

Course: Effective Use Case Development

Goals

Without proper capture of requirements software projects have a high probability of failure. This 3-day course presents the use case technique which has become the norm for functional requirements capture in object-oriented projects, yet can be used just as successfully with procedural development projects. Structured around a problem-driven approach, students will learn use case techniques and concepts in a "just-in-time" manner, emphasizing the semantics of each technique and not just its notation. Numerous examples, and two, parallel case studies let students see how a concept is described in a use case model, and then immediately apply their understanding.

At the end of the course, the student will be able to:

- Recognize and understand the UML use case notation and semantics.
- Describe the iterative process for developing use cases.
- Describe the degree of detail needed to write successful use cases.
- Recognize improper use case descriptions.
- Understand how to apply use cases directly to software testing.

Duration

Three days.

Prerequisites

Experience in requirements gathering, or systems analysis is desirable, but not mandatory.

Cost

Please call **1-610-831-1151** for public enrollment and private, on-site pricing.

Description

This 3-day course is designed to provide students with intensive, practical training in the concepts of requirements specification through use cases. During the course the students write at least three non-trivial use cases. At course completion, students will have a comprehensive understanding of use cases, how to write useful and effective use cases, and both their value and limitations in either procedural or object-oriented development lifecycles.

Topics

<p>Project Failure and Success Factors Why Use Cases? The many dimensions of requirements</p> <ul style="list-style-type: none"> - Functional requirements - Non-functional requirements - Use Cases are not diagrams - Where do use cases fit? <p>Dimensions of use case descriptions</p> <ul style="list-style-type: none"> - Use cases and traditional specifications - Why use cases alone are not enough! <p>Discovering use cases</p> <ul style="list-style-type: none"> - Using an event analysis model - Discovering actors & major use cases <p>The UML Use Case Diagram.</p> <ul style="list-style-type: none"> - Actors - Actor-to-use case associations - Use case-to-use case associations - <<include>> and <<extend>> - Generalization - Does the diagram have value? <p>Distinguishing the two different types of actors</p>	<p>Writing use case descriptions.</p> <ul style="list-style-type: none"> - Style and templates - Scope and presentation format - Use cases must have goals - System-level vs. business-level use cases - Adding exceptions to use cases - KISS: Keep It Simple...and Succeed <p>The power of "essential" use case descriptions</p> <ul style="list-style-type: none"> - Procedural use cases - State-based use cases <p>Major mistakes in use case development</p> <ul style="list-style-type: none"> - Use case pitfalls and abuses - Where UML can lead you astray - Just who should write the use cases? <p>The role of use cases in testing</p> <ul style="list-style-type: none"> - Use cases are directly testable - Use cases and scenarios - Scenarios and test cases <p>CASE tools and use cases</p> <p>Wrap-up</p>
---	--

Audience

Business or system analysts, technical managers, and software developers who wish to learn techniques for capturing requirements for software system development.

**For more information about this course or other courses please contact
Nazzaro & Associates at 1-610-831-1151.**