ProTech
protechtraining.com

"Charting the Course ...

...to your Success!"

Gathering and Documenting User Requirements with Use Cases

Course Summary

Description

This course provides students with the skills and knowledge necessary to mange both developing new functionality and supporting legacy systems changes using use cases. At least 50% of the course will be devoted to hands on work where students can develop their analytical skills and use case writing skills.

The course teaches students how to write and use cases for various purposes, and teaches all the forms that are most commonly used in industry today. The students will be walked through the typical kinds of situations that use cases occur: adding new functionality, changing existing functionality and modifying use cases to reflect changes in system processes.

Students are also taught how to work with the system process model that is the core of a legacy system and how to ensure that their use case development and changes integrate with the system process model. The course emphasizes making testable use cases and the developing testing strategies for changes in use cases.

Topics

- System Lifecycles and Use Cases
- Understanding Requirements
- Stakeholder Requirements
- Modeling Stakeholder Requirements
- Introduction to Use Cases in Depth
- Developing a Use Case for Requirements
- Developing the Use Case Body
- Use Case Elaboration
- Developing Use Cases
- Developing with Use Cases
- Use Cases for User Change Management
- Use Cases for Stakeholder Change Management
- Reverse Engineering Legacy Systems
- Putting It All Together

Prerequisites

There are no prerequisites for this class.

Duration

Four days

ProTech
protechtraining.com

"Charting the Course ...

...to your Success!"

Gathering and Documenting User Requirements with Use Cases

Course Summary

I. Introduction
- A. Complementary approaches to Software Engineering: Structured and OO
- B. Similarities between structured and OO
- C. Moving structured skills to OO skills
- D. The kinds of systems the structured approach deals with
- E. The kinds of systems the OO approach deals with
- F. Issues involved in migrating and supporting legacy mainframe systems

II. System Lifecycles and Use Cases
- A. Difference between a system life cycle and a SDLC
- B. The generic software development and software support processes
- C. Development and Support as change management
- D. Why we use use cases
- E. Different roles use cases play during the system lifecycle
- F. Current caveats for those using use cases – myths and pitfalls
- G. RUP, Rational Software and use cases – how they fit together

III. Understanding Requirements
- A. The purpose of requirements – The stakeholder's views
- B. Requirements versus features versus specifications
- C. Types of requirements: functional versus non-functional
- D. Functional requirement types: Use case versus features
- E. Sources of requirements: users and non-users
- F. The requirements process
- G. Moving from requirements to specifications.

IV. Stakeholder Requirements
- A. The concept of a stakeholder
- B. Why stakeholders requirements are critical
- C. Business requirements – safeguarding stakeholder requirements
- D. Kinds of stakeholder requirements
- E. How stakeholder requirements show up in the system

V. Modeling Stakeholder Requirements
- A. Business process models and business rules
- B. Process modeling activity diagrams
- C. (or Process modeling with business process notation)
- D. Business objects and entities
- E. Business/enterprise modeling

VI. Introduction to Use Cases in Depth
- A. Use cases versus features -- making the call
- B. Actors, systems and goals
- C. Use cases and story telling
- D. What use cases do for us
- E. The necessary role of use cases
- F. Parallel development – use cases and business model
- G. Integration of use cases with system processes

VII. Developing a Use Case for Requirements
- A. Use cases document existing interactions
- B. Use cases are user stories and scripts
- C. The importance of keeping the system a black box
- D. How use cases guide development of a solutions
- E. Summary: The starting form of a use case
- F. Prioritization of use cases
- G. Iterative development of a use case

VIII. Developing the Use Case Body
- A. The RUP textual form
- B. The dialog textual form
- C. The essential textual form
- D. Scope of the use case
- E. Scenarios and paths
- F. The main success scenario
- G. Alternative flows
- H. Exceptions
- I. Generalization, extension and inclusion

IX. Use Case Elaboration
- A. Use case meta information
- B. Secondary actors and channels
- C. Pre and post conditions
- D. Triggering events and success outcomes
- E. Open issues
- F. Managing the use case model

Gathering and Documenting User Requirements with Use Cases

Course Summary (cont'd)

X. Developing Use Cases
- A. Domain experts
- B. Field observations
- C. How to decide if a use case is "good"
- D. Use case pitfalls
- E. Problems with the textual form

XI. Developing with Use Cases
- A. Developing implementable and testable use cases
- B. Developing system use cases
- C. White boxing use cases
- D. Use cases / features matrices
- E. Testing use cases
- F. Developing the use case process model

XII. Use Cases for User Change Management
- A. Using Use cases to document change requests
- B. Use case deltas – what is versus what is requested
- C. White box comparison and use case to system mappings
- D. Reworking the use case
- E. Testing the change and developing the test suite
- F. Managing the user and the change process

XIII. Use Cases for Stakeholder Change Management
- A. Using the process model to document the change request/
- B. Identifying the affected use cases
- C. Developing the change use case
- D. Testing the change use case

XIV. Reverse Engineering Legacy Systems
- A. Developing enough of the process model to manage change requests
- B. Actor identification
- C. Splitting off use cases from the process model
- D. Introduction to refactoring
- E. Introduction to system partitioning for re-engineering

XV. Putting It All Together
- A. Developing a use case standard and protocol
- B. Managing and organizing use cases
- C. Making use cases as useful as possible through the SLC
- D. Benchmarks, process and quality tips and suggestions