MIS
(Management Information System)
(21-972)

Department of Industrial Engineering
Sharif University of Technology

Session #15

Course Description

- **Instructor**
  
  Omid Fatahi Valilai, Ph.D. Industrial Engineering Department, Sharif University of Technology
  
  Email: Fvalilai@sharif.edu, Tel: 021-6616-5706
  
  Website: http://sharif.edu/~fvalilai

- **Class time**
  
  Saturday-Monday 10:30~12:00

- **Course evaluation**
  
  - Mid-term  (20%)
  - Final exam  (20%)
  - Quiz  (10%)
  - Exercise-Projects  (30%)
Course Description (Continued …)

- Mid-term session:
  - Saturday, 7th, Azar 1394
- Final session:
  - Monday, 28th, Dey 1394
- Reference:

Course Description (Continued …)

- Reference:
  - William S. Davis, David C. Yen, “The information system consultant’s handbook: system analysis and design”, 2010, Taylor and Francis
  - Gabriele Piccoli; “Information systems for managers: texts & cases “, 2007, John Wiley & Sons Inc
Course Description (Continued..)

Contents:
- Introduction to Systems Analysis and Design
- Analyzing the Business Case
- Managing Systems Projects
- Requirements Modeling
- Data and Process Modeling
- Object Modeling
- Development Strategies
- User Interface Design
- Data Design
- System Architecture
- Managing Systems Implementation
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
    - RUP is made up of three model types:
      - Business system models ---- Use Case Diagrams
      - Static structure models ---- Class Inheritance Diagrams
      - Dynamic behavior models ---- State Transition Diagrams

Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
    - Use Case Modeling
      - A use case represents the steps in a specific business function or process.
        - An external entity, called an actor, initiates a use case by requesting the system to perform a function or process. The UML symbol for a use case is an oval with a label that describes the action or event.
        - The actor is shown as a stick figure, with a label that identifies the actor's role.
        - The line from the actor to the use case is called an association, because it links a particular actor to a use case.
Object Modeling

- Object Modeling with the Unified Modeling Language
- Object-Oriented Models
- Use Case Modeling

ADD NEW STUDENT Use Case

<table>
<thead>
<tr>
<th>Name:</th>
<th>Add New Student</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor:</td>
<td>Student/Manager</td>
</tr>
<tr>
<td>Description:</td>
<td>Describes the process used to add a student to a fitness-class</td>
</tr>
<tr>
<td>Successful completion:</td>
<td></td>
</tr>
<tr>
<td>1. Manager checks FITNESS-CLASS SCHEDULE object for availability</td>
<td></td>
</tr>
<tr>
<td>2. Manager notifies student</td>
<td></td>
</tr>
<tr>
<td>3. Fitness-class is open and student pays fee</td>
<td></td>
</tr>
<tr>
<td>4. Manager registers student</td>
<td></td>
</tr>
<tr>
<td>Alternative:</td>
<td></td>
</tr>
<tr>
<td>1. Manager checks FITNESS-CLASS SCHEDULE object for availability</td>
<td></td>
</tr>
<tr>
<td>2. Fitness-class is full</td>
<td></td>
</tr>
<tr>
<td>3. Manager notifies student</td>
<td></td>
</tr>
<tr>
<td>Precondition:</td>
<td>Student requests fitness-class</td>
</tr>
<tr>
<td>Postcondition:</td>
<td>Student is enrolled in fitness-class and fees have been paid</td>
</tr>
<tr>
<td>Assumptions:</td>
<td>None</td>
</tr>
</tbody>
</table>
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
  - Use Case Modeling

Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
  - Use Case Modeling
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
    - Class Diagrams
      - A class diagram shows the object classes and relationships involved in a use case.
      - Like a DFD, a class diagram is a logical model, which evolves into a physical model and finally becomes a functioning information system.
      - In a class diagram, each class appears as a rectangle, with the class name at the top, followed by the class's attributes and methods.
      - Lines show relationships between classes and have labels identifying the action that relates the two classes.

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
    - Class Diagram

<table>
<thead>
<tr>
<th>UML Notation</th>
<th>Nature of the Relationship</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0..*</td>
<td>Zero or many</td>
<td>Employee</td>
<td>Payroll Deduction</td>
</tr>
<tr>
<td>0..1</td>
<td>Zero or one</td>
<td>Employee</td>
<td>Spouse</td>
</tr>
<tr>
<td>1</td>
<td>One and only one</td>
<td>Office Manager</td>
<td>Sales Office</td>
</tr>
<tr>
<td>1..*</td>
<td>One or many</td>
<td>Order</td>
<td>Item Ordered</td>
</tr>
</tbody>
</table>
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
  - Class Diagram

A sequence diagram is a dynamic model of a use case, showing the interaction among classes during a specified time period.

A sequence diagram graphically documents the use case by showing the classes, the messages, and the timing of the messages.

Sequence diagrams include symbols that represent classes, lifelines, messages, and focuses.
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
  - Sequence Diagram

Object Modeling
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
    - State Transition Diagram
    - A state transition diagram shows how an object changes from one state to another, depending on events that affect the object.
    - All possible states must be documented in the state transition diagram.
    - In a state transition diagram, the states appear as rounded rectangles with the state names inside.
    - The small circle to the left is the initial state, or the point where the object first interacts with the system.
    - Reading from left to right, the lines show direction and describe the action or event that causes a transition from one state to another. The circle at the right with a hollow border is the final state.
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
    - Activity Diagrams
      - An activity diagram resembles a horizontal flowchart that shows the actions and events as they occur.

- Activity diagrams show the order in which the actions take place and identify the outcomes.

- Activity diagrams also can display multiple use cases in the form of a grid, where classes are shown as vertical bars and actions appear as horizontal arrows.
Object Modeling

- Object Modeling with the Unified Modeling Language
  - Object-Oriented Models
  - Business Process Modeling
  - In addition to sequence diagrams and activity diagrams, you can use business process modeling (BPM) to represent the people, events, and interaction in a system.

- BPM initially as a requirements modeling tool, works well with object modeling, because both methods focus on the actors and the way they behave.

- In a typical BPM diagram, the outside rectangle is called a pool, and designated swim lanes show specific actions and events. The swim lanes can interact when certain events