A Quick Review of UML

CS561

UML: The Unified Modeling Language

- A recognized international standard
- Available from http://www.uml.org
**What is UML?**

UML is a collection of OO design and specification techniques, most of which predate UML, with a standard notation.

**Functional Models**
- Model system functionality (i.e. use case diagrams)

**Object Models**
- Model internal structure of system (i.e. class diagrams)

**Dynamic Models**
- Model behavior/logic flow of system (i.e. sequence diagrams, state machine diagrams)

---

**What is the Purpose of UML diagrams?**

- Models and diagrams help us think through problems, document design decisions, and more efficiently communicate these.
- UML offers a standardized set of diagrams, and a standardized notation, making communication more efficient.
- UML is embodied in a number of tools
  - Diagramming (Día, Visio)
  - Model Transformation (ATL)
  - Code Generation (BOUML, Codegen)
  - Reverse Engineering (AgileJ, CodeLogic, Describe)
UML Diagrams

There are 13 types of diagrams:
- Structure diagrams
  - Class diagram
  - Component diagram
  - Composite structure diagram
  - Deployment diagram
  - Object diagram
  - Package diagram
- Behavior diagrams
  - Activity diagram
  - State machine diagram
  - Use case diagram
- Interaction diagrams
  - Communication diagram
  - Interaction overview diagram
  - Sequence diagram
  - UML timing diagram

http://www.agilemodeling.com/essays/umlDiagrams.htm

Use Case Diagrams

Used to organize use-cases

- Use cases. A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

- Actors. An actor is a person, organization, or external system that plays a role in one or more interactions with your system. Actors are drawn as stick figures.

- Associations. Associations between actors and use cases are indicated in use case diagrams by solid lines. An association exists whenever an actor is involved with an interaction described by a use case.

www.agilemodeling.com
Activity diagrams

Flow charts modeling logic of a single use case

Activity diagram (2)
**Class diagrams**

- A true classic – models domain & design
- Shows classes, their relations, operations, and dependencies

![Class diagram example](www.agilemodeling.com)

**Sequence diagrams**

- Aimed at modeling logic, or flow in your system; who interacts with what, and when
- Different levels
  - Actors
  - Components
  - Objects
- Time ↓
- Actors →

![Sequence diagram example](www.agilemodeling.com)
State Machine diagrams

- Models possible states of objects as well as the flow/interaction of the system.