

Code Analysis and Software Quality Training

Duration: 2 Days

Description

Code Analysis and Software Quality Training helps software teams build secure, sustainable, measurable and maintainable applications. This two-day online course takes an integrated approach to Clean Code principles, static and dynamic code analysis, code metrics, code reviews and secure development practices.

Participants learn to identify code smells and structural problems, interpret analysis results and incorporate quality checks into development processes. The course provides a practical framework for establishing shared coding standards, reducing technical debt and continuously monitoring software quality.

Who Should Attend?

- Software developers
- Software architects and technical leads
- Test engineers and quality assurance specialists
- DevOps and DevSecOps teams
- Managers responsible for software quality processes
- Professionals seeking to improve code reviews and quality standards

Learning Outcomes

- Apply Clean Code principles to software projects
- Distinguish static and dynamic code analysis methods
- Identify code smells and structural problems
- Interpret code metrics and analysis reports
- Establish effective code review processes
- Make technical debt visible and manageable
- Integrate quality checks into CI/CD processes
- Contribute to organizational coding standards

Outline

1. An Organizational Perspective on Software Quality

- The impact of software quality on projects and organizations
- Sustainable software development
- Identifying quality risks early in development
- Organizational coding standards and quality policies
- Relationships between software quality, cost and technical debt

2. Clean Code and Sustainable Coding

- Clean Code principles
- Writing readable and maintainable code
- Reducing duplication
- Meaningful naming and appropriate structure
- Identifying code smells
- Determining refactoring needs

3. Static Code Analysis

- What is static code analysis?
- Detecting errors and quality issues without running code
- Code smells, structural issues and potential vulnerabilities
- Defining analysis rules and thresholds
- Interpreting reports
- Integrating static analysis into development

4. Dynamic Code Analysis and Runtime Assessment

- Differences between static and dynamic analysis
- Examining application behavior at runtime
- Assessing performance, resource usage and memory issues
- Identifying runtime errors and risks
- Feeding findings into quality improvement

5. Code Metrics and Quality Standards

- Measuring code complexity and dependencies
- Indicators of readability and maintainability
- Testability and reusability
- Measuring and prioritizing technical debt
- Establishing organizational coding standards

- Defining measurable software quality objectives

6. Secure Code Development

- Secure development principles
- Code-level security
- Common security errors and risks
- Adding security controls to the software development lifecycle
- Managing quality and security checks together

7. Code Review and White-Box Testing

- Purpose and scope of code reviews
- Effective review methods
- Preparing review checklists
- White-box testing approaches
- Managing feedback and corrections
- Building a quality culture within software teams

8. Code Quality in CI/CD Processes

- Continuous integration and code quality
- Incorporating automated quality checks into CI/CD
- Quality gates
- Quality monitoring and reporting
- Establishing early warning mechanisms
- Improving organizational software quality maturity

Prerequisites

Basic programming knowledge and a general understanding of software development processes are recommended. Advanced experience with a particular programming language or code analysis tool is not required.

Customized Code Analysis and Software Quality Training

The program can be tailored to the organization's programming languages, development methods, quality standards, project needs and team experience. Examples and exercises for .NET or Java teams can be adapted to their technology environment. Related .NET and Java training develops technical skills, while secure software development courses provide a deeper security focus. DevOps training connects quality checks with automated development and deployment, and ISTQB testing courses develop testing and quality assurance skills. Complete the contact form for program, scheduling and proposal information.