IF2261 Software Engineering

Quality Management

Program Studi Teknik Informatika
STEI ITB

Overview

- Software quality assurance (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.

**** SEPA 6th ed, Roger S. Pressman***

Quality Concepts

- Variation control is the heart of quality control
  - Form one project to another, we want to minimize the difference between the predicted resources needed to complete a project and the actual resources used, including staff, equipment, and calendar time
- Quality of design
  - refers to characteristics that 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 a set of auditing and reporting functions that assess the effectiveness and completeness of quality control activities

**** SEPA 6th ed, Roger S. Pressman***

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

**** SEPA 6th ed, Roger S. Pressman***

Software Quality Assurance

- Definition of Software Quality serves to emphasize:
  - 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.

** SEPA 6th ed, Roger S. Pressman*

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.

** SEPA 6th ed, Roger S. Pressman*

Software Reviews

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

**** SEPA 6th ed, Roger S. Pressman***

Software Defects

- Industry studies suggest that design activities introduce 50-65% of all defects or errors during the software process
- Review techniques have been shown to be up to 75% effective in uncovering design flaws which ultimately reduces the cost of subsequent activities in the software process

**** SEPA 6th ed, Roger S. Pressman***

Sample Driven Reviews

- Samples of all software engineering work products are reviewed to determine the most error-prone
- Full FTR resources are focused on the likely error-prone work products based on sampling results
- To be effective the sample driven review process must be driven by quantitative measures of the work products (see SEPA page 758)

** SEPA 6th ed, Roger S. Pressman*

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 in the "vital few"

** SEPA 6th ed, Roger S. Pressman*

Six Sigma for Software Engineering

- The most widely used strategy for statistical quality assurance
- Three core steps:
  - Define customer requirements, deliverables, and project goals via well-defined methods of customer communication.
  - Measure each existing process and its output to determine current quality performance (e.g., compute defect metrics)
  - Analyze defect metrics and determine vital few causes.
- For an existing process that needs improvement
  - Improve process by eliminating the root causes for defects
  - Control future work to ensure that future work does not reintroduce causes of defects
- If new processes are being developed
  - Design each new process to avoid root causes of defects and to meet customer requirements
  - Verify that the process model will avoid defects and meet customer requirements

**** SEPA 6th ed, Roger S. Pressman***

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
- Software reliability problems can usually be traced back to errors in design or implementation.
- Measures of Reliability
  - Mean time between failure (MTBF) = MTTF + MTTR
  - MTTF = mean time to failure
  - MTTR = mean time to repair
  - Availability = [MTTF / (MTTF + MTTR)] x 100%

**** SEPA 6th ed, Roger S. Pressman***

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 can 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.

**** SEPA 6th ed, Roger S. Pressman***

ISO 9000 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 9001:2000 is the quality standard that contains 20 requirements that must be present in an effective software quality assurance system.

**** SEPA 6th ed, Roger S. Pressman***

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

**** SEPA 6th ed, Roger S. Pressman***