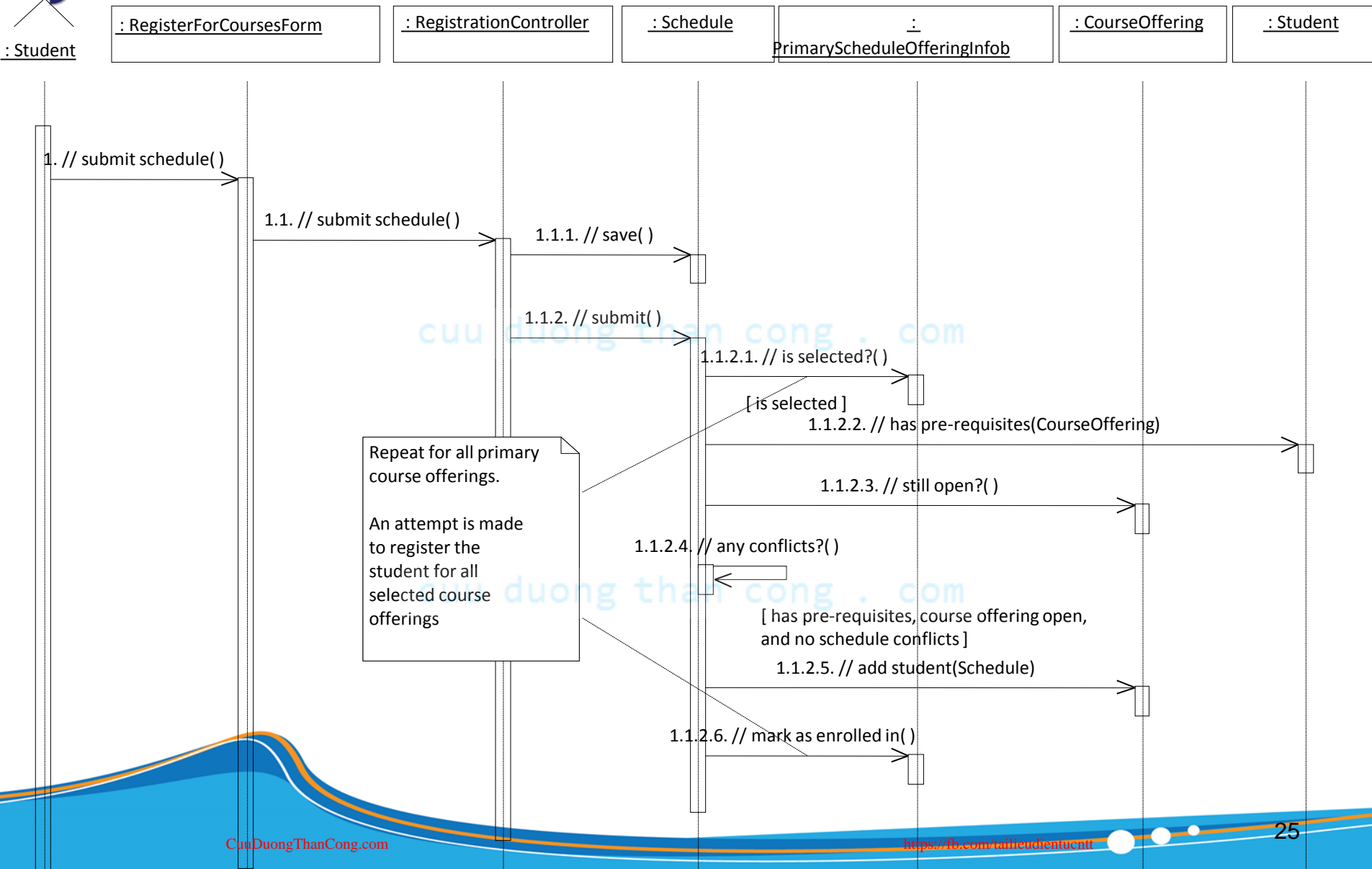
Đây là một
script mẫu



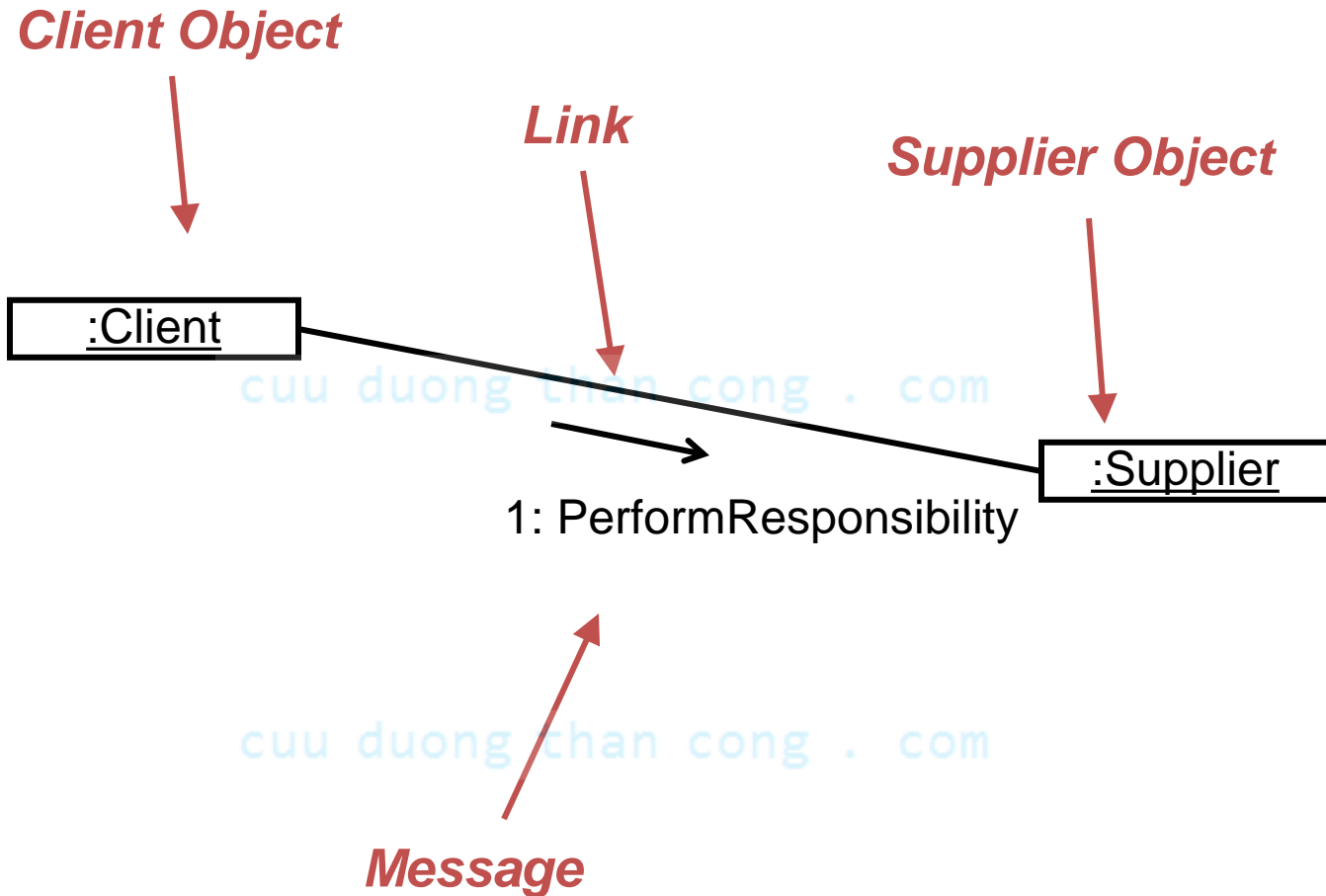
Ví dụ: sơ đồ Sequence



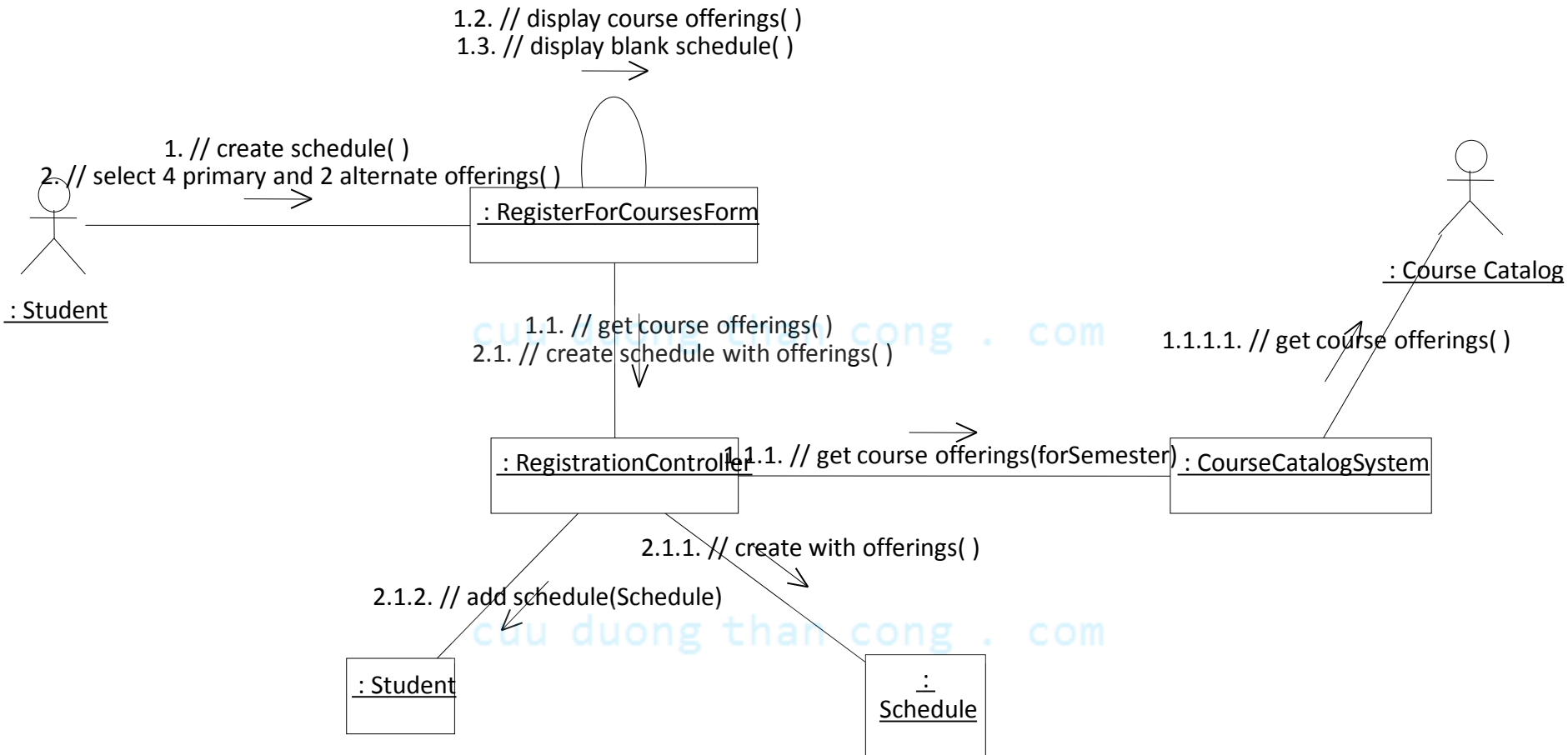
Ví dụ: sơ đồ Sequence



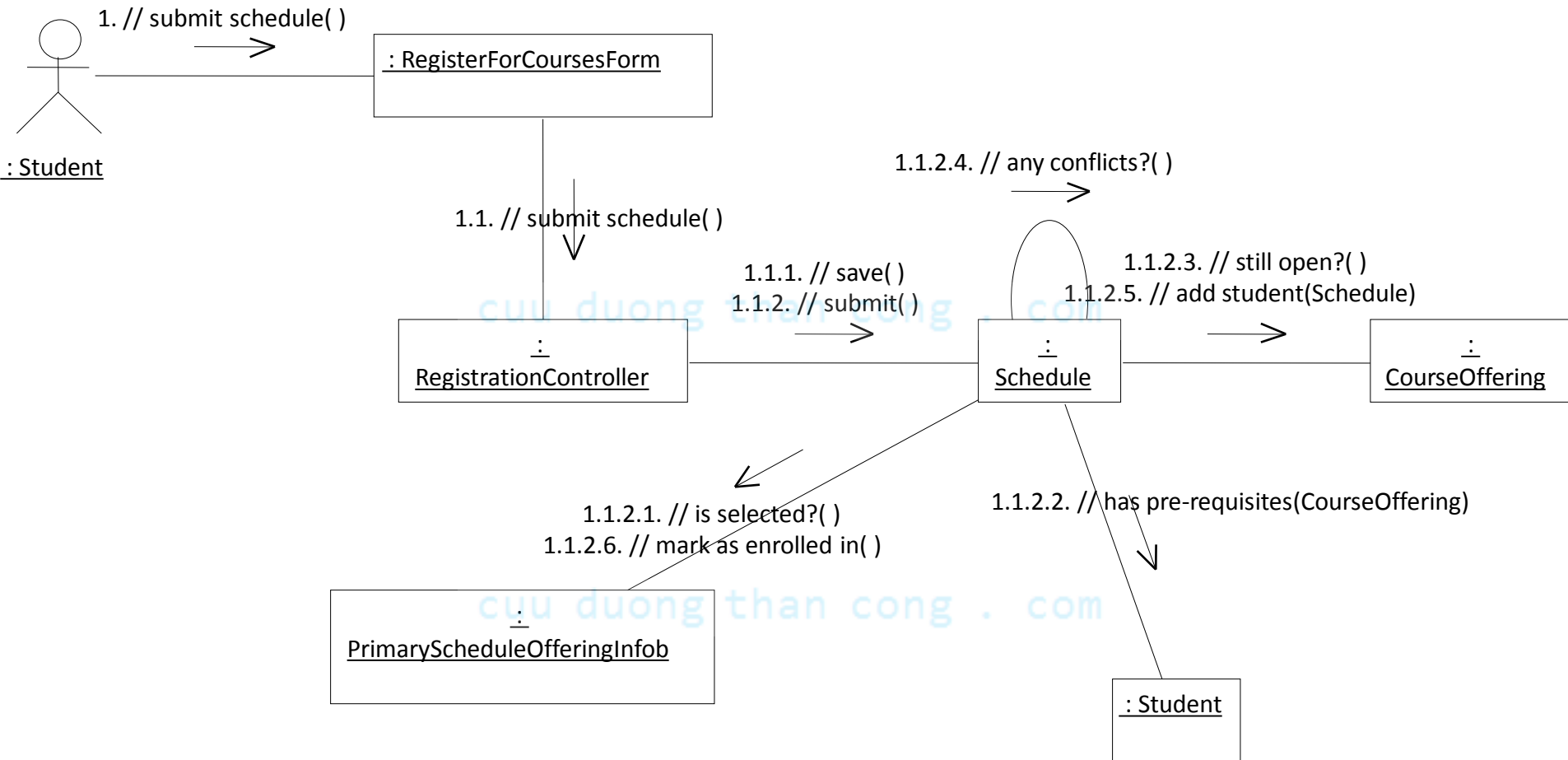
Sơ đồ Communication



Ví dụ: sơ đồ Communication

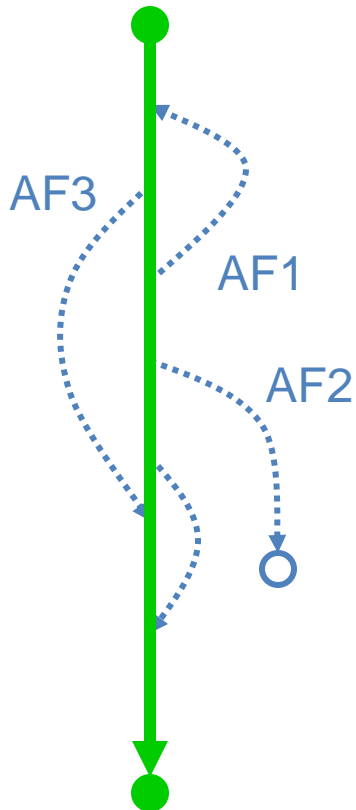


Ví dụ: sơ đồ Communication

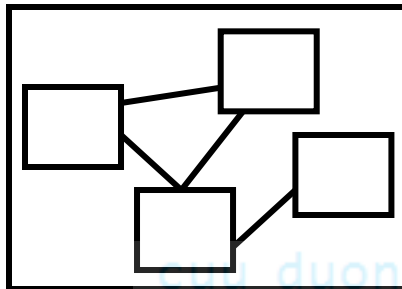


Một Interaction Diagram là chưa đủ

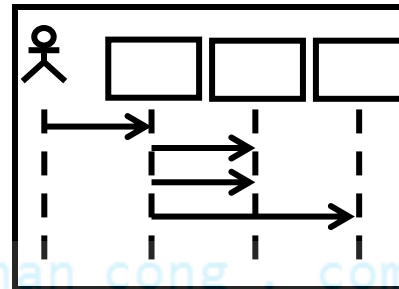
Basic Flow



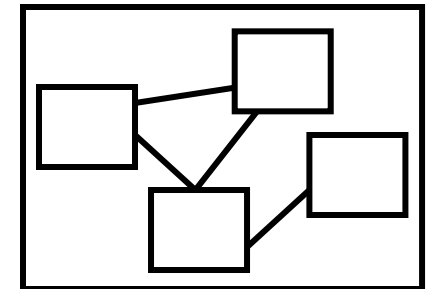
Alternate Flow 1



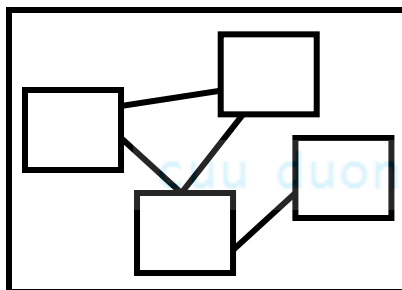
Alternate Flow 2



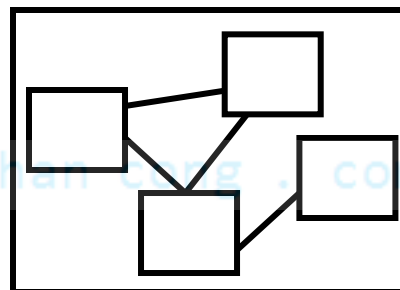
Alternate Flow 3



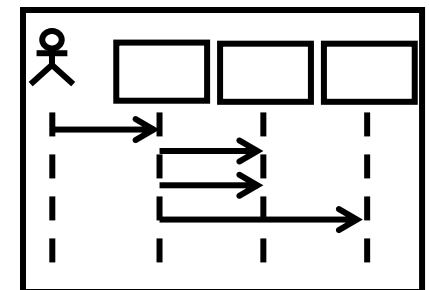
Alternate Flow 4



Alternate Flow 5



Alternate Flow n



Sơ đồ Communication và Sơ đồ Sequence

- Sơ đồ Communication
 - ▣ Chỉ ra các mối quan hệ bổ sung cho các tương tác
 - ▣ Trực quan hóa tốt hơn các mẫu cộng tác
 - ▣ Trực quan hóa tốt hơn các hiệu ứng tác động lên một đối tượng cụ
 - ▣ Dễ sử dụng hơn trong các vấn đề cần giải quyết tập thể
- Sơ đồ Sequence
 - ▣ Chỉ ra rõ ràng chuỗi các thông điệp
 - ▣ Trực quan hóa tốt hơn toàn bộ luồng sự kiện
 - ▣ Tốt hơn cho các đặc tả real-time và cho các scenario phức tạp