ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 4, Issue 4 , April 2017

Role of Cost as a Quality Character in the Software Metrics of Project Development

John Vijay T, Rao T V, Gopi Chand M , Harika D

Research Scholar, JNTU, Hyderabad
Professor& Head, CSE, PVPSIT, Vijayawada
Professor & Head IT, VCE, Hyderabad
Asst. Professor, SVEC, Tadepalligudem

ABSTRACT: The practice of applying software metrics to operational factors and to maintain factors is a complex task. Successful software quality assurance is highly dependent on software metrics. It needs a linkage between the software quality model and software metrics through quality factors in order to offer measure method for software quality assurance. The contributions of this paper to focus on Software quality metrics based on external factors one of which is project Cost, which has influence over the quality of the final product. In this study, an attempt has been made to show how the cost is playing a vital role in the quality and the development of the software Project.

KEYWORDS: Software Metrics, Software External Character, Project Cost.

I. INTRODUCTION

Measurement not only helps to visualize the abstraction of software development process and product but also provide an infrastructure to perform comparison, assessment and prediction of software development artefacts. Software metrics, as a tool to measure software development progress, has become an integral part of software development as well as software research activities. Software metrics are categorized into three main groups viz., Product metrics, Process metrics and Resource metrics. Product metrics involve measurement of different feature of documents and programs generated during the software development process. Process metrics relate to measurement of activity happened during software development life cycle, such as software design, implementation, test, and maintenance. Resource metrics measure supporting resources such as programmers, and cost of the product and processes, etc. [5].

Project planners need to consider risk situations when developing cost and schedule estimates. “quality of a project” Specifically related to software project, has many nuances and overtones. For example:

1. *Elegance or beauty* in the eye of the beholder
2. *Fitness of use* for various purposes
3. *Satisfaction of user requirements*, both explicit and implicit
4. *Freedom from defects*, perhaps to Six Sigma levels
5. *High efficiency of defect removal* activities
6. *High reliability* when operating
7. *Ease of learning* and *ease of use*
8. *Clarity of user guides* and HELP materials
9. *Ease of access* to customer support
10. *Rapid repairs* of reported defects [22].

Research on software systems has been roughly split into two aspects – process and product. Product research is mainly focused on the “what is produced” – generally technology focused, and addresses problems mostly within the ambit of computer science. These may include inventing technologies to improve a software system’s quality attributes, such as performance and security, or identifying better approaches to implementing existing or new technologies. For example, Web Services was invented in response to the growing interoperability needs among software systems. Whereas Process research instead addresses “how the product is being produced”–in particular, the managerial and administrative aspects of developing software systems. The focus is on the practices that guide the development of a software system from its conceptualization through its evolution to a possibly much larger system such as the kind that serves an enterprise. Examples of research within the process area may include methods for decision making, life cycle choice, cost and schedule estimation, design and analysis, and risk management [1]. In this regard, the objective of the present article is to focus on the influence of project cost over the development and the Quality of the project by considering the existing models and on-going projects.

II. LITERATURE REVIEW

A. Cost Estimating Models:

Budgeting, project planning and control, trade-off and risk analyses, software return on investment analyses are among the many uses of software engineering cost, schedule and quality models and estimation. Significant research on software cost modeling began with the extensive 1965 SDC study of the 104 attributes of 169 software projects [14]. This led to the development of some useful partial models in the late 1960's and early 1970's. In the late 1970's more robust models such as SLIM [16], Checkpoint [11], PRICE-S [15], SEER [10], and COCOMO [2] were developed. Although most of these models were developed at about the same time, they all faced the same dilemma: as software grew in size and importance it also grew in complexity, making it very difficult to accurately predict the cost, schedule and/or the quality of software product. This dynamic field of software estimation sustained the interests of these researchers who succeeded in setting the stepping stones of software engineering cost models. [20].

B. Cost Estimation Methodologies/Techniques:

Most estimating methodologies are predicated on analogous software programs. Expert opinion is based on experience from similar programs;

- *Analogies:* Cost and schedule are determined based on data from completed similar efforts.
- *Expert (engineering) opinion*: Cost and schedule are estimated by determining required effort basedon input from personnel with extensive experience on similar programs.
- *Parametric models:* The most commonly-used technology for software estimation is parametric models, a variety of which are available from both commercial and government sources. The estimates produced by the models are *repeatable*, facilitating sensitivity and domain analysis. The models generate estimates through statistical formulas that relate a dependent variable (e.g., cost, schedule, resources) to one or more independent variables.
- *Cost Performance Report (CPR) analysis:* Future cost and schedule estimates are based on current progress. This method may not be an optimal choice for predicting software cost and schedule because software is generally developed in three distinct phases (requirements/ design, code/unit test, integration/test) by different teams.

C. Factors influencing the Project cost:

To avoid the miseries of a runaway program, one must carefully plan for and control the cost and schedule of the software development which measures are important for determining and justifying the required funding, while size is by far the most significant driver of cost and schedule, other factors impact them as well. These factors are usually more qualitative in nature and address the development and operational environments as well as the software's characteristics. Most software cost estimating models use these factors to determine environmental and complexity factors which are in turn used in computations to calculate effort and cost [19].

The traditional Cost of Quality is comprised of:

- *The cost of defect prevention* - This can include process improvement, training, root cause analysis, or any other activities to prevent defects.
- *The cost of defect detection* - This can include inspections, testing, or any quality control (QC) activities designed to find problems before customers find them.
- *The cost of failures* - This includes both internal and external failure costs. Internal failures are the costs associated with the product before the customer receives it. External failures are the costs associated with the product after the customer receives it. In the manufacturing context, this also includes warranty costs [21].

Table 1. Various Costs involved in Project Designing

PREVENTION COSTS	DETECTION COSTS	INTERNAL FAILURE COSTS	EXTERNAL FAILURE COSTS
Root cause analysis	All stages of Testing	Rework	Rework
Training Personnel	Reviews and Inspections	Resources support due to delay	Penalties due to delay
	Improvement Costs		Patches and Post-release fixes

Using the below mentioned metrics related to cost the products test state and indicative level quality can be measured, which will be useful for product release decision [8].

- Test cost normalized to product size (TCS) = total cost of testing the product in dollars / source lines of code in thousands
- Test cost as a ration of development cost (TCD) = total cost of testing the product in dollars / total cost of developing the product in dollars
- Cost per defect unit = Total cost of a specific test phase in dollars / number of defects found in the Product after release

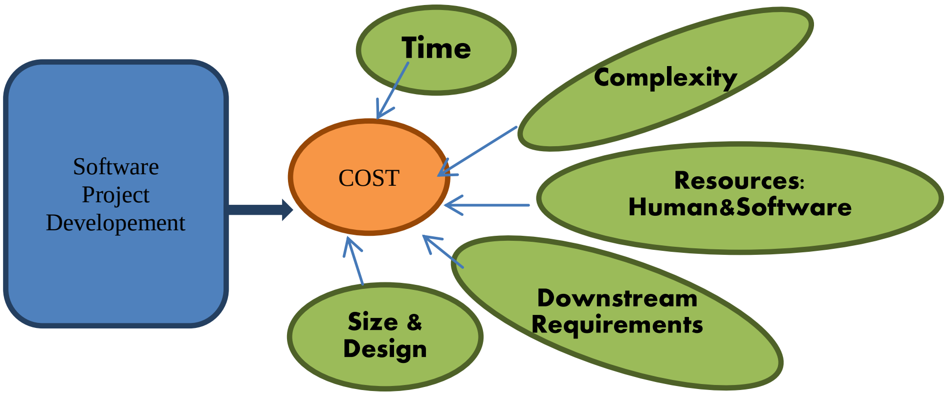

Fig 1. Factors influencing the Cost of project in Software development

From the buyer related approach a relationship between cost and quality is a quality product is designed for a specific use at an acceptable price [23], which is influenced by many factors.

The following summarizes the resources you should consider when costing software development.
• *Human resources*: This includes the number and qualifications of the people required, as well as their functional specialties. Boehm asserts that human resources are the most significant cost drivers on a software development effort. [2] Development personnel skills and experience (reflected in their productivity) have the greatest effect on cost and schedule.
• *Hardware resources*: This includes development (host) and target computers, and compilers. Hardware resources used to be major cost drivers when development personnel needed to share equipment with multiple constituencies. Now that virtually everyone has a PC or workstation on his or her desk, the issue is whether the target computer significantly differs from the development computer. For instance, if the target machine is an air or space borne system, the actual CPU may be technology-driven and not usable for all required development activities.

• *Software resources*: Software is also used as a tool to develop other software. CASE tools needed for development, test, and code generation must be considered. Your toolset might include: business systems planning tools, program management tools, support tools, analysis and design tools, programming tools, integration and test tools, prototyping and simulation tools, maintenance tools, cost/schedule estimating tools, and architectural tools.
• *Reusable resources*: Reusable assets are a valuable resource that must be considered in determining your cost requirements. This includes the assets you will develop for future reuse by other programs, as well as searching the reuse repositories for existing code that can be integrated into your development. Reusable assets will have significant impact on your program cost and schedule.
• *Scrap and Rework*: A major factor in both software development cost and schedule is that which is either scrapped or reworked due to the introduction of an error, defect, or failure on initial execution (including costs associated with fixing failures that occur after the system is operational. Rework costs are very high. Boehm's data suggest rework costs are about 40% of all software development expenditures.

Estimating the quality, reviewing is a key activity that can find defects at an early stage of system and software development. Since it is often cheaper to fix defects at an early stage, reviewing is a good technique for improving both product quality and project cost effectiveness because it costs 100 times as much to fix a bug after delivery as during development [12]. It was observed that the average cost of a defect exceeded $5,000 in some organizations where one can improve a lot of things for the cost of few defects!

As two of the most popular defect removal activities, Inspection and Testing are of the most labor-intensive activities in software development life cycle and consumes between 30% and 50% of total development costs according to many studies. However, most of the current defect removal strategies treat all instances of software artifacts as equally important in a value-neutral way; this becomes more risky for high-value software under limited funding and competitive pressures [17].

As Benjamin Franklin explained:

"I conceive that the great part of the miseries of mankind are brought upon them by the false estimates they have made of the value of things." — Benjamin Franklin. [9]

III. MATERIAL AND METHODS

A survey has been conducted for the present study to identify the impact of project cost over the quality of the project since it plays an important role in the delivery and the quality of the project. 50 projects of 5 categories, 10 projects for each (in terms of project funding) were selected for the study as: Highly Expensive, Expensive, Medium, Limited funding and very poor funding. Quality of the project was estimated by giving percentage to the Customer satisfaction like:
*Well satisfied: 81-100%, Satisfied: 61-80%, Good: 41-60%, Average: 21-40%, Not satisfied: <20%

Table 2. Project Cost Vs Quality of the Project

S. No	Project Category	No. of projects taken	Level of Project Cost	Average % of satisfaction by the Software Personnel	Average no.of Defects found	Average No.of Risks	Quality of the project observed after delivery (in terms of Customer satisfaction)
1	A	10	Highly Expensive	>85	150	18	Satisfied
2	B	10	Expensive	>85	245	29	Well Satisfied
3	C	10	Medium	80-85	298	36	Good
4	D	10	Limited funding	<=85	352	42	Average
5	E	10	Very poor funding	<=50	409	49	Not Satisfied

IV. RESULTS AND DISCUSSION

Based on the results shown by the study it was noticed that cost of the project has much influence over the quality of the project. It was also noticed that the customers satisfaction was very poor with the delivered project when the project was completed with very poor funding were more number of risks were identified because it directly influences the resource utilization, man power and many other associated issues. On the other hand, the software personnel satisfaction regarding the project with which they are involved was also correlated with the project cost. This might be due to the work pressure on the limited software personnel employed in the project and also to reduce other software and hardware expenses [17].

Designing the software is a tough task with given requirements while maintaining a balance between software quality, project cost and delivery time. Despite the complex nature of IT, shareholders expect nothing less than continuous improvement in service quality with simultaneous cost reductions. This has been the case in the IT market and major operational focus for many organisations.

V. CONCLUSION

Approaches for identifying risks are usually separate from cost estimation, but technique that identifies risks in conjunction with cost estimation will be useful in improvising the quality of the project. Cost of the project plays a significant role not only in the development but also in the quality of a project in addition to the other external factors such as time, resources etc.

REFERENCES

[1] Apurva Jain, "Theory of Value-Based Software Engineering," PhD Dissertation, Department of ComputerScience, University of Southern California, December 2007.

[2]. Boehm, 1981 - Software Engineering Economics, Barry W. Boehm, Prentice-Hall, 1981.

[3]. Boehm BW, *Software Risk Management.* Washington, D. C., IEEECS Press, 1989.

[4]. Boehm BW, Royce WE, *Ada COCOMO and the Ada process model*, Proceedings, Fifth COCOMO Users' Group Meeting, SEI, October 1989.

[5]. C. Ebert and R. Dumke, *Software Measurement: Establish - Extract - Evaluate - Execute*,Springer-Verlag NewYork, Inc., 2007.

[6]. Charette RN, *Software Engineering Risk Analysis and Management*, McGraw-Hill, 1989.

[7]. Conte S, Dunsmore H, Shen V: *Software Engineering Metrics and Models*. Menlo Park, CA, Benjamin/Cummings Publishing Co., Inc, 1986.

[8]. Farooq SU. Quadri SMK, Ahmad N. Software measurements and metrics: role in effective software testing. Internal Journal of Engineering Science and Technology.2011; 3(1):671-680.

[9]. Franklin, Benjamin, Poor Richard's Almanac, 1733.

[10]. Jensen, 1983 - "A Comparison of the Jensen and COCOMO Schedule and Cost Estimation Models," Jensen R. W., Proceedings of the International Society of ParametricAnalysts, pp. 96-106, April 1983.

[11]. Jones, 1997 - Applied Software Measurement, Capers Jones, McGraw Hill, 1997.

[12]. Keun Lee, "Development and Evaluation of Value-Based Review (VBR) Methods," PhD Dissertation, Department of Computer Science, University of Southern California, May 2006.

[13]. Madachy R, *Knowledge-based risk assessment and cost estimation*, Proceedings of the Ninth Knowledge-Based Software Engineering Conference, Monterey, CA, IEEE Computer Society Press, September 1994.

[14]. Nelson, 1966 - "Management Handbook for the Estimation of Computer Programming Costs, " E. A. Nelson, AD-A648750, Systems Development Corp., Oct 1966.

[15]. Park, 1988 - "The Central Questions of the PRICE Software Cost Model," Park R., 4th COCOMO Users' Group Meeting, November 1988.

[16]. Putnam, 1992 - Measures for Excellence, Lawrence H, Putnam and Ware Myers, Yourdon Press Computing Series, 1992.

[17]. Qi Li, "VALUE-BASED, DEPENDENCY-AWARE INSPECTION AND TEST PRIORITIZATION", PhD Dissertation, Department of Computer Science, University of Southern California, December 2012.

[18]. Rook P, *Cost estimation and risk management tutorial* , Proceedings of the Eighth International Forum on COCOMO and Software Cost Modeling, SEI, Pittsburgh, PA, 1993.

[19]. Software Estimation, Measurement, and Metrics, Chapter-13, GSAM Version 3.0

[20]. SunitaDevnani-Chulani, Bayesian Analysis Of Software Cost And Quality Models, A Dissertation Presented To The Faculty Of The Graduate School, University Of Southern California, 1999.

[21]. The Cost of Software Quality - A Powerful Tool to Show the Value of Software Quality, Randall W. Rice, CSTE, CSQA, CTFL, Rice Consulting Services, Inc.

[22]. The Economics of Software Quality, Capers Jones and Olivier Bonsignour, p-9.

[23]. Wallmuller, Ernest, *Software Quality Assurance - A Practical Approach*, Prentice Hall, 1994.

AUTHOR'S BIOGRAPHY

A high caliber individual with gravitas and a proven 15 years of track record of global IT experience in leading/managing multiple projects and bids covering the Implementation lifecycle of Oracle applications and various other products thus managing the client expectations and relationship. He has a long track record of ensuring projects are delivered to the highest quality, within budget by effectively organizing, managing and utilizing all resources.

Being a Computer Science Engineering Post Graduate (M. Tech CSE) and Master of Business Administration (M.B.A Project Management), has good experience in Account management, Program management and Project Management.

Completed mandatory 35 PDU for PMP Certification and appearing for the exam in Dec.