



www.primaryview.org

from the notebooks of John Artim

Use Cases and Software Requirements Specification

Analysis and Specification Review

John M. Artim, ©2003, 2004. Permission to copy, modify, or distribute is granted.

Use Cases

- A use case model describes the work done in a domain
- Use cases emphasize the division of a system in the large into actors, their tasks, and the system under discussion (SUD) supporting those tasks
- Use case modeling is used to document functional requirements—those requirements that describe the work an SUD must do
- Use cases must be user-centric
 - That is, the focus must be on what the user and their organizations are trying to accomplish

Scenarios

- Scenarios define an individual use case by example
 - The scenario is one concrete example of how a use case can be done
 - This concreteness supports consensus among project stakeholders creating a useful sense of what is being accomplished

Concept Modeling

- Concept models provide consistent use of language in scenarios
- Consistency of scenario language greatly improves the ability of third-parties to interpret the meaning of use cases

Concept Modeling—Glossaries

- A glossary is the simplest form of concept model
- It provides a great deal of consistency for little effort
- Large glossaries or glossaries authored by many analysts can be costly and difficult to maintain

Concept Modeling—Object Models

- Object models provide a much more robust representation of the language used in scenarios
 - More robust because it highlights any ambiguity
 - And provides a basis for describing and talking about the ambiguity
- But object models take more time and resource to complete

State Diagrams

- When you have business entities that exhibit state-based behavior, describe that entity in a state diagram
 - Tie the state diagram to the class defining the entity (if you do a class model)
 - Cross-check the set of states you describe with the business process the entity must support

Activity Diagrams

- If you need to describe typical flow through a set of use cases
- If you need to describe all flow through a set of use cases
- This is helpful for describing business processes and for describing user tasks

Always Maintain Strong Separation

- Between Entities, Tasks, and Interfaces (presentation)
 - Collect business-concept, business-activity, and presentation requirements in separate classes
 - This is the basic three-way category split in requirements specification!
- Among Entities
 - Don't blur the boundaries between one concept and the next