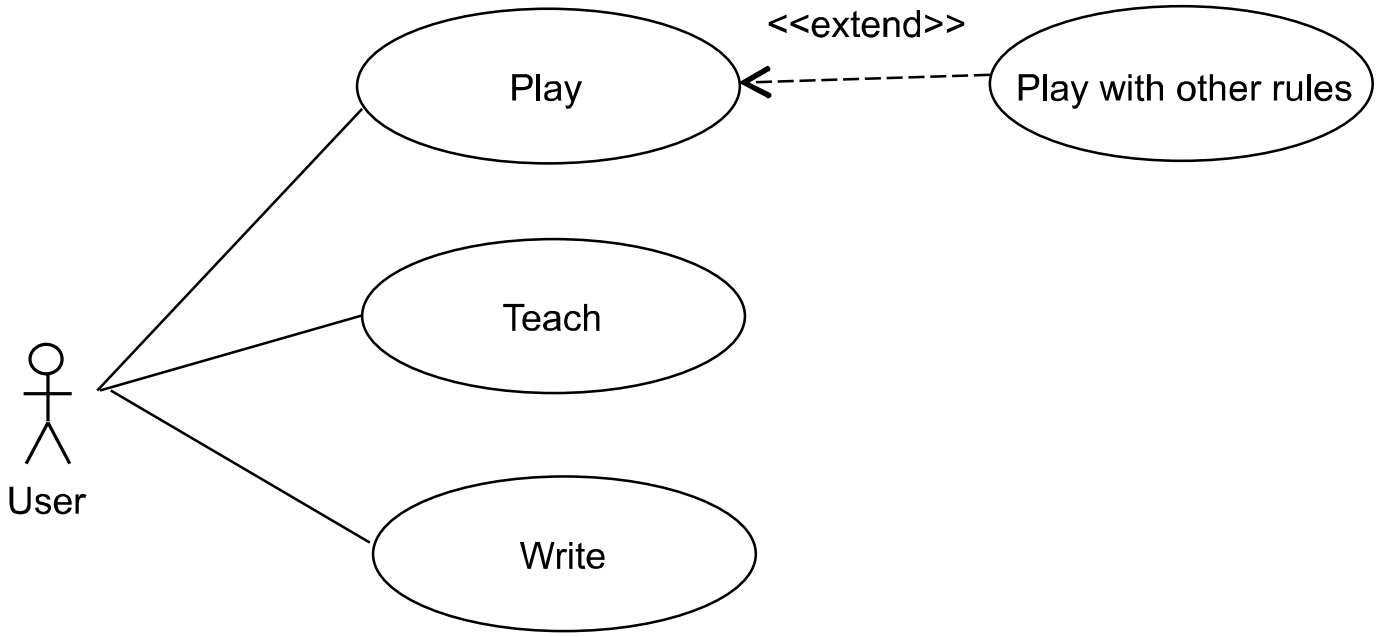
Goals and responsibilities

- The very same chess program, with identical structure and behavior, could be used with a different goal?
- For instance, could it be used to learn to play chess? Responsibility of the program: teach chess
- Or to write a chess book, like a chess game editor? Responsibility of the program: write chess texts
- Or to play a game of loser's chess (where who is checkmated wins)? Responsibility: play games with rules slightly different from chess

Each responsibility corresponds to (at least) a use case

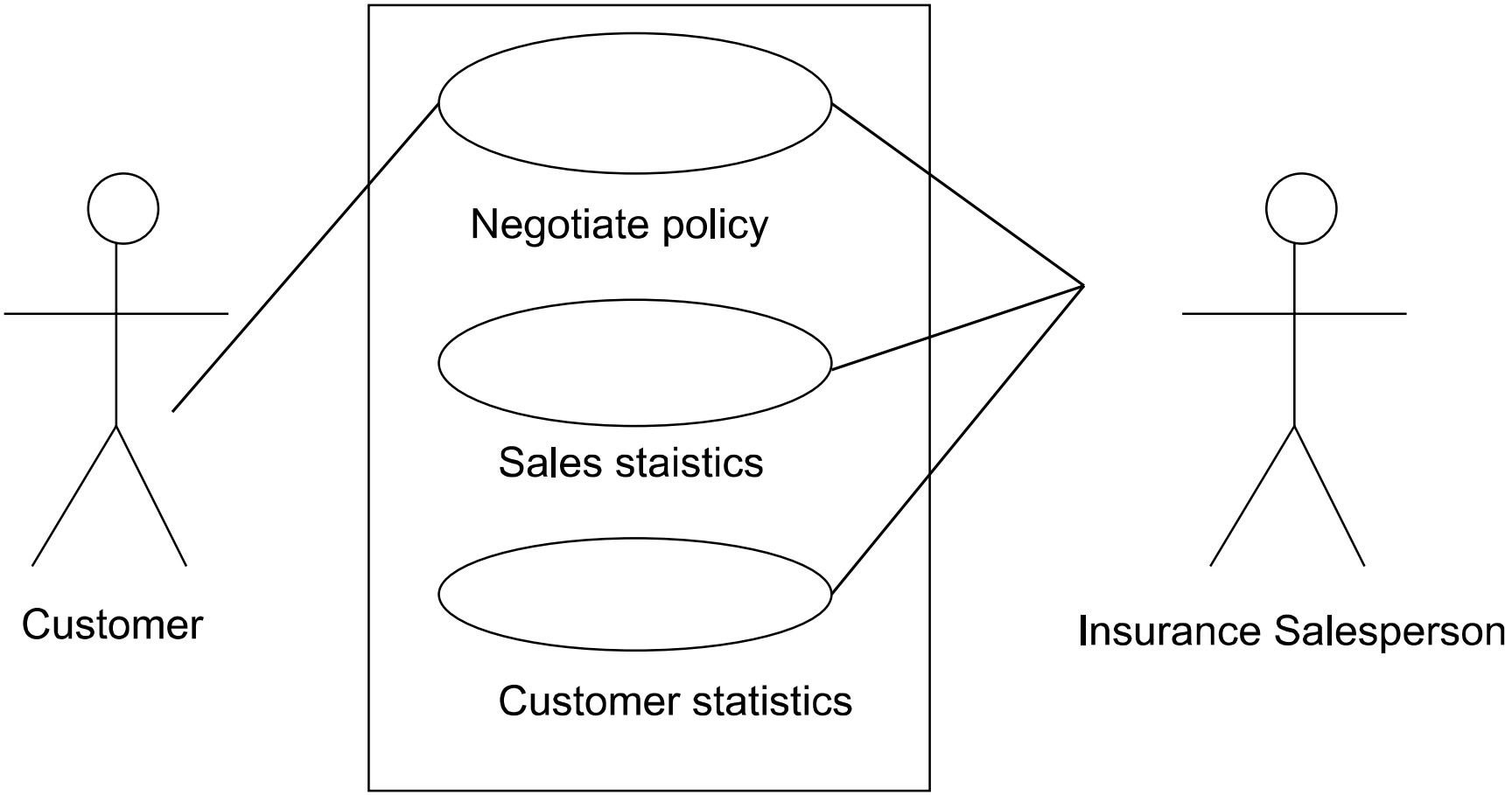
From responsibilities to use cases



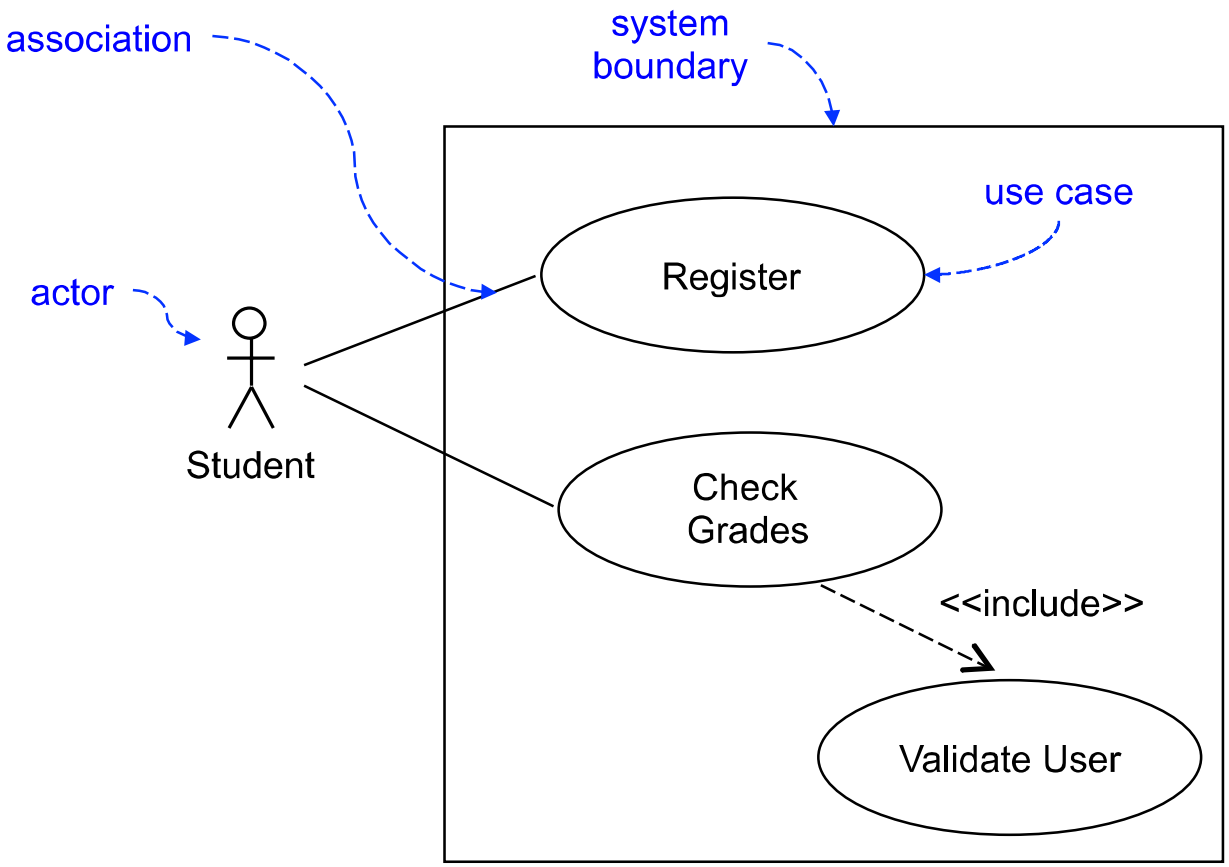
Use Case diagram

- It describes the externally observable behavior of a system, as related to **requirements**
- It describes the main interactions between the system and external entities, including users and other systems
- It is a summary of the main scenarios where the system will be used
- It describes the main user roles

Example


Negotiate policy
Sales staistics
Customer statistics
Customer
Insurance Salesperson

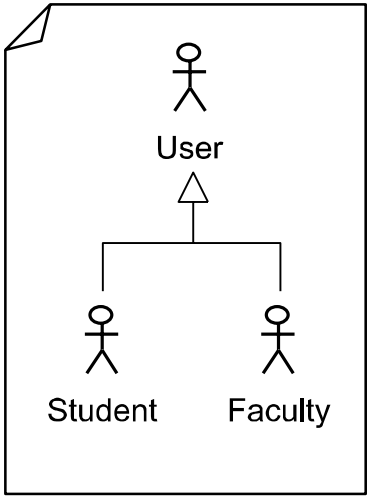
Use Case: elements

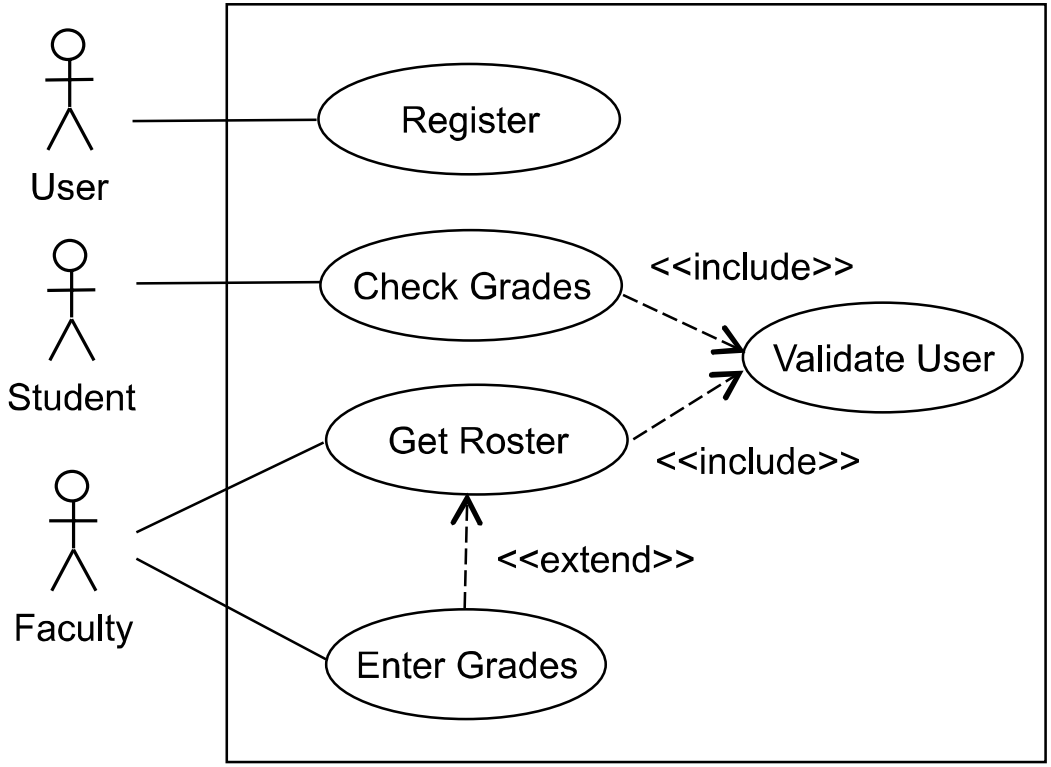


Elements of a Use Case Diagram

- **Actor**:
  - Represents a role played by external entities (humans, systems) that interact with the system
- **Use case**:
  - Describes what the system does (i.e., functionality)
  - Scenario: sequence of interactions between the actors and the system
- **Relationships**:
  - Association between actors and use cases
  - Extension (or generalization) among actors
  - Dependency among use cases: *include* and *extend*

Example


User
Student
Faculty


User
Student
Faculty
Register
Check Grades
Get Roster
Enter Grades
Validate User
<<include>>
<<include>>
<<extend>>

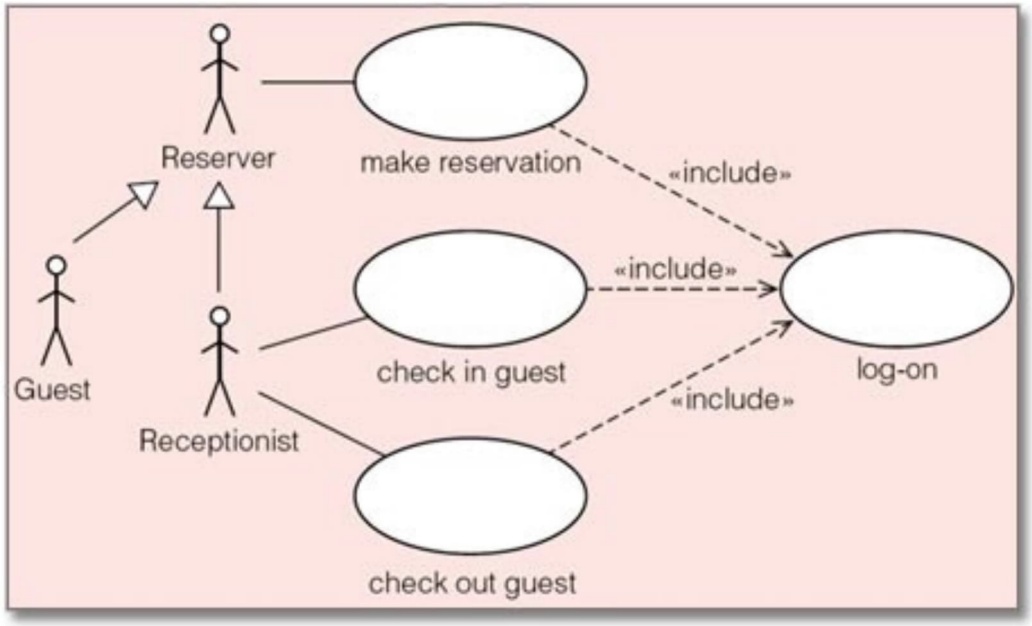
Use Case Scenario

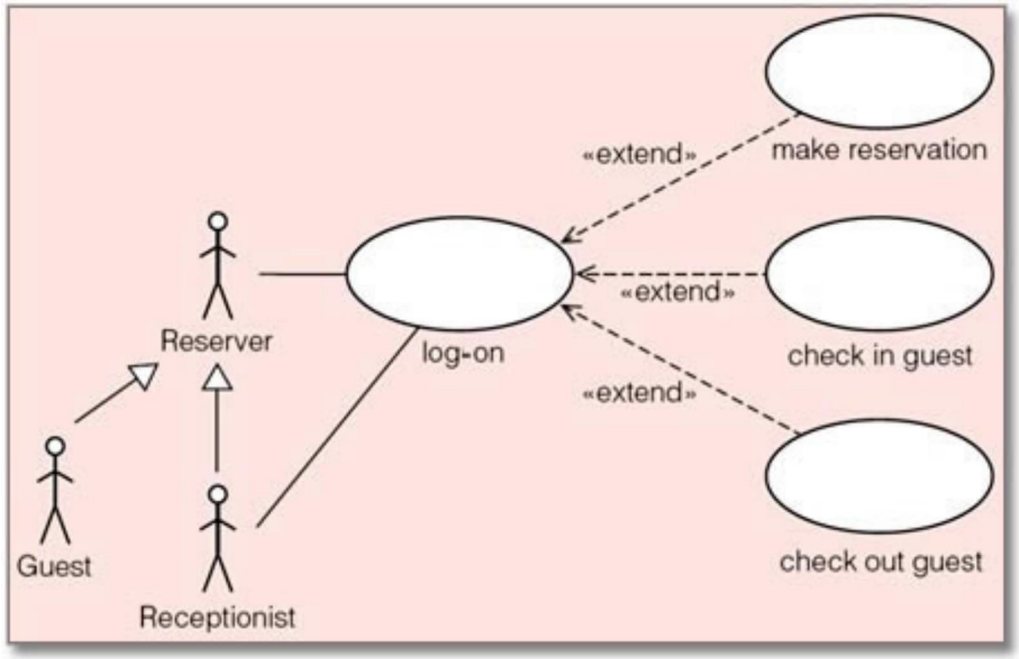
Use Case: Check Grades	
Description: View the grades of a specific year and semester **Actors**: Student **Precondition**: The student is already registered **Main scenario:**	
User	**System**
3. The user enters the year and semester, e.g., Fall 2013.	1. The system carries out “Validate User”, e.g., for user “miner” with password “allAs”. 2. The system prompts for the year and semester. 4. The system displays the grades of the courses taken in the given semester, i.e., Fall 2013.
Alternative: The student enters “All” for the year and semester, and the system displays grades of all courses taken so far. **Exceptional:** The “Validate User” use case fails; the system repeats the validation use case.	

<<extend>> vs <<include>>

- A use case B is included in use cases C and D when these have some common steps represented by B
- A use case B extends a use case C when B applies optionally, under some condition (usually specified in the scenario)

- Note: the lower diagram is formally correct but should be avoided, because the main functions should NOT be described as extensions of logon





Exercise

Draw a use case diagram and a related scenario for the following situation:

- A user can borrow a book from a library;
  - extend it with borrowing a journal
- a user can give back a book to the library
  - including the use case when the user is identified

Exercise: include or extend?

Main use cases: a customer buys something (eg. a book) from a virtual store like Amazon

- The user must be identified
- The book is not currently available, delayed delivery
- When the book is received the service must be graded
- The book is delivered via air mail
- The book is an ebook and can be delivered via Internet