Introduction to Software Quality

by
Ramraj S ME.,
Assistant Professor
Department of Software Engineering

5 -Jan -2016

Agenda

- Introduction
- Views of Software Quality
- Measurement of Software Quality
- Conclusion

Introduction

- What is Software Quality
- What are the different views in Software Quality
- Measurement related to Software Quality

Views of Software Quality

"Quality is never an accident; it is always the result of intelligent effort. John Ruskin"

- Transcendental View
- User View
- Manfacturing View
- Product View
- Value Based View

Transcendental View

- It is said to be philosphical
- No Measurement
- Only through experience

User View

- User is concerned with whether or not a product is fit for use.
- A product is considered to be of good quality if it satisfies the needs of a large number of customers.

Apart from doing basic functionalities, the following features are also measured:

1. Usability
2. Testability
3. Reliability
4. Efficiency

Manfacturing View

- Quality is related to conformance to requirements
- Hom much the product is satisfying the requirements is taken as measurement.
- Process level improvement plays a vital role in this view
- CMM and ISO are developed based on Manfacturing view

Product View

- Quality is related to the some property of the product
- If a product is manufactured with good internal properties, then it will have good external qualities
- If a code is having good level of modularity then it is easy for testable and maintainable.

Value Based View

- It relates two terms excellence and worth
- How much a customer is willing to pay for a certain level of quality
- It is a tradeoff between cost and quality.

Measuring Quality

Need to measure quality

- Used to fix a baseline
- Is thera a need for process improvements?
- How much to spend on process improvement

Measurement of User's View

Key terms to Measure:

1. Functionality
2. Reliability
3. Usability

- How many Functionality of a product delivers ? Is a metric to measure.
- It is measured through the following mannner:
  Number of test cases passed by the functionalities / Total number of test cases designed to verify the functionalities

References I

[1] K.Naik, *"Software Testing and Quality Assurance Theory and Practice"*, Chapter 17

Thank you