Object-Oriented Language and Theory

Lab 11: Usecase diagram, Activity diagram and Sequence diagram

**Lab’s Objectives:**

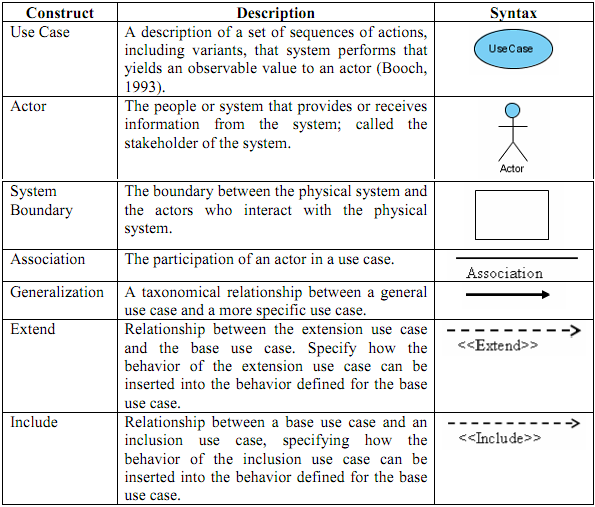
In this lab, you will practice with:

* Familiarize with the concept underlining Use Case Diagram for requirement analysis.

**1. Definition of Use Case Diagram**

* A Use Case is a set of scenarios describing an interaction between a user and a system.
* Use Case diagram displays (in symbolic form) the relationship among actor and use cases.
* Use case represents the system’s functionality, the requirements of the system from the user’s perspective.

**UML Notation for Use Case Modeling**



**2. Mail Order System**

In order to improve the operational efficiency of a mail order company, the chief executive officer is interested in computerizing the company’s business process. The major business activities of the company can be briefly described as follows:

A customer registers as a member by filling in the membership form and mailing it to the company. A member who has not been active (no transactions made) for a period of one year will be removed from the membership list and he/she needs to re-apply for the reinstatement of the lapsed membership.

A member should inform the company of any changes of personal details such as home address, telephone numbers, etc. A member can place an order by filling out a sales order form and faxing it to the company or by phoning the Customer Service Assistant with the order details.

The Customer Service Assistant first checks for the validity of membership and enters the sales order information into the system.

The Order Processing Clerk checks the availability of the ordered items and holds them for the order. When all the ordered items are available, he/she will schedule their delivery.

The Inventory Control Clerk controls and maintains an appropriate level of stock and is also responsible for acquiring new items.

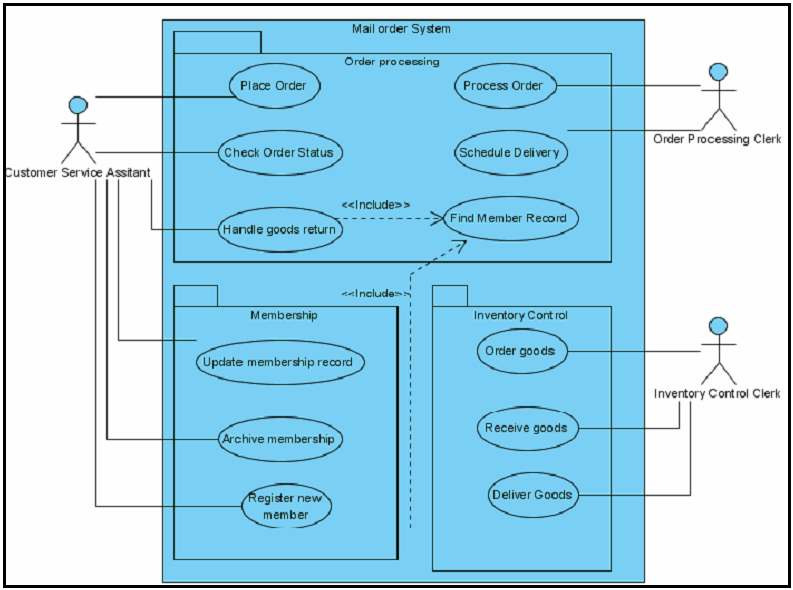
If there is a problem with an order, members will phone the Customer Service Assistant. The Customer Service Assistant will take appropriate action to follow up the sales order.

Members may return defective goods within 30 days and get their money back.

The system will record the name of the staff member who has initialized an updated transaction to

the system.

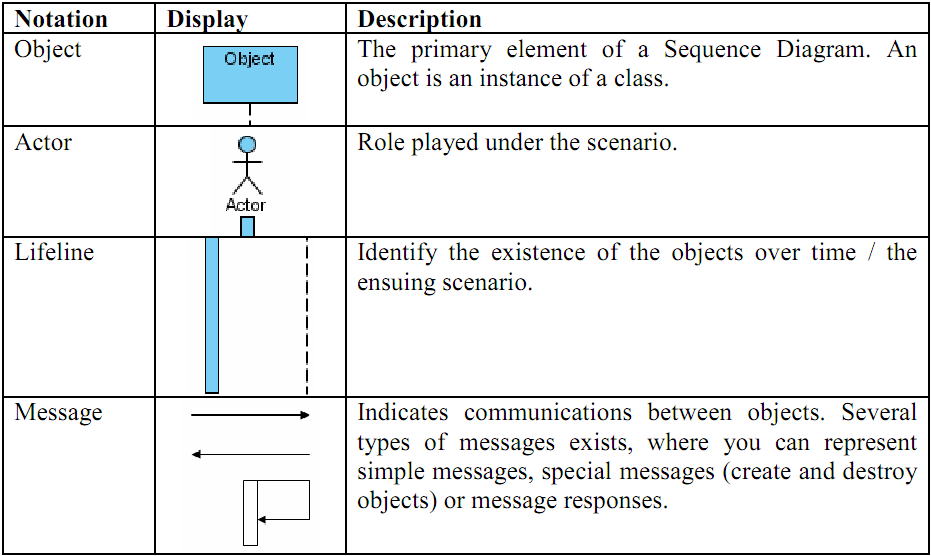
**Create usecase diagram**



**3. Definition of Sequence Diagram**

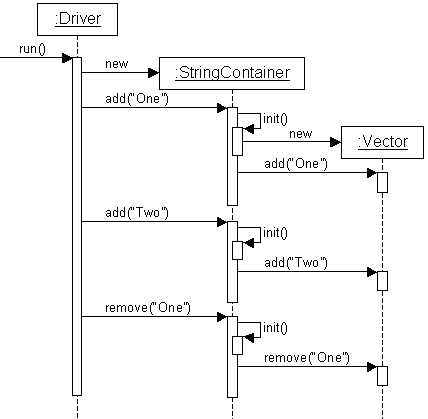
* Shows how processes operate with one another in what order.
* Illustrates how objects interact with each other.
* Emphasizes time ordering of the messages.
* Can model simple sequential flow, branching, iteration, recursion and concurrency.

**Key Components of System Sequence Diagram**

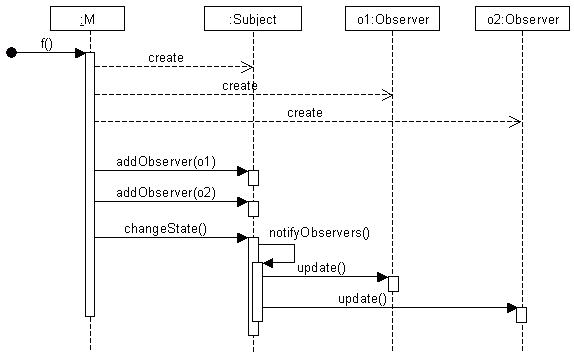
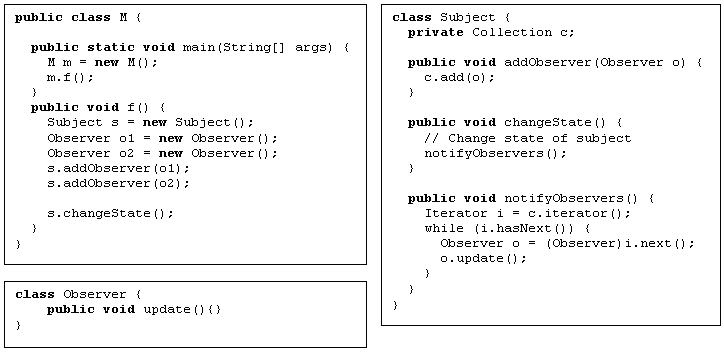


**4. For the Java source code below, build the sequence diagram that represents the behavior for the call d.run() in main() method.**

**import** java.util.Vector;  
**public class** Driver {  
    **private** StringContainer b = null;  
    **public static void** main(String[] args){  
        Driver d = **new** Driver();  
        d.run();  
    }  
    **public void** run() {  
        b = **new** StringContainer();  
        b.add("One");  
        b.add("Two");  
        b.remove("One");  
    }  
}  
**class** StringContainer {  
    **private** Vector v = null;  
    **public void** add(String s) {  
        init();  
        v.add(s);  
    }  
    **public boolean** remove(String s) {  
        init();  
        **return** v.remove(s);  
    }  
    **private void** init() {  
        **if** (v == null)  
            v = **new** Vector();  
    }  
}



**5. For the source code written in Java below, assuming that class M already exists, build the sequence diagram starting with the method call m.f().**



**Bài luyện tập: Chương trình quản lý sổ địa chỉ đơn giản**

Xây dựng một chương trình đơn giản được sử dụng để quản lý sổ địa chỉ. Một sổ địa chỉ chứa một tập hợp các mục, mỗi mục ghi: họ, tên, địa chỉ, thành phố, tỉnh thành, mã bưu điện và số điện thoại của một người.

Người sử dụng có thể thêm mới một mục thông tin vào sổ địa chỉ, chỉnh sửa thông tin hiện có về một người (ngoại trừ tên người đó) và xóa bỏ mục thông tin không sử dụng.

Chương trình cho phép sắp xếp các mục trong sổ địa chỉ theo thứ tự bảng chữ *cái theo tên hoặc* theo mã bưu điện. Người dùng có thể in ra tất cả các mục trong sổ địa chỉ.

Mỗi sổ địa chỉ được lưu trữ vào một tệp văn bản, người dùng có thể tạo một sổ địa chỉ mới, lưu thông tin sổ địa chỉ vào tệp hoặc mở nội dung sổ địa chỉ từ tệp có sẵn (các mục chức năng như: New, Open, Close, Save và Save as...). Tùy chọn Thoát cho phép đóng sổ địa chỉ đang mở và chấm dứt chương trình.

Chương trình sẽ theo dõi xem có bất kỳ thay đổi nào được thực hiện đối với sổ địa chỉ hiện tại kể từ lần lưu cuối cùng hay không và sẽ cung cấp cho người dùng cơ hội lưu các thay đổi khi đóng sổ địa chỉ hoặc khi mở sổ địa chỉ khác hay khi thoát chương trình.

- *Bước 1*: Xác định các yêu cầu chức năng của chương trình

***+ Xác định các tác nhân:***

+ Trả lời các câu hỏi sau để xác định Actor cho hệ thống:

 Ai sử dụng hệ thống này?

 Hệ thống nào tương tác với hệ thống này?

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên tác nhân** | **Mô tả ngắn** |
| 1 | Người dùng | Người sử dụng chương trình |
| 2 |  |  |

***+ Xác định các ca sử dụng”***

+ Trả lời câu hỏi các Actor sử dụng chức năng gì trong hệ thống? chúng ta sẽ xác định được các Use Case cần thiết cho hệ thống.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Ca sử dụng** | **Mô tả ngắn** | **Tác nhân** | **Độ phức tạp** |
| 1 | thêm mới một mục thông tin | thêm mục thông tin mới vào sổ địa chỉ về một người (ngoại trừ tên người đó) | Ngưởi sử dụng |  |
| 2 | chỉnh sửa thông tin | Chỉnh sửa thông tin hiện có về một người (ngoại trừ tên người đó) | Ngưởi sử dụng |  |
| 3 | xóa bỏ mục thông tin | Xoá bỏ mục thông tin không sử dụng | Ngưởi sử dụng |  |
| 4 | sắp xếp các mục trong sổ địa chỉ | Sắp xếp thông tin theo bảng chữ cái hoặc theo mã ZIP | Ngưởi sử dụng |  |
| 5 | In thông tin | In thông tin của sổ địa chỉ ra màn hình | Ngưởi sử dụng |  |
| 6 | lưu trữ sổ địa chỉ | Lưu trữ thông tin sổ địa chỉ | Ngưởi sử dụng |  |
| 7 | mở nội dung sổ địa chỉ | Mở thông có trong sổ địa chỉ | Ngưởi sử dụng |  |
| 8 | Thoát | Thoát chương trình | Ngưởi sử dụng |  |
| 9 |  |  |  |  |
| 10 |  |  |  |  |
| 11 |  |  |  |  |
| 12 |  |  |  |  |

***+ Xác định các quan hệ***

Phân tích và các định các quan loại hệ giữa các Actor và Use Case, giữa các Actor với nhau, giữa các Use Case với nhau sau đó nối chúng lại chúng ta sẽ được bản vẽ Use Case.

- *Bước 2*: Xây dựng biểu đồ Usecase tổng quan cho chương trình quản lý sổ địa chỉ

- *Bước 3*: Đặc tả kịch bản chi tiết cho từng Usecase

- Mô tả kịch bản bằng lời

- Xây dựng sơ đồ hoạt động mô tả nội dung kịch bản cho từng Usecase

|  |  |  |  |
| --- | --- | --- | --- |
| **Mã UC (UC #)** | xxxxx | **Tên usecase** |  |
| **Tác nhân** |  | | |
| **Điều kiện trước** |  | | |
| **Luồng thực thi chính** |  | | |
| **No.** | **Thực hiện** | **Hành động** |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| **Luồng thực thi mở rộng** |  | | |
| **No.** | **Thực hiện** | **Hành động** |  |
| 6a. |  |  |  |
| 7a. |  |  |  |
| 8a. |  |  |  |
| **Điều kiện sau** |  | | |