

## **Chapter 7**

# **Behavioral design**

MS. Le Thanh Trong

# Outline

1. Sequence diagram
2. Collaboration diagram
3. Activity diagram

# Outline

- 1. Sequence diagram**
2. Collaboration diagram
3. Activity diagram

# Sequence diagram

- ❖ Mô hình hóa **khía cạnh động** của hệ thống
- ❖ Nhấn mạnh sự **tương tác giữa các đối tượng** theo thời gian
- ❖ Biểu đồ chỉ ra tập **các đối tượng** và **các thông điệp** trao đổi giữa chúng, diễn tả các hoạt động như thế nào **theo trình tự thời gian**
- ❖ Phân biệt:
  - Sequence diagram tập trung vào luồng điều khiển
  - Collaboration diagram tập trung vào luồng dữ liệu

# Sequence diagram

- ❖ Biểu đồ trình tự có thể sử dụng cho nhiều cách:
  - Lập mô hình **tương tác ở mức cao** giữa các đối tượng hoạt động trong một hệ thống
  - Dùng để mô hình hóa **giao tác giữa các hệ thống con**
  - Lập mô hình tương tác giữa các đối tượng nhằm **thực hiện một use case**
  - ...

# Sequence diagram

- ❖ Biểu đồ trình tự gồm các phần tử mô hình như sau:
  - Đối tượng (object)
  - Đường sống, sinh tồn (life line)
  - Thông điệp (message)
  - Vùng hoạt động (activation)
  - Chú thích (note) và ràng buộc (constraint)
- ❖ Ngoài ra biểu đồ trình tự chứa các **tác nhân kích hoạt** biểu đồ. Tuy nhiên tác nhân không phải là phần tử của biểu đồ

# Sequence diagram

- ❖ Biểu đồ chứa một loạt các đối tượng sắp xếp theo hàng ngang. Còn thời gian biểu diễn theo trục thẳng đứng
- ❖ Chu kỳ sống của đối tượng được biểu diễn bằng các đường thẳng đứng gọi là đường sinh tồn
- ❖ Các thông điệp là các đường ngang nối liền các đường sinh tồn giữa các đối tượng
- ❖ Có nhiều loại thông điệp có ký hiệu và ngữ nghĩa khác nhau.

# Sequence diagram

## ❖ Đối tượng:

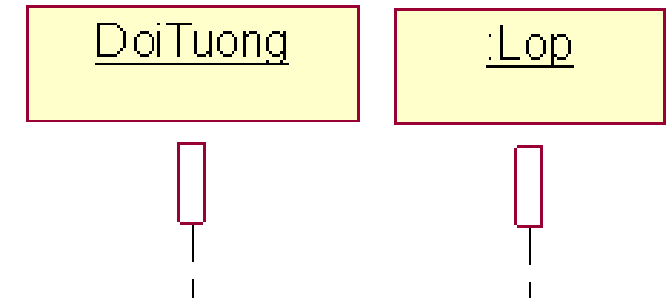
- Biểu diễn bằng hình chữ nhật, bên trong là tên đối tượng

## ❖ Đường sinh tồn:

- Biểu diễn bằng đường gạch thẳng đứt nét cho biết thời gian sống của đối tượng

## ❖ Vùng hoạt động:

- Biểu diễn bằng hình chữ nhật hẹp dọc theo đường sinh tồn
- Cho biết thời gian thực thi một hoạt động tương ứng.

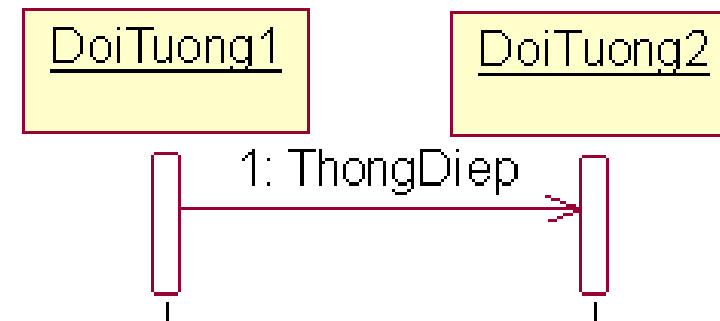




# Sequence diagram

## ❖ Thông điệp đơn giản

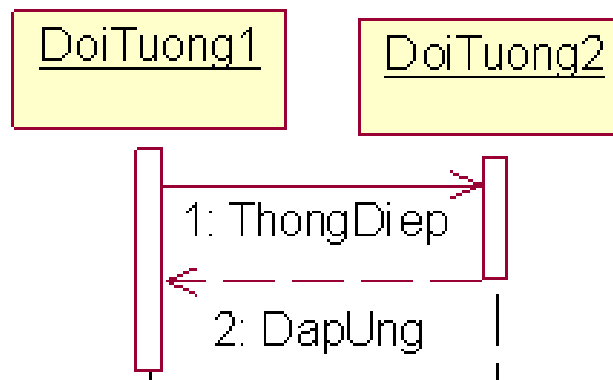
- Thông điệp được gửi đi giữa các đối tượng, đây là thao tác lớp được định nghĩa sau này.
- Thông điệp biểu diễn một đối tượng gọi thao tác của đối tượng khác



# Sequence diagram

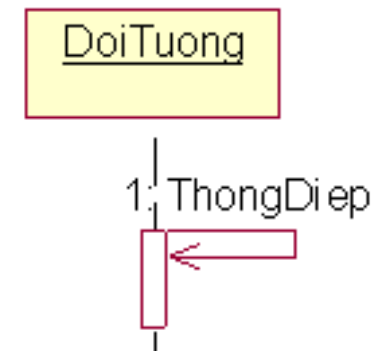
## ❖ Thông điệp:

- Đối tượng gửi thông điệp và chờ đến khi thông điệp được giải quyết xong



## ❖ Thông điệp phản thân:

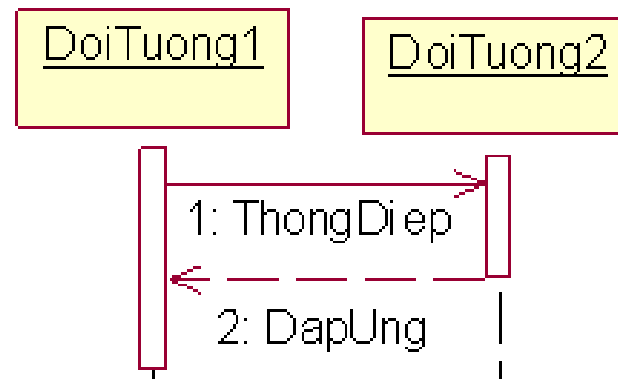
- Đối tượng gọi chính thao tác của mình



# Sequence diagram

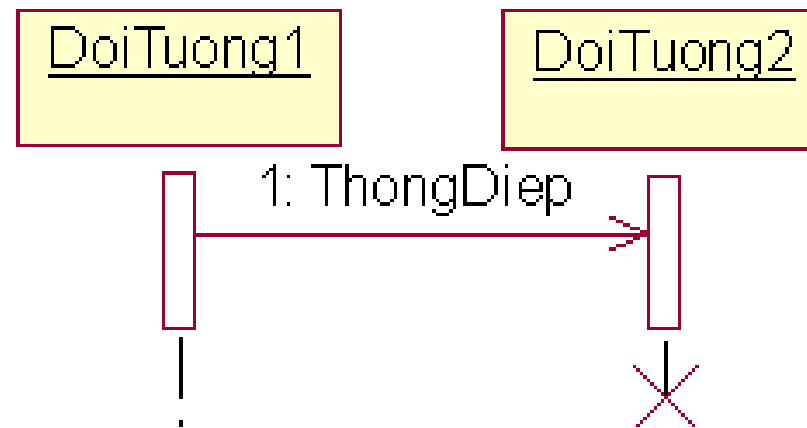
## ❖ Thông điệp đáp ứng:

- Sau khi nhận thông điệp từ đối tượng gọi nào đó. Một đối tượng có thể gửi thông điệp đáp ứng cho đối tượng gọi.



# Sequence diagram

- ❖ Thông điệp hủy đối tượng:
  - Hủy bỏ đối tượng



# Sequence diagram

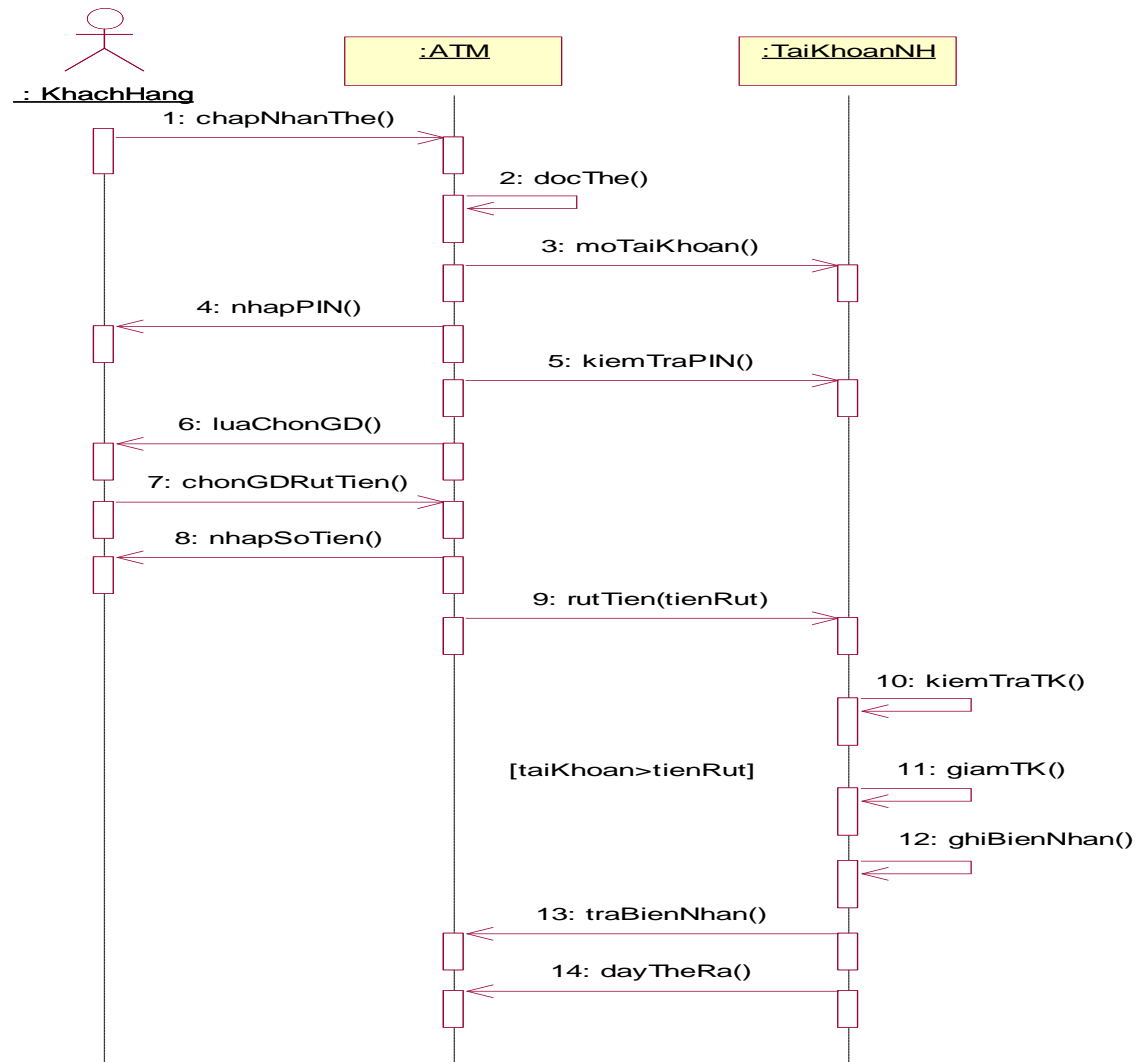
- ❖ Các bước thực hiện biểu đồ tuần tự
  - Xác định ngữ cảnh tương tác: hệ thống, hệ thống con, use case hoặc thao tác.
  - Xem xét các kịch bản thay thế
  - Vẽ các biểu đồ
    - Đặt các đối tượng từ trái sang phải
    - Bắt đầu bằng thông điệp khởi đầu tương tác, đặt các thông điệp theo chiều từ trên xuống dưới theo trục thời gian. Biểu diễn các đặc tính của các thông điệp cần thiết để giải thích ngữ nghĩa tương tác.
    - Thêm các vùng hoạt động nếu cần để trực quan hóa hành động
    - Thêm các ràng buộc khác nếu cần (thời gian)
    - Thêm các chú thích vào biểu đồ.

# Sequence diagram

- ❖ Khi xây dựng biểu đồ tương tác, ban đầu chỉ tập trung vào các thông tin liên quan đến khách hàng.
  - Chưa ánh xạ thông điệp thành thao tác
  - Chỉ đặc tả thông điệp không hình thức
  - Chưa ánh xạ đối tượng thành lớp
- ❖ Biểu đồ tạo ra trong giai đoạn này chỉ quan tâm đến luồng tác nghiệp
- ❖ Sau đó biểu đồ sẽ được bổ sung chi tiết hơn như ánh xạ đối tượng thành lớp, thông điệp thành thao tác, bổ sung một số đối tượng khác như giao diện, điều khiển

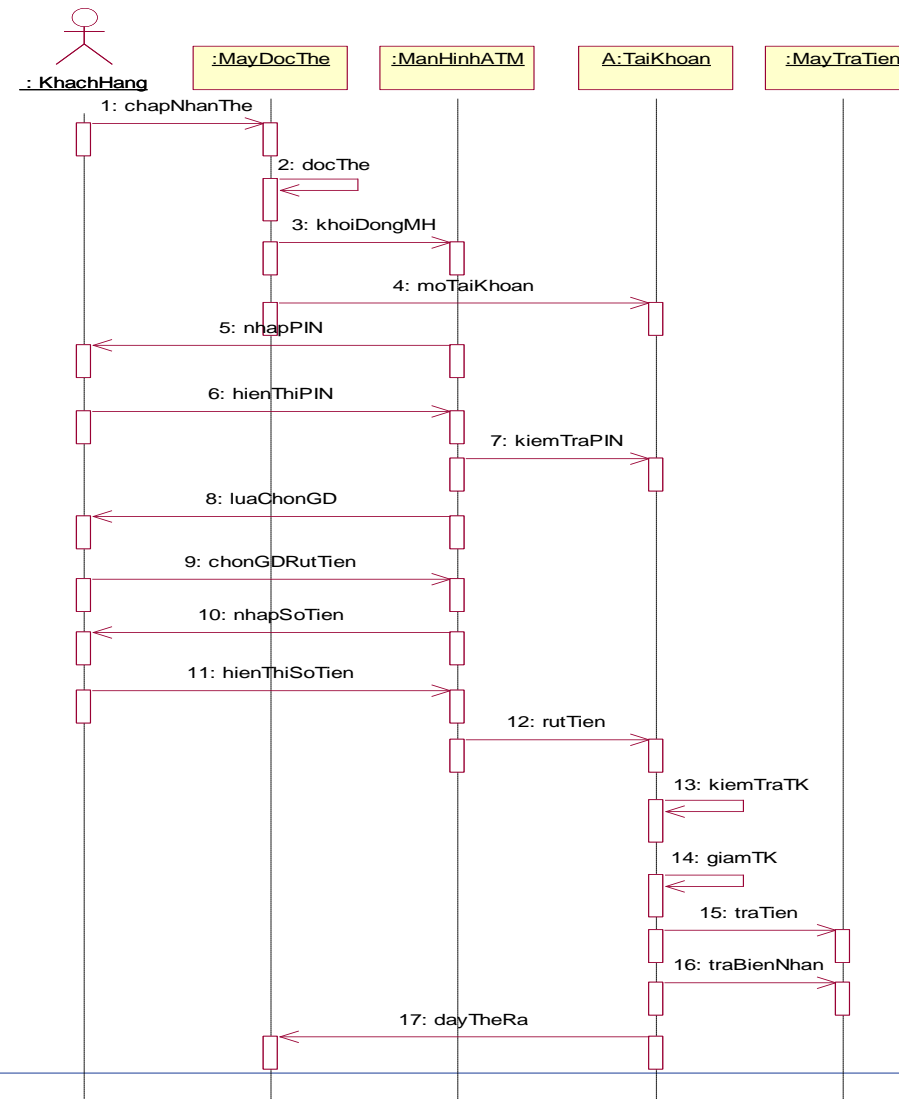
# Sequence diagram

❖ Biểu đồ trình tự của hệ thống ATM ban đầu



# Sequence diagram

❖ Biểu đồ trình tự ở mức chi tiết hơn



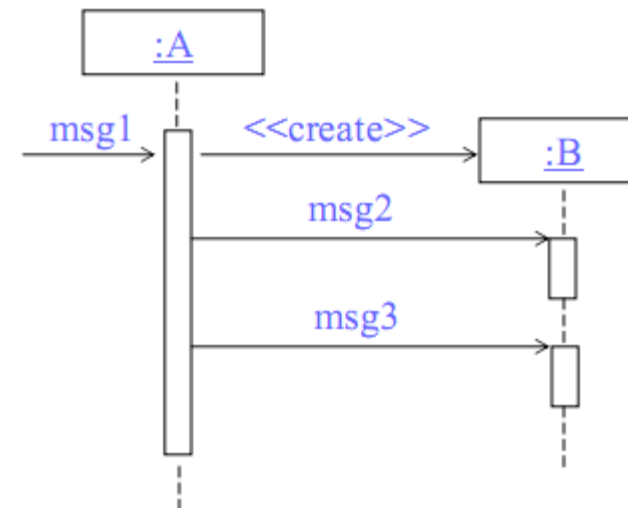
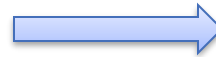


# Sequence diagram

❖ Thể hiện một thao tác của một lớp

```
public class A
{
    private B objB;
    public void msg1()
    {
        objB = new B();
        objB.msg2();
        objB.msg3();
    }
}

public class B
{
    ...
    public void msg2() { ... }
    public void msg3() { ... }
}
```



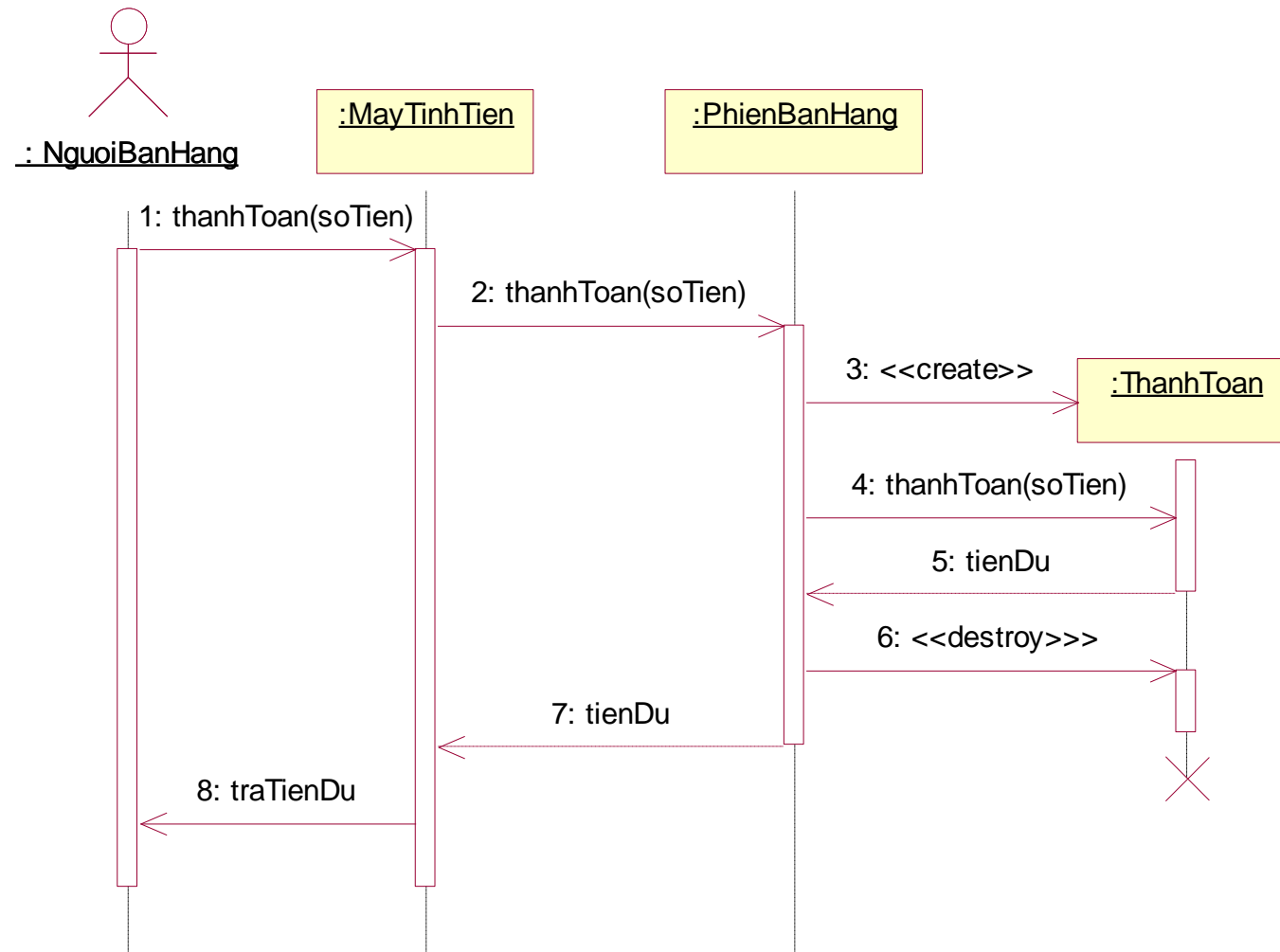
# Sequence diagram

## ❖ Minh họa chức năng tính tiền:

- Các đối tượng:
  - Người bán hàng
  - Máy tính tiền
  - Phiên bán hàng
  - Thanh toán
- Các thao tác (kịch bản)
  - Người bán hàng gửi thông điệp thanh toán với số tiền tương ứng
  - Máy tính tiền yêu cầu phiên bán hàng thanh toán với các thông tin trên
  - Phiên bán hàng thực thi các công việc sau:
    - Tạo đối tượng thanh toán
    - Yêu cầu thanh toán với thông tin vừa nhận
    - Đối tượng thanh toán trả về số tiền dư
    - Thông điệp hủy đối tượng thanh toán
  - Phiên bán hàng trả thông tin số dư về cho máy tính tiền
  - Máy tính tiền thực hiện thông điệp trả tiền dư về cho người bán hàng.

# Sequence diagram

## ❖ Chức năng tính tiền



# Bài tập: Vẽ SD cho việc phúc khảo

## ❖ Phúc khảo

1. Sinh viên gửi đơn phúc khảo cho PĐT
2. PĐT kiểm tra thông tin trong đơn
3. PĐT gửi đơn cho khoa quản lý lớp học
4. Khoa lập danh sách và thông báo, gửi đơn đến gv
5. GV chấm lại bài, điền kết quả phúc khảo vào đơn và gửi lại khoa
6. Khoa xác nhận kết quả và gửi lại đơn cho PĐT
7. PĐT kiểm tra, lập bản tổng hợp kết quả, cập nhật điểm trên HT và thông báo kết quả đến sinh viên

## ❖ Đăng ký học phần

## ❖ Cấp bảng điểm sinh viên

# Outline

1. Sequence diagram
- 2. Collaboration diagram**
3. Activity diagram

# Collaboration diagram

- ❖ Trong hệ thống HĐT, các đối tượng phối hợp hoạt động với nhau tạo ra chức năng mà người sử dụng yêu cầu
- ❖ Để phối hợp hoạt động, các đối tượng cần giao tiếp với nhau bằng cách **gởi thông điệp**
- ❖ Biểu đồ cộng tác tập trung vào
  - Quan hệ giữa các đối tượng
  - Cấu trúc tổ chức của các đối tượng
  - Luồng dữ liệu trong kịch bản

# Collaboration diagram

- ❖ Hiển thị **luồng dữ liệu** khác với biểu đồ tuần tự hiển thị **luồng điều khiển**.
- ❖ Là sự mở rộng của biểu đồ đối tượng
- ❖ Chỉ ra:
  - Thứ tự gọi các thông điệp: mỗi thông điệp được gán một số tuần tự
  - Điều kiện gọi các thông điệp

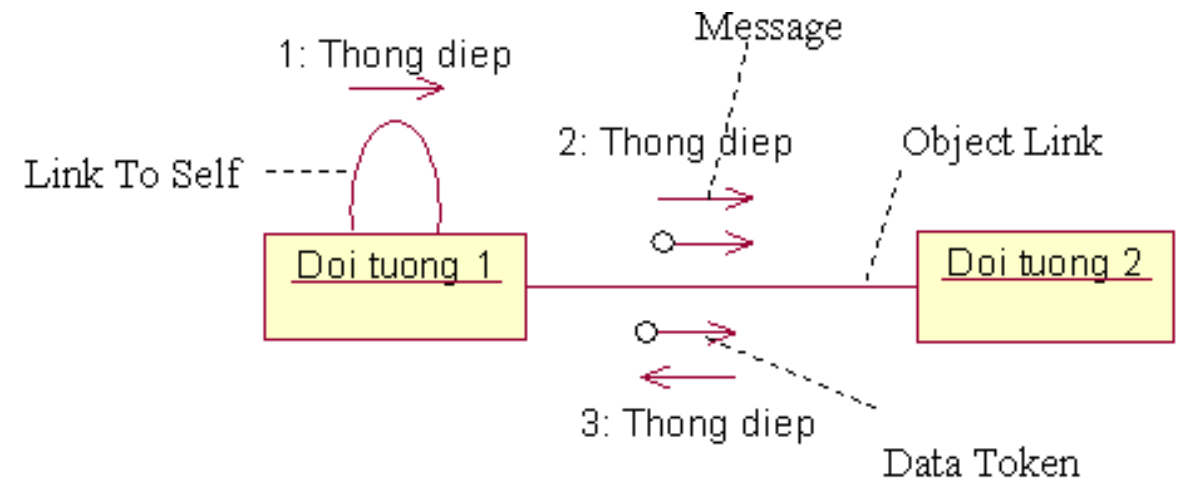
# Thành phần

## ❖ Bao gồm:

- Các đối tượng
- Các liên kết (đường thẳng nối các đối tượng để chỉ ra dòng chảy thông điệp giữa các đối tượng)

## ❖ Các phần tử mô hình biểu diễn trong biểu đồ cộng tác:

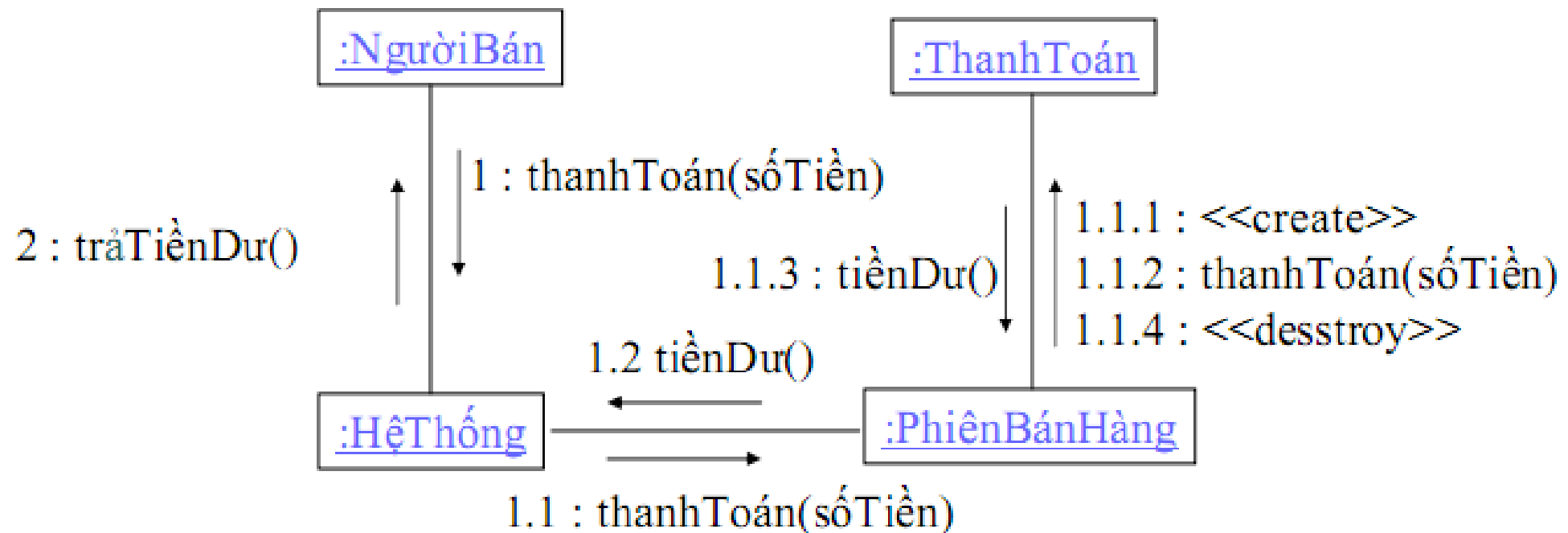
- Đối tượng (object)
- Liên kết (link)
- Thông điệp (message)
- Luồng dữ liệu (data token)





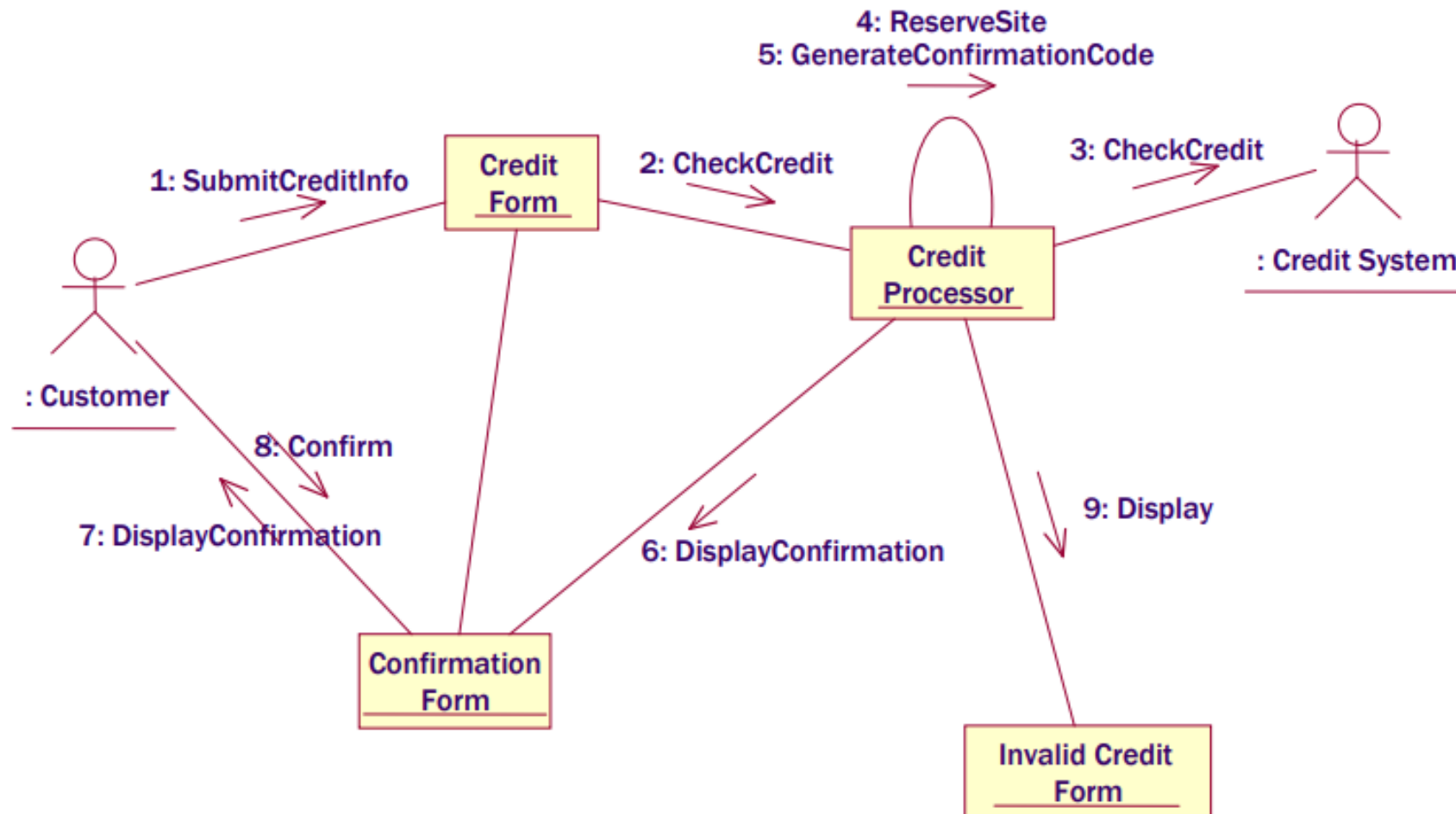
# Ví dụ

## ❖ Ví dụ hoạt động “thanh toán”



# Ví dụ

## ❖ Ví dụ hoạt động “đặt chỗ chuyến bay”



# Bài tập

- ❖ Xây dựng sơ đồ cộng tác
  - Đăng ký học phần
  - Phúc khảo
  - Cấp bằng điểm sinh viên

# Outline

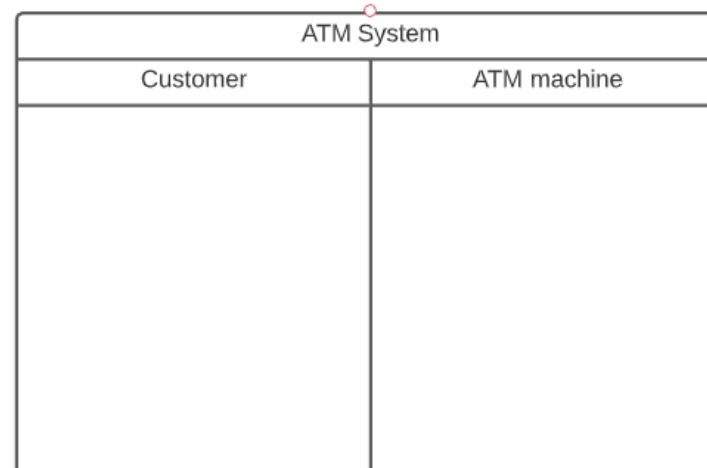
1. Sequence diagram
2. Collaboration diagram
- 3. Activity diagram**

# Activities Diagram

- ❖ Mô tả các hoạt động, luồng xử lý bên trong hệ thống
- ❖ Nhấn mạnh vào trình tự và điều kiện của luồng
- ❖ Các hoạt động được diễn ra bởi:
  - Kết thúc quá trình thực thi
  - Đối tượng và dữ liệu được phát sinh
  - Một số sự kiện bên ngoài luồng được kích hoạt
- ❖ Được sử dụng để mô tả các quy trình nghiệp vụ trong hệ thống hoặc luồng xử lý của một use-case

# Thành phần

- ❖ **Swimlane:** Xác định các đối tượng tham gia hoạt động trong một quy trình



- ❖ **Start - End:** Bắt đầu và kết thúc quy trình

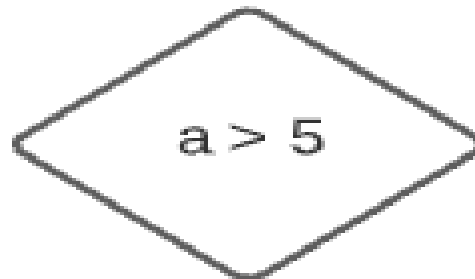


# Thành phần

- ❖ **Activity:** mô tả một hoạt động trong hệ thống, hoạt động được thực hiện bởi các đối tượng

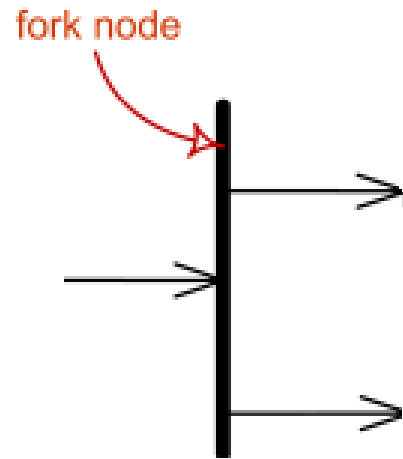


- ❖ **Branch**

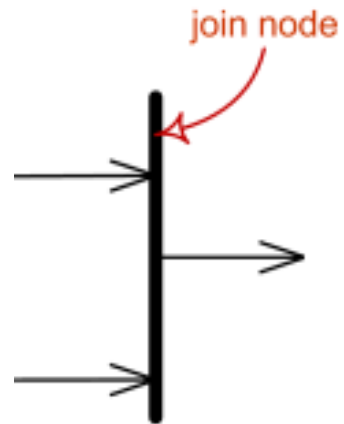


# Thành phần

## ❖ Fork



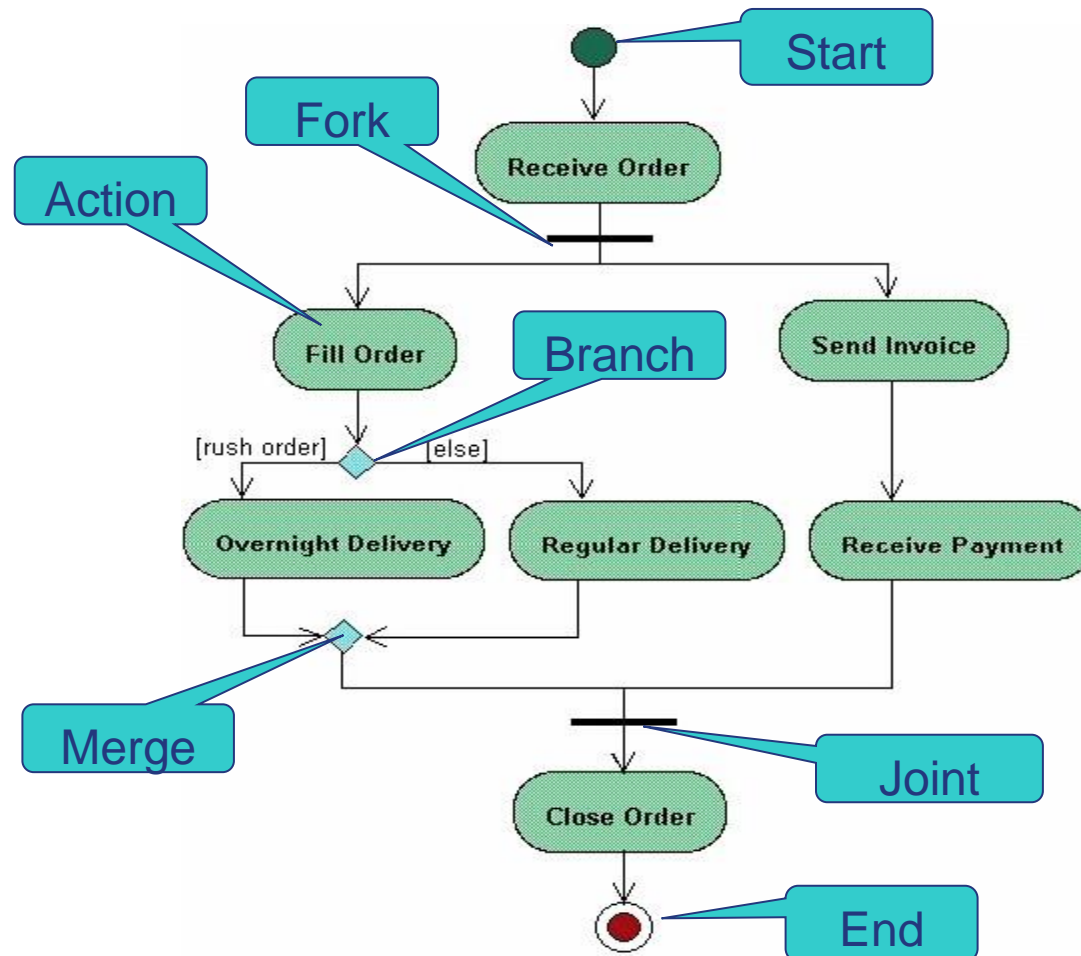
## ❖ Join





# Ví dụ

## ❖ Ví dụ hoạt động “xử lý đơn hàng”



# Bài tập

- ❖ Xây dựng sơ đồ hoạt động cho các quy trình
  - Đăng ký học phần
  - Phúc khảo
  - Cấp bằng điểm sinh viên

