

Overview

- This chapter provides an introduction for software quality assurance (SQA). SQA is the concern of every software engineer to reduce cost and improve product time-to-market. A Software Quality Assurance Plan is not merely another name for a test plan, though test plans are included in an SQA plan. SQA activities are performed on every software project. Use of metrics is an important part of developing a strategy to improve the quality of both software processes and work products.

Quality Concepts

- Variation control is the heart of quality control (software engineers strive to control the process applied, resources expended, and end product quality attributes).
- Quality of design - refers to characteristics designers specify for the end product to be constructed
- Quality of conformance - degree to which design specifications are followed in manufacturing the product
- Quality control - series of inspections, reviews, and tests used to ensure conformance of a work product to its specifications
- Quality assurance - consists of the auditing and reporting procedures used to provide management with data needed to make proactive decisions

Cost of Quality

- Prevention costs - quality planning, formal technical reviews, test equipment, training
- Appraisal costs - in-process and inter-process inspection, equipment calibration and maintenance, testing
- Failure costs - rework, repair, failure mode analysis
- External failure costs - complaint resolution, product return and replacement, help line support, warranty work

Total Quality Management

- Kaizen - develop a process that is visible, repeatable, and measurable
- Atarimae hinshitsu - examine the intangibles that affect the process and work to optimize their impact on the process
- Kansei - examine the way the product is used by the customer with an eye to improving both the product and the development process
- Miryokuteki hinshitsu - observe product use in the market place to uncover new product applications and identify new products to develop

Software Quality Assurance

- Conformance to software requirements is the foundation from which software quality is measured.
- Specified standards are used to define the development criteria that are used to guide the manner in which software is engineered.
- Software must conform to implicit requirements (ease of use, maintainability, reliability, etc.) as well as its explicit requirements.

SQA Group Activities

- Prepare SQA plan for the project.
- Participate in the development of the project's software process description.
- Review software engineering activities to verify compliance with the defined software process.

- Audit designated software work products to verify compliance with those defined as part of the software process.
- Ensure that any deviations in software or work products are documented and handled according to a documented procedure.
- Record any evidence of noncompliance and reports them to management.

Software Reviews

- Purpose is to find defects (errors) before they are passed on to another software engineering activity or released to the customer.
- Software engineers (and others) conduct formal technical reviews (FTR) for software engineers.
- Using formal technical reviews (walkthroughs or inspections) is an effective means for improving software quality.

Formal Technical Reviews

- Involves 3 to 5 people (including reviewers)
- Advance preparation (no more than 2 hours per person) required
- Duration of review meeting should be less than 2 hours
- Focus of review is on a discrete work product
- Review leader organizes the review meeting at the producer's request
- Reviewers ask questions that enable the producer to discover his or her own error (the product is under review not the producer)
- Producer of the work product walks the reviewers through the product
- Recorder writes down any significant issues raised during the review
- Reviewers decide to accept or reject the work product and whether to require additional reviews of product or not

Statistical Quality Assurance

- Information about software defects is collected and categorized
- Each defect is traced back to its cause
- Using the Pareto principle (80% of the defects can be traced to 20% of the causes) isolate the "vital few" defect causes
- Move to correct the problems that caused the defects

Software Reliability

- Defined as the probability of failure free operation of a computer program in a specified environment for a specified time period
- Can be measured directly and estimated using historical and developmental data (unlike many other software quality factors)
- Software reliability problems can usually be traced back to errors in design or implementation.

Software Safety

- Defined as a software quality assurance activity that focuses on identifying potential hazards that may cause a software system to fail.
- Early identification of software hazards allows developers to specify design features to eliminate or at least control the impact of potential hazards.
- Software reliability involves determining the likelihood that a failure will occur, while software safety examines the ways in which failures may result in conditions that can lead to a mishap.

ISO Quality Standards

- Quality assurance systems are defined as the organizational structure, responsibilities, procedures, processes, and resources for implementing quality management.
- ISO 9000 describes the quality elements that must be present for a quality assurance system to be compliant with the standard, but it does not describe how an organization should implement these elements.
- ISO 9001 is the quality standard that contains 20 requirements that must be present in an effective software quality assurance system.

SQA Plan

- Management section - describes the place of SQA in the structure of the organization
- Documentation section - describes each work product produced as part of the software process
- Standards, practices, and conventions section - lists all applicable standards/practices applied during the software process and any metrics to be collected as part of the software engineering work
- Reviews and audits section - provides an overview of the approach used in the reviews and audits to be conducted during the project
- Test section - references the test plan and procedure document and defines test record keeping requirements
- Problem reporting and corrective action section - defines procedures for reporting, tracking, and resolving errors or defects, identifies organizational responsibilities for these activities
- Other - tools, SQA methods, change control, record keeping, training, and risk management

REFER TO CHAPTER 8 OF ***“Pressman. R. S., Software Engineering a practitioners Approach. 5th Edition”*** ACCORDINGLY.