Software quality management
From Wikipedia, the free encyclopedia

Software quality management is a management process that aims to develop and manage the quality of software to make sure the product satisfies the user. The goals of SQM (software quality management) is to make sure the product follows regulations and meets the quality standards expected by the customer. Software quality managers have to test the product before it is released to the market and they do this by a series of steps called the software cycle in order to reveal and fix bugs before release. Their job is not only to make sure that their software is in good shape for the consumer, but also to encourage the quality culture to everyone and avoid fraud by protecting their software through proper development.

Contents

1 Definitions
2 Description
3 SQM Roles
4 Software quality management and software lifecycle
5 Links to IT methods
6 Associations and organizations
7 See also
8 References

Definitions

The aim of Software Quality Management (SQM) is to manage the quality of software and of its development process. A quality product is one which meets its requirements and satisfies the user. A quality culture is an organizational environment where quality is viewed as everyone’s responsibility.

Description

The computer scientist Ian Sommerville[1] uses SQM as an umbrella-term that includes the following quality layers:

Software Quality Assurance (SQA) layer

An Organizational quality guide of

- Standards, regulations, and procedures to produce, verify, evaluate and confirm work products during the software development lifecycle
- Incorporated knowledge base of best practices
- Off-the-shelf software tools selected to apply the above

Software Quality Plan (SQP) layer

A project level quality plan written by each project for declaring project commitment to follow an applicable set of standards, regulations, procedures and tools during the development lifecycle. In addition, SQP should contain quality goals to be achieved, expected risks and risk management. SQP sources are derived from

- SQA components that are adopted as is or customized to the project's needs
- New procedures, standards and tools complementing missing or not-applicable SQA components that have been written in particular for the project, or imported from outside the organization.
Any deviation of an SQP from SQA should be justified by the project manager and be confirmed by the company management.

**Software Quality Control (SQC) layer**

Ensures in-process that both SQA and SQP are being followed by the development teams.

**SQC activities include**

- Mentoring how to produce artifacts, such as well-defined engineering documents using standard templates
- Mentoring how to conduct standard processes, such as quality reviews
- Perform in-process quality reviews to verify, evaluate and confirm artifacts
- Verify and evaluate to improve the use of methods, procedures and adopted software tools

**SQM Roles**

- to ensure that the required level of quality is achieved in a software product
- to encourage a company-wide "Quality Culture" where quality is viewed as everyone's responsibility
- to reduce the learning curve and help with continuity in case team members change positions within the organization
- to enable in-process fault avoidance and fault prevention through proper development

Many people use the terms SQM and SQA (Software quality assurance) interchangeably.

**Software quality management and software lifecycle**

Software quality management can be realized in various ways depending on organization and type of realized project, but it should support whole software development lifecycle, meaning:

- Collecting requirements and defining scope of IT project, focused on verification if defined requirements will be testable. One of the products can be test strategy.
- Designing the solution, focused on planning test process e.g. what type of tests will be performed, how they will be performed in context of test environments and test data. One of the products can be test plan including test schedule.
- Solution implementation supported by creating test cases and scenarios, executing them and registering defects including coordination of fixing them. Products can be test cases and scenarios, reports from test iteration realization.
- Change management, supported by verification how planned changes can influence the quality of created solution and eventual change of test plan. One of the products can be changes in test plan, test cases and scenarios.
- Closing project, supported by realization number of tests focused on complex verification of overall quality of created solution. It can include System Integration Tests, User Acceptance Tests and Operational Acceptance Tests. One of the products can be recommendation about production start of the system.

**Links to IT methods**

Software quality management is a topic strongly linked with various project management, development and IT operation methods like:

- Project management method PRINCE2 defines:
  - component "Quality in a project environment", which describes necessity of double-checked and objective control of created products. It proposes using 4 elements: quality management system, function of quality control, planning quality and quality controls.
"Quality Review Technique" which is focused on verification if created products fulfills defined quality criteria.

  - 3.4.12 Plan Quality,
  - 3.5.2. Perform Quality Assurance,
  - 3.6.7. Perform Quality Control

- Development method RUP defines discipline testing, which is engaged in all phases starting from Inception, finishing at Transition.
- Development method MSF defines tester role and stabilization phase, which focuses mainly on testing a solution.[6]
- Agile methods do not precisely define the tester's role or mechanisms related to software quality management. The methods define only such techniques as continuous integration and test-driven development. Nevertheless, there appears lastly the publication about agile testing.
- Operational method CMMI defines among others process area PPQA "Process and Product Quality Assurance", which is required already at CMMI level 2.
- Operational method COBIT defines among others process P08 Manage Quality.
- Operational method ITIL is defined among others by publication Continual service improvement.
- V-Model – model, which defines the software development lifecycle and test process.
- ISO 9000 – family of standards is related to quality management systems and designed to help organizations ensure that they meet the needs of customers and other stakeholders[1] while meeting statutory and regulatory requirements related to the product.

**Associations and organizations**

- ISTQB, International Software Testing Qualifications Board is non-profit association registered in Belgium and working world-wide. It manages certification process for software testers. There is already over 200,000 ISTQB® certificates (date: March 2012).[7]
- ASQ, American Society for Quality is a knowledge-based global community of quality professionals, with nearly 80,000 members dedicated to the promotion and advancement of quality tools, principles, and practices in their workplaces and in their communities.

**See also**

- Agile testing
- Software assurance
- Quality assurance
- Software quality
- Software quality control
- Software quality assurance
- Software quality analyst

**References**

This article is based on material taken from the Free On-line Dictionary of Computing prior to 1 November 2008 and incorporated under the "relicensing" terms of the GFDL, version 1.3 or later.

1. Ian Sommerville (2004), Software Engineering, 7th ed., chapter 27
7. ISTQB (http://www.istqb.org/)


- This page was last edited on 30 June 2017, at 23:29.
- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.