Quality Management

Dr. Raman Chadda[1]
Arzoo Gupta[2]
Bhupinder Singh[3]
Yatin Chadha[4]

Abstract

With the advancements in the technology, software has become a crucial part. The future cannot be even imagined without software's. Software engineering has evolved over years promoting a good development cycle of software products. But somewhere somehow quality is still compromised. Though it is a systematic and disciplined approach towards the development, not much focus is laid on the quality of the developed software. This paper will provide a brief knowledge of the proposed ideas for implementing concepts to enhance the quality of the software product some of which will work in interaction with software development life cycle models.

KeyWords: *Quality Engineering, Software Engineering, Software Development Life Cycle, Six Sigma.*

1 Prof. & Head Deptt. of C.S.E, C.G.C Technical Campus, Jhanjeri, Mohali
2 Deptt. of C.S.E, C.G.C Technical Campus, Jhanjeri, Mohali
3 Deptt. of C.S.E, C.G.C Technical Campus, Jhanjeri, Mohali
4 Deptt. of C.S.E, C.G.C Technical Campus, Jhanjeri, Mohali
5

Introduction

There are many companies who work for the development of various kinds of software and promise the customer a good quality product. But, are they able to manage good quality? Before that, let's see what actually quality management is.

Software quality management -

- Concerned with ensuring the required level of quality is achieved in a software product

- Involves the definition of appropriate quality standards and the definition of procedures to ensure that these standards are followed

- Works best when a 'quality culture' is created where quality if seen as everyone's responsibility

- Quality means that a product satisfies the demands of its specifications

- It also means achieving a high level of customer satisfaction with the product

- In software systems this is difficult -

Customer quality requirements (e.g. efficiency or reliability) often conflict with developer quality requirements (e.g. maintainability or reusability). Software specifications are often incomplete, inconsistent, or ambiguous.

Quality management ensures that an organization, product or service is consistent. It has four main components –

- Quality improvement

- Quality assurance - establishing organizational quality standards and procedures

- Quality planning - selecting and modifying applicable quality standards and procedures for a particular project

- Quality control - ensuring quality standards and procedures are followed by development team

Note: Quality management should be separated from project management to ensure independence.

Quality management, therefore, uses quality assurance and control of processes as well as products to achieve more consistent quality.

Software quality measurement quantifies to what extent a software or system rates. With the speed at which software engineering is evolving, quality management also needs to keep pace. In recent years many areas and new applications have emerged such as defect detection techniques, six sigma concepts etc. Though quality management works for the quality of the developed product but it is a different aspect and not specific to software engineering. [1]

ISO 9000

- International set of standards for quality management

- Quality standards and procedures must be documented in an organizational quality manual

- An external body is often used to certify that the quality manual conforms to ISO 9000 standards

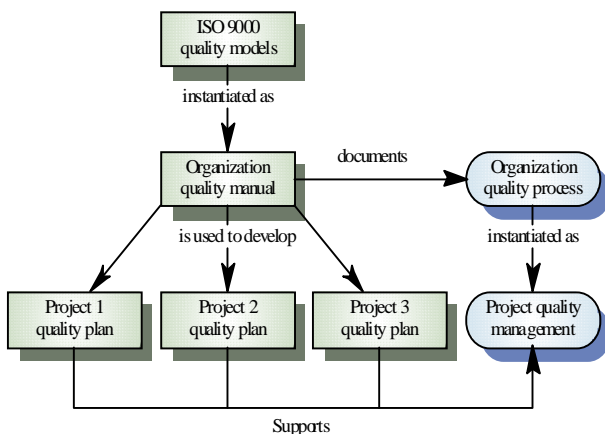- Many customers are demanding that suppliers are ISO 9000 certified[2]

```
              ISO 9000
             quality models

              instantiated as

   Organization         documents        Organization
   quality manual     ──────────────►    quality process

   is used to develop                    instantiated as

Project 1      Project 2      Project 3    Project quality
quality plan   quality plan   quality plan   management

                    Supports
```

Figure 1.1 – ISO Standards

2. PROPOSED IDEAS

There were many concepts which were proposed for enhancing and managing the quality and some of which are –

Six Sigma Concepts

- Is the most widely used strategy for statistical quality assurance

- Uses data and statistical analysis to measure and improve a company's operational performance

- Identifies and eliminates defects in manufacturing and service-related processes

- The "Six Sigma" refers to six standard deviations (3.4 defects per a million occurrences)

- Three core steps

- Define customer requirements, deliverables, and project goals via well-defined methods of customer communication

- Measure the existing process and its output to determine current quality performance (collect defect metrics)

- Analyze defect metrics and determine the vital few causes (the 20%)

- Two additional steps are added for existing processes (and can be done in parallel)

  - Improve the process by eliminating the root causes of defects

  - Control the process to ensure that future work does not reintroduce the causes of defects

- All of these steps need to be performed so that you can manage the process to accomplish something

- You cannot effectively manage and improve a process until you first do these steps (in this order): [3]
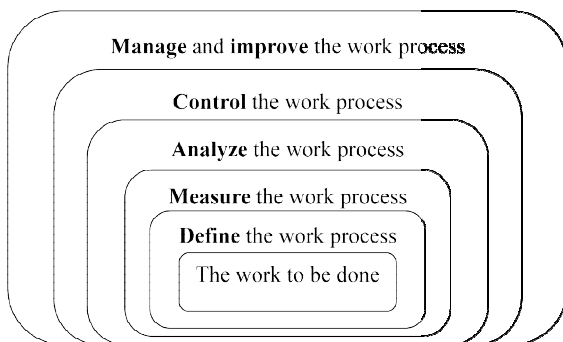

Figure 2.1 – Six Sigma Concept

Quality Engineering

According to the work that has been proposed, quality engineering can be defined as a field which works in phases to enhance the quality of the developed product. It works with the software development life cycle models. It provides some rules and guidelines using which the companies can have their own processes or procedures for enhancements according to their standards.
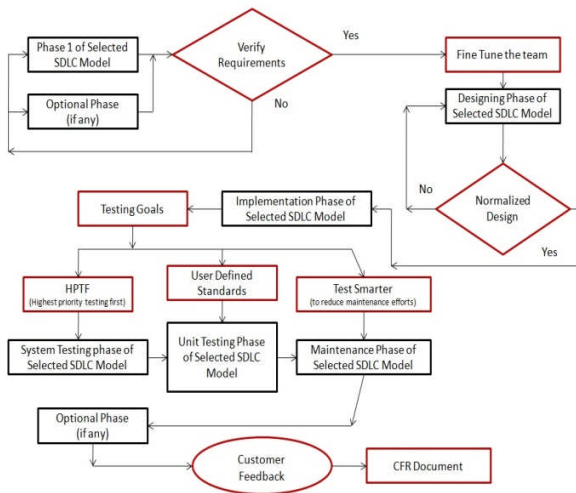
Figure 2.2 – Basic Model of Quality Engineering

The phases in red color are the phases of the basic model of quality engineering and every phase of SDLC (software development life cycle) is accompanied by a phase of quality engineering model. So, after the development of every phase its quality is enhanced separating the work in modules therefore providing modularity and independence. This basic model can embed any of the SDLC model for instance lets take an example of waterfall model.
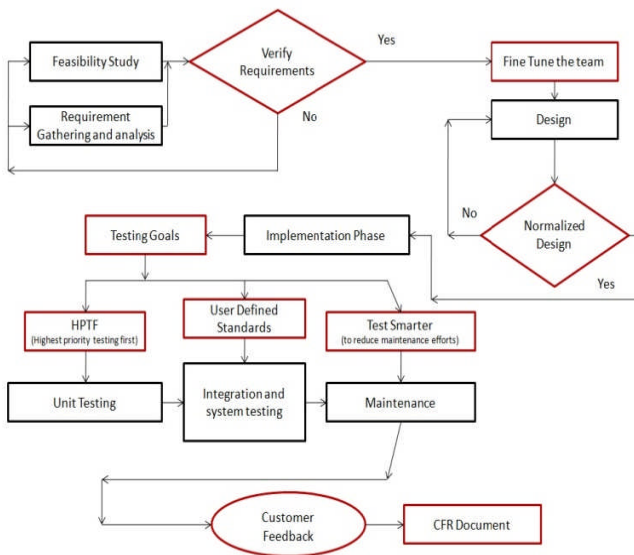
Figure 2.3 – Basic Model of Quality Engineering embedding waterfall model

In the figure above, a SDLC model i.e. the waterfall model is embedded. After gathering the requirements firstly they are verified and then only it continues with the next phase. Same in the case of the design, till the design is not normalized its implementation cannot take place. Various standards are also provided before the testing phases to reduce the efforts by performing the appropriate testing in a systematic manner.

There are some rules and guidelines which needs to be followed -

- Every phase of SDLC model should be followed by a phase of quality engineering model.
- Verify the requirements before proceeding.
- According to the project choose the appropriate team members who can perform various tasks in one go.
- Designing phase should be validated for a simpler cleaner design which is easier to test and rework.
- Before proceeding to the testing phases, testing goals should be defined to ensure interoperatibility and cooperation.
- Optimize the testing cycle such that the highest priority test are performed first such that the lowest priority test can be out sourced or ignored.

- Focus on maintaining the most crucial modules such that these modules receive the highest share of the test resources.
- Identify and eliminate causes of faults and errors.

According to the basic model and the rules provided a company can design its own model of quality engineering according to their needs and requirements.[5]

CONCLUSION

Quality engineering deals with the enhancements of the software product by providing a basic model and various rules and guidelines so that the company could work according to their needs and requirements. Also, six sigma concepts promotes the quality by evaluating the needs of the customer, measuring the existing system and by analyzing the defect metrics. Quality management needs to maintain its evolution with that of software engineering so as to provide greater quality with satisfaction.

REFERENCE:

[1] Software Engineering: A practioner's approach by Roger S. Pressmen.
[2]www.iso.org/
[3]Software Measurement: Elbert, C. and R. Dumke
[4]Software Engineering by Rajib Mall
[5]www.wikipedia.org
[6]www.google.com