



# Tutorial - UML Introduce 1

Designed by ZHU Yueming

## Reference

1. Bernd Bruegge and Allen H.Dutoit, Object Oriented Software Engineering Using UML, Patterns, and Java Third Edition
2. Zhang Yuqun, Slides of Object Oriented Analyze and Design
3. Wang Wenmin, Slides of Object Oriented Analyze and Design

## Experimental Objective

1. Learn how to design use case diagram according to a paragraph of requirement description
2. Understand different relationship between use cases.

## General Description

Software engineers often have to communicate properties of the systems they develop with other stakeholders who may or may not be familiar with the technical details of the system that is being examined. UML provides a unified, consistent way to communicate information about software systems in a way designed to be intuitive for both technical and nontechnical individuals. Many tools exist that support working with UML that serve to support the development of a UML model and expressing that model by generating graphical artifacts as specified by the UML standard.

In this laboratory, you can use any plot tools to accomplish exercise. We recommend [ProcessON](#) or [Xunjie](#) as cartographic software. In this semester, we'll take two weeks to develop three most common UML diagrams:

- User case diagram
- Class model diagram
- Sequence diagram

## Topic 1. Use case diagram

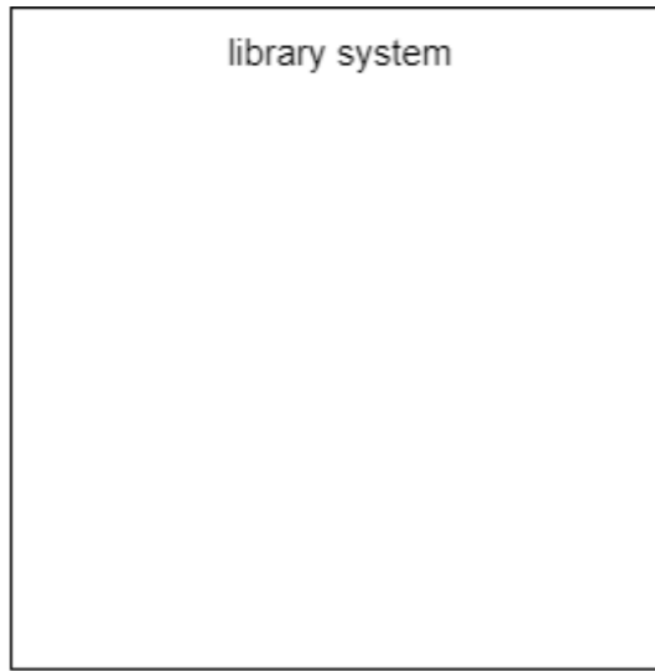
**Actor:** An actor in a use case is an **external** agent that uses or interacts with the system. An actor can be a user or a role, such as a person or an external system characteristic of the environmental, such as time or temperature change.



**Use case:** A use case is a collection of related scenarios, including normal and alternative scenarios. The scenarios can be regarded as a behaviorally related sequence of steps, automated or manual, for the purpose of completing a business task.



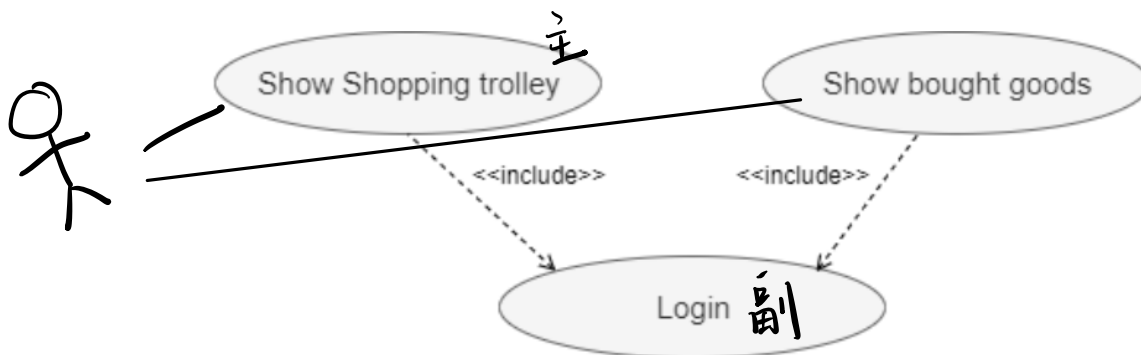
**System boundary:** System boundary refers to the boundary between different systems, and we can use it to distinguish elements within the system from the elements outside the system. Use case always serve as the function in the system boundary while actors refer to the external element that interact with system.



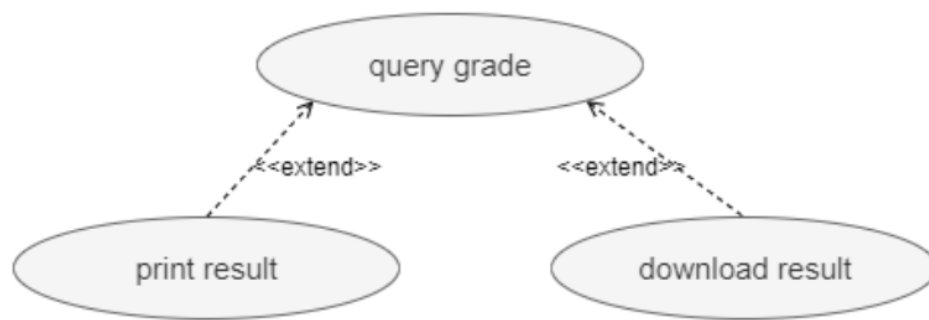
### Relationships of Use case diagram:

When describing a complex system, its use case model can become quite complex and can contain redundancy. We use these three types of relationships including inclusion, extension, and inheritance to reduce the complexity of the model.

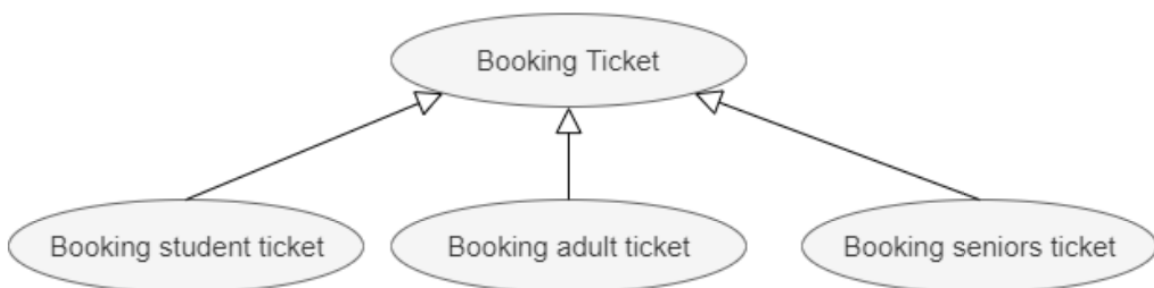
- **Include (包含):** Include is a directed relationship between two use cases, implying that the behavior of the included use case is inserted into the behavior of the including use case.



- **Extend (扩展):** This relationship specifies that the behavior of a use case may be extended by the behavior of another use case.



- **Generalization (泛化):** A generalization is a taxonomic relationship between a more general classifier and a more specific classifier. Each instance of the specific classifier is also an indirect instance of the general classifier. Thus, the specific classifier inherits the features of the more general classifier.



## 1. Basic Exercises

- **Exercise 1:**

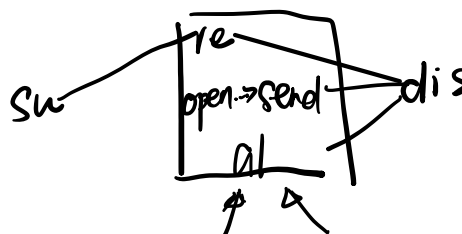
Translate the following English statements into a representative use case diagram: Library System: The system shall have the following four types of actors: `BookBorrower`, `JournalBorrower`, `Browser`, and `Librarian`. A user playing the role of a `BookBorrower` shall be able to borrow a book, return a book, and extend the loan of a book. A `JournalBorrower` shall be able to borrow a journal and return a journal. A `Browser` shall be able to browse, and a librarian shall be able to update the catalog.

- **Exercise 2:**

Translate the following English statements into a representative use case diagram. In this diagram you should add the use case relationship of inclusion and extension.

**Fried System:** When a fire occurs, the `supervisor` needs to `report the disaster`, and then the system will automatically send the disaster to `dispatcher`. After the `dispatcher` receives the report, he can `send the fire resource allocation request` to the system according to the fire situation and accident location. In this step, the `dispatcher` may `choice to open the map` finding the specific location of disaster. At this point, the system will automatically calculate whether to meet the needs of `dispatchers`. When the fire resources in the inventory information database meet the needs of the dispatcher, the `dispatcher` can allocate the corresponding fire resources including allocating fire `extinguishers` and `allocating fire trucks`.

There are two actors in this use case diagram: `supervisor` and `dispatcher`. You need to find the use cases and the relationship of these use cases. Please begin to finish this use case diagram.



## 2. Comprehensive Exercise

### 2.1. WeChat Payment

Mobile payment is very popular in China. Many people use wechat as their third-party payment platform. The following description is a simple design of its payment function.

You can do multiple operations in wechat pay, including checking how much balance you have in your wallet, adding bank cards to recharge or withdraw, looking up transaction history, sending lucky money to your contacts, and making payment to strangers by a QR code. In last case, we have two ways of transaction, either the payer scanning the QR code of payee's or the payee scanning the QR code of payer's. In addition, user also can change his nickname, password or other information.

**Question: Draw an use case diagram according to the above scenario**

### 2.2. Parting

In ShenZhen city, a automatic parking payment system is widely used by almost all shopping malls. Then, it is a time for you to help a newly opened shopping mall design the system. The following description is a simple requirement of the parking management:

Before a car enter the parking lot, there is an automatic scanning device that can register the time and the license plate number of the car. Before leaving, the customer has to pay the parking fee, otherwise he/she cannot drive out of the parking lot. When paying, the customer can pay the parking fees directly, or he/she can choice to deduct the payment with his/her VIP points if he/she is the VIP of this shopping mall. Customers need to drive out of the parking lot within 30 minutes after the successful payment time, otherwise, the cost will be recalculated. There are several administrators who can view the parking list, and they can also query and modify the parking info of each car. On the other hand, in order to promote sales, the shopping mall set a VIP Day on the 1st day of each month. On the 12:00 am in each VIP Day, the system could send points in each VIP account automatically.

**Question: Draw an use case diagram according to the above scenario**

