



CLASSIFICATION AND CALCULATION OF DIFFERENT TYPES OF COSTS OF SOFTWARE QUALITY TESTING

SIBGATULLINA, T[aisiya]

Abstract: *It is difficult to imagine implementation of application or a new release without any testing process. There are many approaches of measure quality of processes in organization, but still there is very little knowledge is available about the economics of software quality. The costs incurred and benefits of implementing different quality practices over the software development life cycle are not well understood. There are many definitions of quality being given by experts that explains quality for manufacturing industry but still unable to define it with absolute clarity for software engineering. In the article there are some useful practices how to calculate return of investments of implementation automatization of testing processes and wch impact they have on cost, quality and time.*

Key words: *capability maturity model integration, return of investments, cost of quality, software testing*

1. INTRODUCTION

There are three main measurements for quality of project: cost, scope and time. How company can govern those parameters in order to achieve better results? How quality could be measured? In 1960-x appears concept of Quality costs. It is the costs associated with preventing, finding, and correcting defective work. Cost of Quality for company is a very difficult topic, each company has own process for achieving quality. We can divide all companies based on maturity levels which were proposed by CMMI (Capability Maturity Model Integration) methodology. This methodology gets some practices and statistic information for each maturity level of process, but there are now clear formulas how to measure Quality of Process.

Probably the reason is that there are no single standards of Software industry. Definition exists for what constitutes good quality in software; it is generally taken to mean that a software product provides value (satisfaction) to its users, makes a profit, and generates few serious complaints.

2. LEVEL OF QUALITY

There are a lot of deferent standards and glossaries which try to define lexicon for software quality: ISO 9000, IEEE, CMMI, and GOST (for Russia)

The following are the key dimensions of software quality.

- Level of satisfaction: The degree to which customers or users perceive that a software product meets their composite needs and expectations.
- Product value: The degree to which a software product has value for its various stakeholders relative to the competitive environment.
- Key attributes: The degree to which a software product possesses a combination of desired properties, e.g., reliability, portability, maintainability.
- Defectiveness: The degree to which a software product works incorrectly in target user environments due to debilitating operational defects.

- Process quality: In relation to the development process by which the product is produced, it means.

But it is not real numbers, how to calculate satisfaction or process of quality. In 1962 J. Juran introduce new terming - Cost of quality (CoQ).

3. HOW TO CALCULATE QUALITY

Cost of software accounting (CoSQ) is useful to enable our understanding of the economic trade-offs involved in delivering good-quality software. It is used in manufacturing area, but now it is used to software as for ordinary product. Different authors and researcher have used different ways to classify components for quality cost; if we look carefully their understanding about various components is approximately the same. As explained by Rex Black that: "Investing in Software Testing, decrease The Cost of Software Quality". He has mentioned costs of quality into two major types: conformance and nonconformance

$$CoSQ = Cost\ Conformance + Cost\ non-conformance \quad (1)$$

Another approach calculates CoQ as fore useful definitions of type of Costs, as applied to software products.

- Prevention Costs: Costs of activities that are specifically designed to prevent poor quality: coding errors, design errors, mistakes in the user manuals. This money is usually spent by the programming, design, and marketing staffs.
- Appraisal Costs: Costs of activities designed to find quality problems, such as code inspections and a type of testing.
- Failure Costs: Costs that result from poor quality, such as the cost of fixing bugs and the cost of dealing with customer complaints.
- Internal Failure Costs: Failure costs that arise before your company supplies its product to the customer. Along with costs of finding and fixing bugs are many internal failure costs borne by groups outside of Product Development. If a bug blocks someone in your company from doing job, the costs of the wasted time, the missed milestones, and the overtime to get back onto schedule are all internal failure costs.
- External failure Costs: Costs which are generated by defective products, services, and processes during customer use. They include warranties, complaints, replacements or recalls, repairs and customer returns.

Total Cost of Software Quality (CoSQ) could be calculated as

$$CoSQ = Prevention\ Cost + Appraisal\ Cost + Internal\ failure\ Cost + External\ failure\ Cost \quad (2)$$

The key consideration in any analysis of the cost of quality is visibility. To make decision to set up or not to set up testing process, and how many resources should be spend on testing in the key point of each project.

When you invest many you usually expect that you will get some values from these investments, the same is for quality process. General formula for calculating return of investment (ROI) is common for all area:

$$ROI = \frac{(Gain\ from\ Investment - Cost\ of\ Investment)}{Cost\ of\ Investment} \quad (3)$$

In the article “The Software Quality Economics Model for Software Project Optimization” authors demonstrate example how to calculate CoSQ based on CMMI Levels. Defect removal effectiveness in testing phase for level 1 – 85%, for level 2 – 89%, for level 3 – 91%, for level 4 – 93%, for level 5 -95%. Based on this information and deferent models of calculation ROI we can get information about effectiveness of investments in testing.

The most common ROI model, and that has been used more often than not in software engineering, is shown below:

$$ROI(1) = \frac{Total\ CoQ - Test\ Investment}{Test\ Investment} * 100\% \quad (4)$$

Testing Investment Options: ROI Analysis			
Testing resources	Case 1 CMM <1 Level	Case 2 CMM 1 Level	Case 3 CMM >3 Level
Staff	\$0	\$60,000	\$60,000
Infrastructure	\$0	\$10,000	\$10,000
Tools	\$0	\$0	\$12,500
Total Test Investment	\$0	\$70,000	\$82,500
Development (Requirement, Design, Code)			
Must-Fix Bugs Found	250	250	350
Fix Cost - \$10 per bug (Internal Failure)	\$2,500	\$2,500	\$3,500
Testing			
Must-Fix Bugs Found	0	600	600
Fix Cost - \$100 per bug (Internal Failure)	\$0	\$60,000	\$60,000
Customer Support			
Must-Fix Bugs Reported	750	150	50
Fix Cost - \$1000 per bug (External Failure)	\$750,000	\$150,000	\$50,000
Cost of Quality (CoQ)			
Conformance	\$0	\$70,000	\$82,500
Nonconformance	\$752,500	\$212,500	\$113,500
Total CoQ	\$752,500	\$282,500	\$196,000
Return on Investment (ROI₁)	#N/A	571%	575%
Return on Investment (ROI₂)	#N/A	62%	74%

Authors of the article calculation on hypothetical case study to illustrate the use of this cost of quality technique to analyze return on the testing investment.

Example: software product, with one new release every quarter. On average, each release contains 1,000 “must-fix” bugs— which we identify and repair over the life of the release. Suppose that bugs found by programmers costs \$10 to fix. Bugs found by customers cost \$1,000 to fix. Let’s look to three deferent cases:

Case 1: Low Quality Results CMMI Level <1. Quality of process less than CMMI model, according to CMMI Statistic in these just 250 bugs, it means that that 750 bugs escape to the field.

Case 2: Good Quality Results CMMI Level 1. The project management decided to improve software testing process (STP) and invested in testing staff \$60,000 and test infrastructure \$10,000 Defect removal operations consist of six test stages: 1) unit test, 2) new function test, 3) regression test, 4) integration test, 5) system test, and 6) external Beta test.

Case 3: High Quality Results CMMI level 3. Assume that company on CMMI level 3 on the CMM scale. By means of more effective defect prevention such as Quality Function Deployment (QFD) and Six-Sigma the defect potentials are lower. Defect removal efficiency is assumed to be 95%. Defect removal operations consist of nine stages: 1) design inspections; 2) code inspections; 3) unit test, 4) new function

test, 5) regression test, 6) integration test, 7) performance test, 8) system test, 9) external Beta test.

To clarify the differences between the three case studies, note that both examples are exactly the same size, but differ in these key elements: CMM levels; Defect prevention; Defect potentials; Defect removal efficiency; Development costs;

ROI (1) shows us that we save money, but what has happened with quality level:

$$ROI(2) = \frac{Total\ CoQ(1) - Total\ CoQ(2)}{Total\ CoQ(1)} * 100\% \quad (5)$$

ROI (1) and ROI (2) shows impact of costs on quality process, but as we mentioned at the beginning of the article we have time as one of key measurement of project. It is clear that we have to pay attention on effectiveness of testing activities. Effectiveness is consists of: quality level, total time of testing and investments in test process. If we have quality level as constant, let’s look to our example once more.

We have deferent numbers of testing types for case 2 and case 3. For case 2 we have 6 test stages and for case 3 we have 9 test stages. It means that test period would be longer and complexity of test process would increase and we have to spend more time and resources to track more activities, but a difference in ROI (2) is just 12%. To make informed decision about invest or not to invest many to test process and make it more complex we should based on not just calculating ROI(1) and ROI(2), but calculate full time of testing process.

$$Time\ ROI = \frac{Time(1) - (Time(2) + Time\ of\ Process)}{Time(1)} * 100\% \quad (5)$$

Time of process is costs of setting up and supporting new process. Return of investments in testing would be less that we have in our example, it means that each time when we want to invest money to make additional type of testing we made process more complex

4. CONCLUSION

There are deferent ways of ROI calculation in testing process, the most popular is described in this article. But they did not cover time investment. In this article we propose calculate time investment as one of base parameter before invest and change current process.

Next step is to make additional research and implement those criteria in decision process of necessity of test automatization.

5. REFERENCES

Black, R. (February 2007). *Pragmatic Software Testing: Becoming an Effective and Efficient Test Professional*. Wiley. ISBN 0470127902, New York

Boehm, B.W.; Brown, J.R.; Kaspar, H.; Lipow, M.; MacLeod, G.J.; Merritt, M.J. (1978) *Characteristics of Software Quality*, North Holland Publishing Company, ISBN-10: 0444851054, Holland, Amsterdam

Daughtrey, T. (2001). *Fundamental Concepts for the Software Quality Engineer*, ASQ Quality Press, ISBN-08-7389-521-5, Milwaukee

Juran, J.M. (1951) *Quality Control Handbook*, New York: McGraw-Hill, OCLC 1220529, New York

Lizic, L.; Kolasinac, A.; Dzenan, A. (2009) *The Software Quality Economics Model for Software Project Optimization*, University of Novi Pozar, ISSN: 1109-2750, Pozar, Serbia