

Writing Quality Requirements

Duration

2 days

Instructor

Karl E. Wiegers

Class Limit

20 students

Prerequisite

None

Price

On-site

Please contact SPC for pricing (contact information on page 2)

Public Training

\$1,495 (2 days)

Materials Provided

- Student manual containing the course slides
- Student handouts with class exercises
- *More About Software Requirements*, by Karl E. Wiegers (Microsoft Press, 2006)

Though everyone in software development knows about requirements, more often than not software is delivered that satisfies all the project specifications yet fails to meet most (or any) of the customers' expectations. Without formal, verifiable requirements, and a system for managing them, the gap between what developers think they're supposed to build and what customers think they're going to get can easily become insurmountable.

Writing Quality Requirements is a two-day workshop intended to help people who perform the requirements analyst role on a software project become more proficient at specifying high-quality requirements of various types.

Participants will spend considerable time in the class reviewing less-than-perfect requirements for problems and rewriting them. Students are asked to bring some requirements from their own project that they can then review and try to improve. Another important aspect of the workshop is to learn about the importance of peer reviews of requirements and to get some practice in examining a colleague's work and providing constructive review feedback.

While participants will not be expert requirement writers after this workshop—that takes a lot of practice and helpful review feedback from others—they will have a good sense of what constitutes high-quality requirements of various kinds and how to write them.

On completion of this seminar, participants will be able to:

- Describe the characteristics of high-quality requirements
- Evaluate functional requirements and quality attributes
- Review and provide feedback on requirements written by other analysts
- Document data dictionary entries and business rules
- Describe the components of a well-structured use case
- Derive functional requirements from the components of a use case
- Write functional requirements and quality attributes that are more precise, richer in detail, less ambiguous, and more actionable than before

Intended Audience

This course is ideally suited to anyone who has to document, analyze, or use requirements on a software project.

TRAINING

Writing Quality Requirements

Instructor

Karl E. Wiegers is a leading speaker and consultant in the requirements engineering and software process improvement arenas. As Principal Consultant with Process Impact, he conducts training seminars for corporate and government clients worldwide and speaks at numerous industry events. Previously, Karl spent 18 years at Eastman Kodak Company.

The author of four books and over 160 articles, Karl has twice won the Software Productivity Award, which honors excellence in productivity-enhancing products and books. Karl received a B.S. degree in chemistry from Boise State College, and a M.S. and Ph.D. degrees in organic chemistry from the University of Illinois. He is a member of the IEEE, IEEE Computer Society, and ACM.

Outline

Group Discussion

- Your requirements-writing problems

Software Requirements Refresher

- Requirements definitions
- Three levels of software requirements: business, user, and functional
- Characteristics of high-quality requirements
- Tips for writing quality requirements

For more information on this or other SPC Springboard courses, please visit www.spcspringboard.com or e-mail SPC at info@spc.ca

Software Productivity Center Inc.
Suite 460 - 1122 Mainland Street
Vancouver, BC V6B 5L1

Vancouver: 604.662.8181 Toll Free: 1.877.548.1948

Fax: 604.689.0141

Outline Cont'd

Software Peer Reviews

- Objectives of software peer reviews
- Types of peer reviews
- Selecting an appropriate review technique
- Choosing reviewers for requirements
- Reviewing requirements effectively

The Requirements Baseline

- Definition of a requirements baseline
- Management review of requirements
- What does "sign-off" mean?

Elements of Requirements Style

- Structures for writing functional requirements
- Common types of requirements ambiguity
- Several techniques for avoiding ambiguity
- Watching out for solution ideas
- Representing requirements using other techniques besides natural language text

Sample Requirements to Evaluate

- Practice session: Critique and improve upon several functional requirements from different projects
- Software quality attributes overview
- Practice session: Critique and improve upon several quality attribute requirements
- Practice session: Writing quality attributes for your project
- The data dictionary
- Practice session: Writing data dictionary entries for your project
- Business rules overview
- Practice session: Writing business rules for your project

Overview of Use Cases

- Definition of use cases and examples
- Preconditions and postconditions
- Alternative flows and exceptions
- From use cases to functional requirements
- Practice session: Writing functional requirements from a use case description

