

# Software Process II

Week 3

# Announcement

- Midterm I
  - 1:00 – 1:50 pm Wednesday 23<sup>rd</sup> February
  - Ch. 1, 2, 3 and 26.5
  - Hour 1, 6, 7 and 19 (pp.331 – 335)

# Agenda (Lecture)

- Study software process models
  - Waterfall
  - Prototyping
  - Incremental
  - UP
  - Spiral
  - Agile
  - XP
  - PSP
  - TSP
  - CMMI

# Agenda (Lab)

- Use case descriptions
- Present project proposals
- Weekly progress report
- Hour 7 quizzes (page 120)
- Submit the report, proposal and the answers of the quizzes by the end of the Wednesday lab session.

# Weekly Progress Report

- From now on, each team is required to submit a weekly project progress report to the instructor by the end of the Wednesday lab session. The report should be typed up and should include
  - The team name and a list of team members' names
  - A list of activities that have done in the previous week and the names of the corresponding contributors
  - A list of activities that will be conducted next week

# Team Lab Assignment #2

- Submit the first version of a use case diagram for your group project
  - Submit a use case diagram.
  - Make slides for presentation
- Due date
  - The beginning of the 2/14 lab session

# Team Homework Assignment #3

- Study PSP, TSP and CMMI and prepare for presentation slides.
- Presentation slides should include, description, visual representation (figure), advantages and disadvantages of each process model
- Due date is by 1:00 pm on February 14<sup>th</sup>

# Use Case

- Use cases are a way to capture system functionalities (i.e., functional requirements)
- Based on use case diagrams and their associated user case descriptions,
  - The rest of UML diagrams are developed.
  - The functions of software products are tested.
- Components
  - Diagrams
  - Descriptions

# Use Case Diagrams / Descriptions

- Use case diagrams show use cases, actors and relations among them.
- Use case descriptions address in details what the system (software product) shall do for the actor to achieve a particular goal (functionality).

# Use Case Development Process (1)

1. Find actors and use cases, and draw a draft of a use case diagram
  - GUI might be helpful for identifying interfaces between user(s) and the system, which initiate functions (use cases)

# Use Case Development Process (2)

2. Refine iteratively a use case diagram by considering relationships between use cases and actors, and between use cases, and between actors
3. Develop each use case (starting with the priority ones) by creating its use description

# Use Case Tutorial - Use Cases

- Represent a distinct functionality for a system
- Each use case must have a name describing the function
- Use an oval with the name of the use case

# Use Case Tutorial - Actors

- A use case must be initiated by someone or something outside the scope of the use case
- An actor does not need to be a human user; any external system or element outside of the use case may trigger the use case
- An actor can be shown with a stick figure with the name of the actor written near the icon

# Use Case Tutorial - Relationships (1)

- An actor is associated with one or more use cases
- A relationship between an actor and a use case indicates the actor initiates the use case, the use case provides the actor with results

# Use Case Tutorial - Relationships (2)

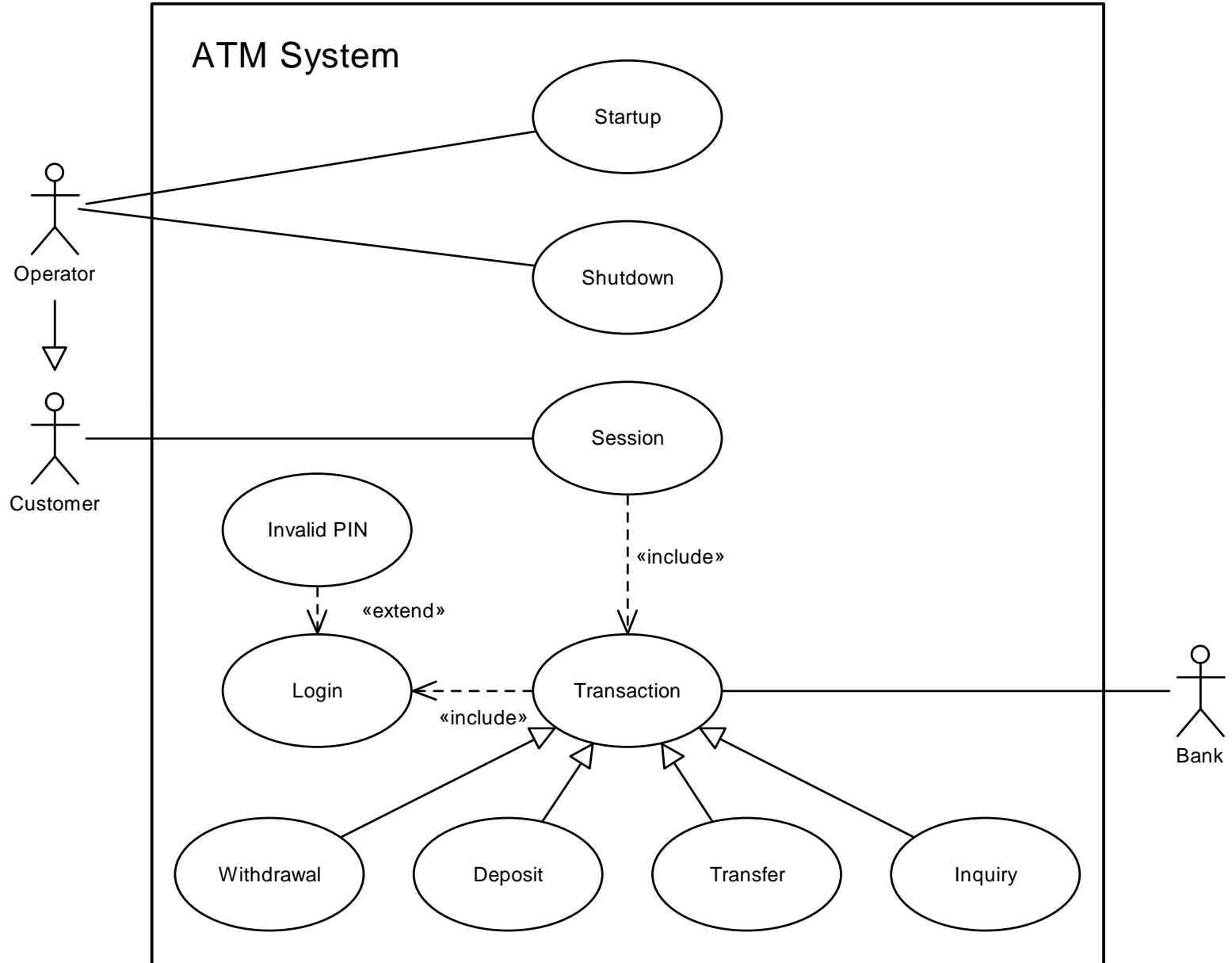
- An association is shown as a solid line between an actor and a use case
- Other types of relationships
  - Actor and use case ***generalization***
  - Use case ***include***
  - Use case ***extend***

# Use Case Descriptions (1)

- Use case name with a use case ID
- Characteristic information (goal, pre-condition, successful end condition, primary actors)
- Main (primary) scenario (“normal” messages flows between an actor and a use case)

# Use Case Descriptions (2)

- Alternative scenario (“exceptional” or “conditional” workflows between an actor and a use case)
- Utilizing other use cases, if necessary



# Use Case Description (3)

## UC1: Startup

### Characteristic Information

|                              |  |
|------------------------------|--|
| <b>Goal</b>                  | Power-up and initialize the ATM            |
| <b>Pre-Condition</b>         | ATM must be in the OFF mode                |
| <b>Success End Condition</b> | ATM is powered up and has been initialized |
| <b>Primary Actor</b>         | Operator                                   |

# Use Case Description (4)

## Primary Scenario

| Step | Actor/System | Action Description       |
|------|--------------|--------------------------|
| 1    | User         | Push the power on button |
| 2    | ATM          | Perform a self-test      |
| 3    | ATM          | Set the ATM in IDLE mode |
| 4    | ATM          | Run the clock            |

## Alternative Scenario

| Step | Condition            | Action Description  |
|------|----------------------|---|
| 2a   | self-test fails      | Set an alarm and notify the operator to correct the problem |
| 3a   | Mode setting failure | Set an alarm and notify the operator to correct the problem |
| 4a   | Clock failure        | Set an alarm and notify the operator to correct the problem |