

Software Quality Assurance

Length of Course: 3 Days

Description

This course provides immediately usable tools and techniques in the latest methods of software quality assurance (SQA) for accurate and thorough verification and validation of software and improved managerial control of software development and enhancement. It includes detailed coverage of software quality factors, the systems development life cycle and SQA factors, the methods, tools and techniques for measurement of software quality and quality control, the costs associated with quality (and not controlling it) at each phase of the systems development life cycle and various effective (SQA) guidelines and standards.

Objectives

At the completion of this course, the student will be able to:

- Understand and effectively apply software quality assurance (SQA) methods, tools and techniques
- Plan for, implement and manage the integrated software quality assurance function
- Provide the necessary software quality assurance steps, controls and results needed at each step or phase of the system's development life cycle to assure communication and satisfaction with both user/client and information systems personnel
- Evaluate how new technologies impact software quality assurance and the system's development life cycle and understand how to benefit from their application

Topics

- Software development life cycle-the product view
- Software configuration management (SCM)
- Software quality assurance
- Testing, validating and evaluating
- Special topics
- Planning, organizing, implementing and managing for SQA

Audience

Information systems managers, supervisors, project leaders, analysts, programmer/analysts, standards and procedures and software quality assurance personnel who are involved with software quality assurance.

Course Outline

I. Introduction and Overview

- A. The software industry today: the state and quality of the ark
- B. Software quality challenges and expectations
- C. Software engineering and SQA: definitions and key components
- D. Structured vs. traditional software development and enhancement and Software Quality Assurance (SQA)
- E. The systems development life cycle: steps or phases and results
- F. The tools, techniques and methods of SQA
- G. Productivity and SQA: the impact of application



II. Software Development Life Cycle: the Product View

- A. The software project infrastructure
- B. Systems development life cycle overview: its relationship to the project infrastructure and key SQA points
- C. Life cycle phases and quality factors
- D. Systems initiation and planning
- E. Analysis and requirements definition
- F. Software design
- G. Coding and unit testing
- H. Testing and integration of system(s)
- I. Production, use, evaluation and enhancement
- J. Quality gates: the different types and categories per type of systems development effort
- K. The concept and application of baselines
- L. SQA events, functions and deliverables

III. Software Configuration Management (SCM)

- A. Introduction and overview
- B. Management concepts, key points and necessary involvement
- C. The SCM plan and tools
- D. Configuration management
- E. Software configuration identification
- F. Software configuration control, auditing and status accounting
- G. Case studies and exercise

IV. Software Quality Assurance

- A. Software development and enhancement standards and the SQA function
- B. Additional SQA events and functions
- C. Testing, verification and validation
- D. Walkthroughs and inspections
- E. Software audits
- F. Management reviews
- G. Planning, budgeting and management of SQA
- H. Case histories and exercises

V. Testing, Validating and Evaluating

- A. SQA testing, validation and evaluation strategies: module, unit, system, integration and acceptance
- B. SQA metrics: measurements for and of effectiveness
- C. Software quality evaluation techniques
- D. Defect tracking: what to track where, analysis and improvement methods
- E. Application exercise

VI. Special Topics

- A. Project data: management, production and control
- B. Organization and personnel
- C. Effects of resource availability on quality
- D. Software reporting metrics

VII. Planning, Organizing, Implementing and Managing for SQA

- A. Tailoring the function
- B. Determining methods and tools
- C. Developing a SQA corrective action procedure
- D. Resource allocation
- E. Cases, exercises and examples



VIII. Review and Evaluation

- A. Developing, and/or updating, individual and organizational action plans
- B. Key points to remember and future information sources

