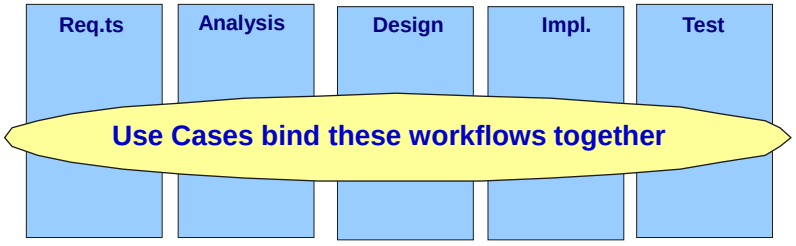
Use Case Modeling

Il-Yeol Song, Ph.D.
Professor
College of Information Science and Technology
Drexel University
Philadelphia, PA 19104
Song@drexel.edu
http://www.ischool.drexel.edu/faculty/song/

Objectives

- Study use case modeling
  - What is a use case modeling?
  - What is a *good* use case?
  - How to identify good use cases?
  - Develop a use case diagram
  - What are the roles of use case modeling?
- Use case relationships
- Use case documentation

Use Case Driven Approach



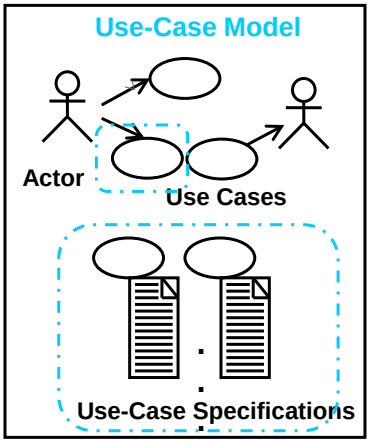
Use cases involve a shift in thinking

From a focus on the *functions* of a system

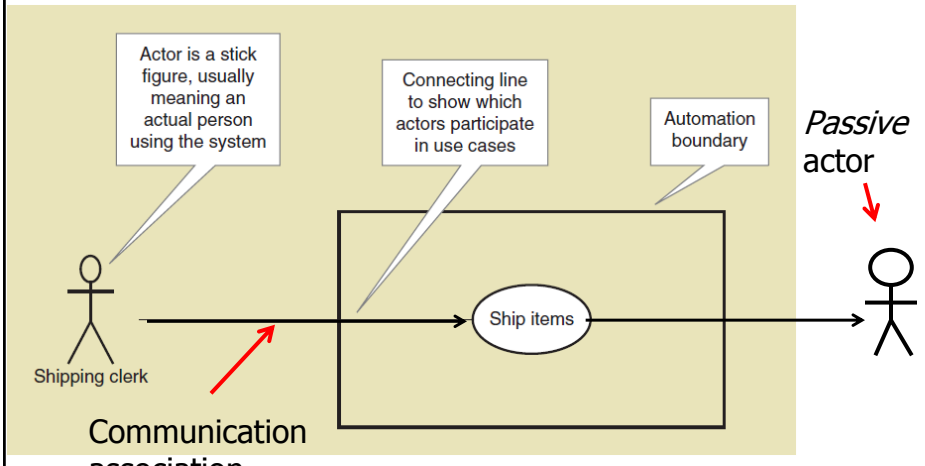
To a focus on the ***value*** a system must deliver ***for its stakeholders***

Use case model

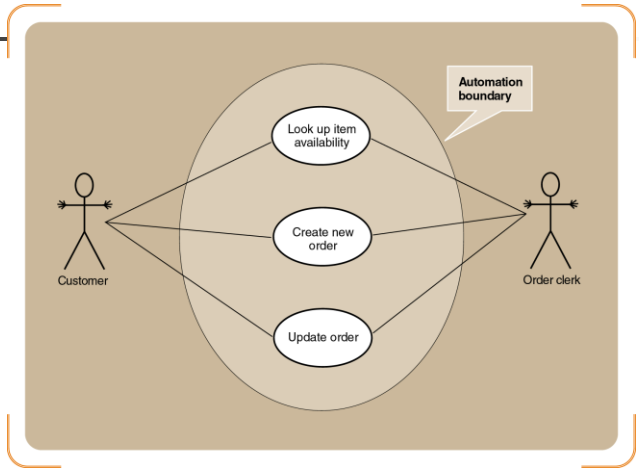
- A technique for modeling ***system functionality*** in terms of users, goals, and tasks
  - Developed by Iva Jacobson
- The Use case model
  - Use case diagram
  - Use case specification
- Use case diagram
  - System being modeled
  - Actors
  - Use cases
  - Communication association



Use Case Diagrams Symbols



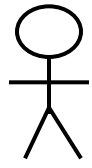
The arrow is optional, but it makes diagram clear



A Use Case Diagram of the Order-Entry Subsystem for RMO, Showing a System Boundary

Use case modeling

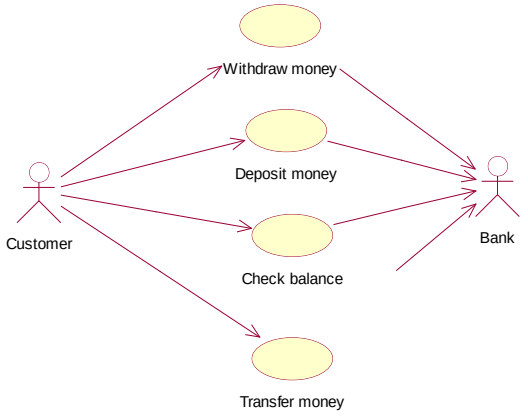
- Use cases define functional requirements
- The set of use cases defines the complete functionality of a system
- A use case is *a sequence of actions* a system performs to achieve *a goal of a particular actor*.
- An actor represents anything that directly initiates an use case (human, SW, HW, Clock) of a computer system.
- A use case diagram represents *who* can do *what*



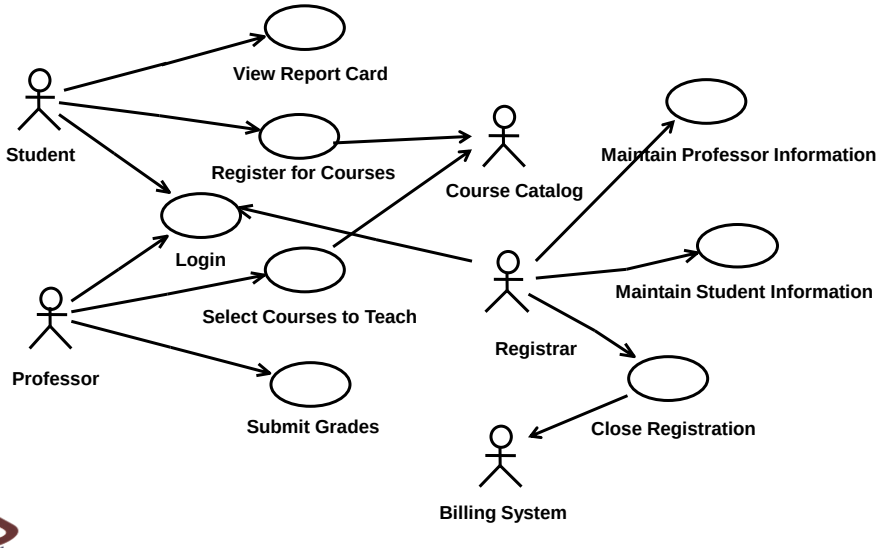
Actor

Use Case

Use case diagram for ATM

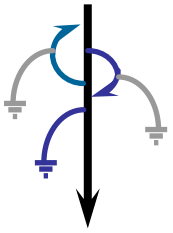


Example: use case diagram



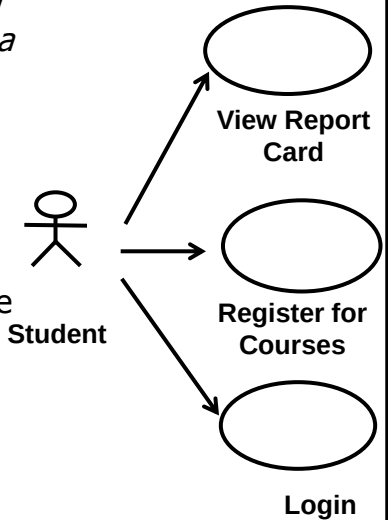
Scenarios

- A scenario is an instance of a use case
- A use case is a set of success and failure scenarios tied together by *a common user goal*.
- Example of scenarios in use case *withdraw money*
  - Success scenarios
    - Get money and a receipt
  - Failure scenarios
    - Entered a wrong PIN
    - Balance is too low



Properties of a good use case

- A use case is a set of *success and failure scenarios* tied together by *a common user goal*.
- Represents a *complete* item of work from *start* to *end*
- Produces *meaningful* results to the actor (achieve a goal)
- An *automated* high-level functionality



Heuristics for use cases

(1) The EBP (Elementary Business Process) test

- The unit of work performed ***by one person in one place at one time*** and still add value to the business
- Use this rule to remove too high level use cases
  - e.g, *Process loan applications* does not satisfy the EBP rule as it includes multiple actors and multiple steps that can be taken at different times. *Submit loan application* satisfies the EBP rule.

Heuristics for use cases

(2) The Automation test: Is the activity an automated one?

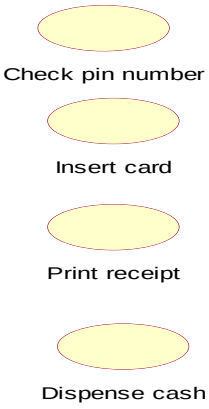
- **A use case is an *automated* (not a manual)** high-level functionality

(3) The Completeness test: Does the task contain all the components **from the start to finish** in achieving the goal?

- A use case must contain all the tasks from the start to end in achieving a goal of an actor

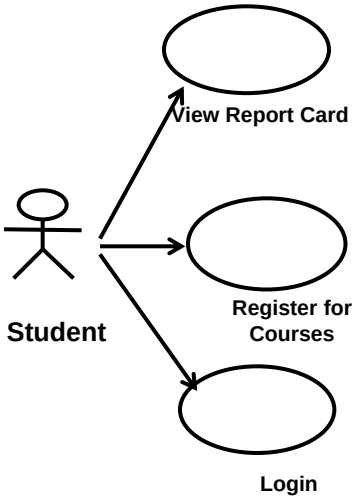
- Use these guidelines to filter out excessive low level use cases

Examples of bad use cases in ATM domain



Properties of a good actor

- An actor is anything that ***directly initiates*** an use case of a computer system
- An actor can be a human being, SW, HW or a clock
- An actor represents *a role*
- Ex: who is an actor?
  - A customer in Video Rental System at Blockbuster?
  - A passenger in Airline ticketing system in airport



Use Case Name

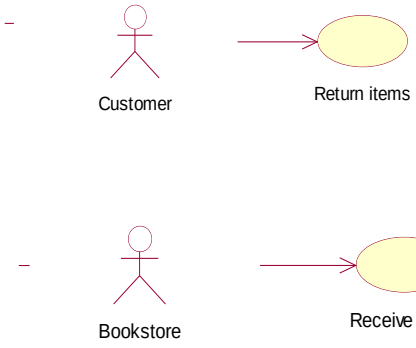
- The use case name is the primary actor's goal.
- Use **Verb + Noun** as a use case name
  - Process rent, Process return
  - Manage inventory
- Examples of bad use case names:
  - Sales
  - Customer service

Exercises

1. *EBP*
2. *Automation*
3. *Completeness*

- Which of the following are good use cases?
  - Negotiate a supplier contract (in Wal-mart)
  - Handle return of sold items (in Wal-mart)
  - Contact customer (in Wal-mart)
  - Ship orders (in Wal-mart)
  - Log in
  - Move a piece on Game Board (in a computer chess game)
  - Make a sandwich (at Burger King)
  - Consider what courses to take next term
  - Professors grade term papers
  - Download an assignment from Blackboard
  - Register a course in Banner

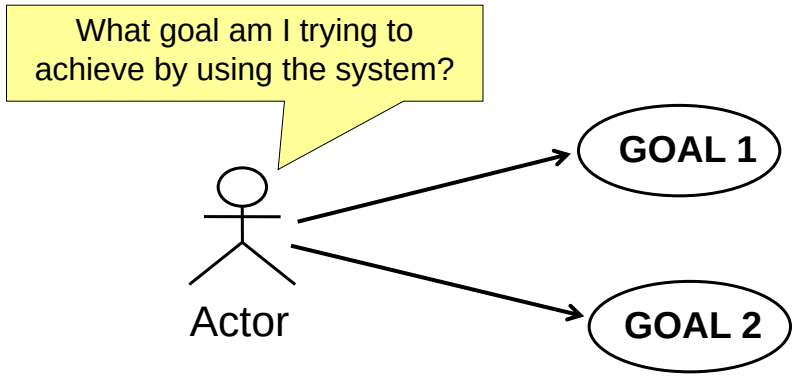
What's wrong with the following use cases in the online book store domain?



How to develop a use case diagram?

- The Actor-based approach (**AGU** Approach)
  - Choose **the system boundary** clearly
  - Identify the primary **actors**
  - For each actor, identify their user **goals**
  - For each goal, identify events that becomes one or more **use cases**

Find use cases

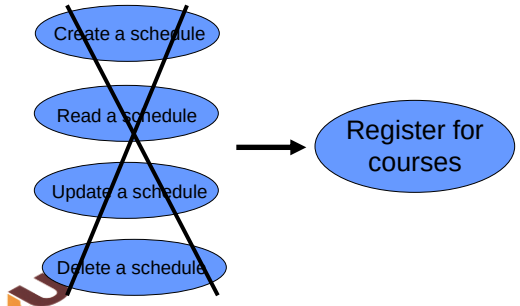


USER/ACTOR	USER GOAL
Order clerk	Look up item availability Create new order Update order
Shipping clerk	Record order fulfillment Record back order
Merchandising manager	Create special promotion Produce catalog activity report

Identifying Use Cases by Focusing on Users and their Goals

CRUD Use Cases

- A CRUD use case is a Create, Read, Update, or Delete use case
- Remove CRUD use cases if they are pure data-management use cases that do not provide results that are of value to actors

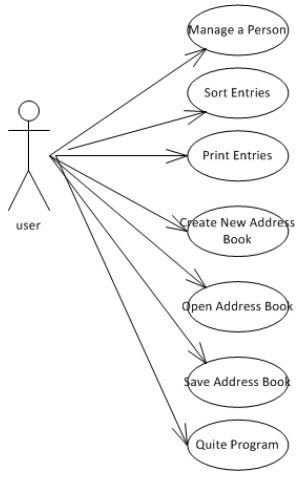


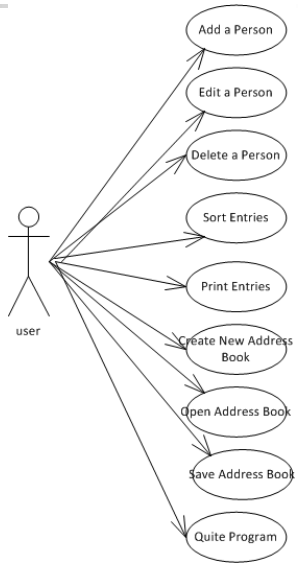
- **Do not confuse use cases with functions**
- **Focus on value**

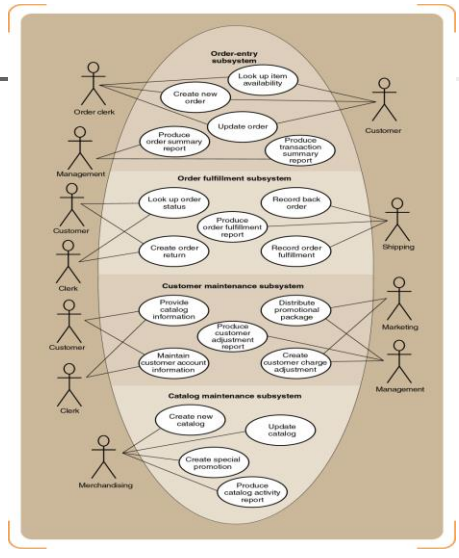
Exercise: Address Book Requirements

- Suppose we want to develop an addressbook SW that can be used to maintain contact information of peope. An address book holds a collection of entries, each recording a person's first and last names, address, phone numbers, and email addresses.
- In addition to maintain the entries, we should be able to sort the entries in the address book alphabetically by last name or by ZIP code. It must be possible to print out all the entries in the address book in "mailing label" format.

Use Case Diagrams for Address Book





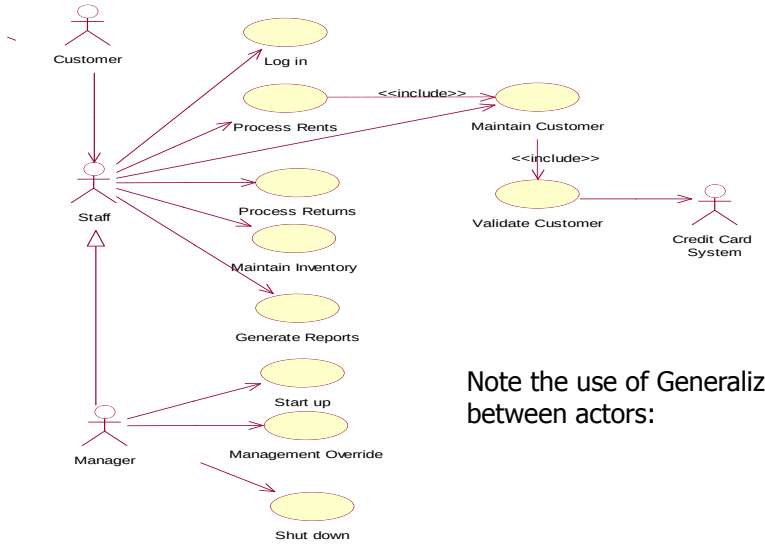


A Use Case Diagram of the Customer Support System

Example: Video Rental System

- This problem is about a small, local video rental store (VRS). The problem will be limited to rental, return, management of inventory (add/delete new tapes, change rental prices, etc.) and producing reports summarizing various business activities. The rental items of the store are limited to video tapes. Customer ID number (arbitrary number), phone number or the combination of first name and last name are entered to identify customer data and create an order. The bar code ID for each item is entered and video information from inventory is displayed. The video inventory file is decreased by one when an item is checked out. When all tape IDs are entered, the system computes the total rental fee, and payments are processed. A return is processed by reading the bar code of returned tapes. Any outstanding video rentals are displayed with the amount due on each tape and a total amount due. The past-due amount must be reduced to zero when new tapes are taken out. For new customers, the unique customer ID is generated and the customer information is entered into the system. Videos are stacked by their category such as Drama, Comedy, Action, etc. Any conflict between a customer and computer data is resolved by the store manager. Rental fees can be paid by either cash, check or a major credit card. Reporting requirements include viewing customer rental history, video rental history, and titles by category, top ten rentals, and items by status, and overdue videos by customers and outstanding balances by customers.

Use Case Diagram of VRS



Note the use of Generalization between actors:

Exercise: Develop a Use Case Diagram (Restaurant Order System: ROS)

The *Fancy Eight* is a newly opened five star restaurant in Walnut St. , Philadelphia. We would like to develop an automated restaurant order system (ROS) for *Fancy Eight* . Initially, we will focus on only orders, payments, and menus. That is, a customer is assigned to a table by a hostess. A hostess may reserve tables for customers. A waiter or waitress will bring a menu, then the customer picks any number of courses, which may include any combination of appetizer, salad, soup, wine, main dish, drinks, and desserts. After the meal, the waiter or waitress brings a bill. The customer can pay by cash, check or a credit card. The owner may add/delete menus and change prices occasionally. The current system will not include any functionalities on inventory or purchase of supplies yet.

Show a Use Case Diagram for ROS

Exercise: Develop a Use Case Diagram for Online Ticketing System (OTS)

The purpose of this system is to create a software system that will be able to purchase tickets for users. The system should help the customers find events they want and purchase one or more tickets. A customer can be a guest user for one time purchase or create an account if one desires to be a regular customer. The system would display available events and its associated tickets for each event, help in selecting proper seats depending on time and location, charge the customer for tickets. OTS accepts only credit cards. A customer can choose a different delivery method such as express, USPS, or pick up at the counter. After the purchase, an email notification is sent to the customer with the purchase and payment information. The office of OTS can create a new event, close an event that are passed, and mark the event if no more seats are available. If customers want to return a ticket, they need to contact the office to get credit. If customers lose a ticket, they also contact the office. The office also frequently creates hot tickets for some events to increase sales. The system could be similar to the ticketing system of Stub Hub or Ticketmaster. You may add and unstated, but desired features to OTS.

Review

- What are two components of the use case model?
- What is a use case?
- What is an actor?
- What are properties of a *good* use case?
- What are the 3 tests for testing the validity of a use case?
- What is the relationship between a use case and scenarios?
- What requirements does use cases define?
- What are the 4 steps of defining a use diagram?
- What is the meaning of generalization between two actors?
- Most of all, learn to create a good use case diagram with the correct notation.