

Software Quality Assurance – A Life Cycle Approach

(3 Days)

This course is designed for Quality Assurance professionals and others who want to learn how to design and apply quality assurance throughout the software development lifecycle.

You will learn how to deliver quality software based on pre-defined quality success factors, how to measure the right things to achieve optimal results and how to perform quality control (testing and inspections) in the context of the software life cycle. You will learn the reasons SQA is needed, as well as the various SQA guidelines and standards used to achieve high-quality software.

To reinforce the learning objectives, you will work in teams to perform meaningful exercises that will give you practical experience in applying the materials. These exercises are performed at every step along the way in the workshop.

By attending this course, you will be able to understand both the theory and practical application of software quality practices. The course concludes with developing your own action plan for achieving software quality in your organization.

Who Should Attend

- QA Analysts
- QA Managers
- Test analysts
- Testers
- Project Leaders
- Stakeholders

Return on Investment

You will be able to:

- Understand and effectively apply software quality assurance (SQA) methods, tools and techniques
- Plan, implement and manage the software quality assurance function in your organization
- Provide the appropriate software quality assurance activities, controls and results at each step or phase of the system's development life cycle to assure communication and satisfaction with all stakeholders
- Evaluate how new technologies and projects impact software quality assurance and the system's development life cycle and understand how to benefit from their application

Course Topics

Module 1 – Introduction and Overview (2.5 hrs.)

- The Software Industry Today And Where We're At In Software Quality
- "State Of The Practice" In Software Quality
- Challenges And Expectations In Software Quality
- A Few Causes of Software Failure
- The Need for Testing
- The Role Of Testing and SQA In Software Development, Maintenance And Operations
- The Balance of QA and Testing
- How Much Testing is Enough?
- Key SQA terms
- Key Elements of SQA
- Structured vs. traditional software development and enhancement and Software Quality Assurance (SQA)
- The "V" Model of Building and Testing Software
- The Systems Development Life Cycle: Steps Or Phases And Results
- The Tools, Techniques And Methods Of SQA
- Productivity and SQA: How to Deliver Quality Software and Still Get Things Done on Time
- Where Defects Originate
- The Relative Cost of Fixing Defects
- How to Save Money in Software Development and Testing

Module 2 – The Role of Software QA (3 hrs.)

- The Software Project Infrastructure
- An Overview of Software Development Life Cycles
- Life Cycle Phases And Quality Factors
- Steps in Selecting an SDLC
- Phases in the Lifecycle
- Systems Initiation And Planning
- Analysis And Requirements Definition
- Software Design
- Coding And Unit Testing
- Testing And Integration Of System(S)
- Production, Use, Evaluation And Enhancement
- Quality Gates: The Different Types And Categories Per Type Of Systems Development Effort

- The Deming Workbench Model
- Steps in Creating Life Cycle Processes
- Establishing Life Cycle Processes
- Monitoring Life Cycle Processes
- Evaluating the Impact of Changes on Life Cycle Processes
- The Concept of Baselines
- The Application Baseline
- The Testing Baseline
- How Baselines Help You
- SQA Events, Functions And Deliverables

Module 3 – Software Configuration Management (3 hrs.)

- What is SCM?
- The Problem
- Characteristics of Good SCM Programs
- Why is SCM Important?
- What Happens When Good SCM Practices are Missing?
- The SCM Triangle
- Major Perspectives of SCM
- IEEE Standard 828-1990 for SCM Plans
- The SEI Perspective of SCM
- The SCM Plan And Tools
- Management Concepts, Key Points And Necessary Involvement
- SCM Staffing
- The Role of the SCM Manager
- The Key Activities of SCM
- Key SCM Terms
- A Basic CM Process
- How Much Process is Needed?
- Software Configuration Identification
- CM Baselines
- Version Control and Releases
- Software Configuration Control, Auditing And Status Accounting
- Case Studies And Exercise

Module 4 – Software Quality Assurance (4 hrs.)

- Software Development And Enhancement Standards And The SQA Function
- IEEE 12207 Life Cycle Processes and Roles

- What is Software Quality Assurance?
- The Dimensions of SQA
- The Components of SQA
- Additional SQA Events And Functions
- Testing, Verification And Validation
- Walkthroughs And Inspections
- Software Audits
- Management Reviews
- Planning, Budgeting And Management Of SQA
- The SQA Plan
- Case Histories And Exercises

Module 5 – Testing, Validation and Evaluation (4 hrs.)

- Software Testing Terminology
- Phases of Testing: Module, Unit, System, Integration And Acceptance
- Validation And Evaluation Strategies
- SQA and Test Metrics
- What Needs to be in Place to Capture Measurements and Metrics
- Testing Dashboards
- Tools and Techniques for Measuring Test Activities
- Prerequisites for Test Evaluation
- Software Quality Evaluation Techniques
- Defect Tracking: What To Track Where, Analysis And Improvement Methods
- Test Reporting – Defects, Status and Final Reports
- How to Use Charts and Graphs to Tell Your Information

Module 6 – Special Topics (2 hrs.)

- Project Data: Management, Production And Control
- Process Improvement
- Project Organization And Personnel
- Project Structures
- Effects Of Resource Availability On Quality
- Training Considerations
- Software Reporting Metrics
- Test Coverage and How to Measure and Report it

Module 7 – Planning, Organizing, Implementing and Managing SQA (1.5 hrs.)

- Tailoring The Function
- The Role of the SQA Plan
- Determining Methods And Tools
- Developing A SQA Corrective Action Procedure
- The Goal/Question/Metric Paradigm
- Identifying Needs and Goals
- Answering the Right Questions
- Getting the Right Data
- Critical Success Factors
- How to Perform Root Cause Analysis
- Resource Allocation
- A Basic Framework for Improvement
- Developing Your Own Action Plan for Improvement
- Cases, Exercises And Examples