Using Agility in Measurement to Improve Software Metrics

Garima Rastogi[1] Dr. MP. Thapliyal[2]

[1]Research Scholar, Singhania University, Jhunjhunu, Rajasthan

[2]Reader, H.N.B Gharhwal University,Computer Sc., Srinagar, Garhwal,India

Abstract Traditionally, software development has been managed through plan based approaches that are full of limitations such as poor quality, not meeting customer requirements, unrealistic project development durations etc. To overcome all these problems agile software development methods and techniques have gained attention in the recent few years. The main motive behind adopting agile software development is to improve end-product quality, enhance developer's moral and achieve stakeholder satisfaction. However, agile adoption always comes with special challenges and thus fundamental organizational changes are necessary for successful outcomes. Agile software methodologies are quickly becoming widely used in a variety of industry projects; their flexibility provides the means to address many common problems faced in the development of software system. Agile software projects are characterized by iterative and incremental development, accommodation of changes and active customer participation. The process is driven by creating business value for the client, assuming that the client (1) is aware of it, and (2) is capable to estimate the business value, associated with the separate features of the system to be implemented. With this research we will try to review and discuss the current usage of measurement theory in software engineering also we will outline some steps to help an organization to start and sustain a measurement program.

Keywords: agile, end-product quality, stakeholder, iterative, estimate.

--♦--

1. INTRODUCTION

Measurement lies at the heart of many systems that govern our lives. Economic measurements determine price and pay increases. Measurements in radar systems enable us to detect aircraft when direct vision is obscured. Without measurement, technology cannot function.

There are two kinds of quantification – measurement and calculation. Measurement is a direct quantification, as in measuring the height of a tree or the weight of a shipment of bricks. Calculation is indirect, when we take measurements and combine them into a quantified item that reflects some attribute whose value we are trying to understand.

1.1 Product and Process Metrics

Product metrics are measures of the software product at any stage of its development, from requirements to installed system. Product metrics may measure the complexity of the software design, the size of the final program or the number of pages of documentation produced. Process metrics, on the other hand, are measures of the software development process, such as overall development time, type of methodology used, or the average level of experience of the programming staff.

1.2 Objective and Subjective Metrics

Objective metrics should always result in identical values for a given metric, as measured by two or more qualified observers. For subjective metrics, even qualified observers may measure different values for a given metric, since their subjective judgment is involved in arriving at the measured value.

1.3 Primitive and Computed Metrics

Primitive metrics are those that can be directly observed, such as the program size (in LOC), number of defects observed in unit testing, or total development time for the project. Computed metrics are those that cannot be directly observed but are computed in some manner from other metrics.

2. AGILE SOFTWARE DEVELOPMENT

Agile software development (ASD) is a methodology for the creative process that anticipates the need for flexibility and applies a level of pragmatism into the delivery of the finished product. Agile software development focuses on keeping code simple, testing often, and delivering functional bits of the application as soon as they're ready. The goal of ASD is to build upon small client-approved parts as the project progresses, as opposed to delivering one large application at the end of the project. Agile planning activities for large-scale development efforts should rely on these five levels.

Level 1 - Product Visioning

The highest-level view that the stakeholder can paint of the future is the product vision. In this vision, they explain what an organization or product should look like after project completion. They indicate what parts of the system need to change and what efforts can be used to achieve this goal.

Level 2 - Product Roadmap

The era of large-scale projects that deliver results in years is behind us. Customers demand faster frequent changes, and delivery is measured in weeks or months. The higher frequency and smaller timeframes force a product owner into thinking in steps - into thinking of a road towards the final product. A product roadmap is created and communicated to fellow delivery people to provide a map so concept is more of a reality.

Level 3 - Release planning

In small projects, the product backlog can provide adequate project overview. The size, duration and deliverables are easily recognized, and there is no need to synchronize deliverables or teams. All of this changes when applying agile concepts to programs. The first time when grouping activities and allocating them to teams occurs during release planning.

Level 4 - Iteration planning

For each iteration within the release, a planning session occurs to append detail and increase accuracy. Before or during the session, detail is added to the features by breaking them down into tasks. The actual capacity of the individual teams is known with more certainty than during the release planning session. The combination of these increased accuracies helps the team commit to delivering a number of features during the iteration with a high degree of certainty.

Level 5 - Daily Plan

The stand-up meeting is part of everyday life for agile teams. This daily meeting is not often seen as a planning session, but it certainly is. The people look a day ahead, have learned from the earlier days in the iteration, and tell each other what they plan on doing. Issues are raised, possibly addressed, and the success of delivering the desired features within the iteration can be determined after the meeting.

3. REVIEW OF LITERATURE

Mr. Hassan Hajjdiab and A1 Shaima Taleb; (2011), they have presented an in-depth case study for adopting agile methods to a government entity in the United Arab Emirate (U.A.E). Mr. Lehman, T.J.; Sharma, A; (2011), they have presented the various forces that influence both the client and development organizations that are engaged in Software Development as a Service. Mr. Nagy, A; Najima, M; Mkrtchyan, L; (2010), they have designed and developed a project health measurement model to evaluate the factors affecting software development of the project. Mr. Patcha, K.K.; (2009), they have presented simple, easy to understand approach to developing business application software using agile techniques and concepts yet still remaining true to the RUP. Cohen, S.J.; Money, W.H.; (2008), they have presented a “Bridge” method for successfully integrating agile concept with traditionally linear and sequential software development lifecycles.

4. PURPOSE OF THE STUDY

Although there is no consensus on what metrics work best for a specific organization, it is still a widely accepted belief that software metrics can help organizations in achieving better productivity. In the course of developing software, it is common for software teams to focus more on the deliverables. For them to be sure that they are on the right path of achieving their business goal, some kind of measurement program is needed to measure and monitor the progress correctly and efficiently. A well planned measurement program tailored for a specific organization can help provide specific information for managing the multitude of concurrent projects, tracking roadblocks, and suggesting improvements in the software development process thereby leading to fulfillment of the short term as well as long term project and business goals.

The fact that various models of software development are being followed in the industry, it is difficult to propose best practices for all the approaches in this research. With this study, we make an endeavor to collect and review the existing methodologies for establishing a successful metrics program in software development industry and thus outline some steps to help an organization to start and sustain a measurement program.

5. Discussion: Software Metrics and Industry

It can be a challenging task for a management to analyze how to gain maximum quantitative output from the metrics program. There can be various factors which affect the decision if an organization thinks to change the measurement program or design a new metrics program like management may be concerned about the financial aspects, project manager may be concerned about timelines due to changes and team workers may be concerned about the extra workload.

We tried to define certain steps which should be followed while selecting, designing and implementing a successful value added metrics programme.

Step 1- Identify metrics consumers

Consumer is a person who will take decisions and actions based on the metric programs. These consumers cab of different type and each can have different type of interest in the metrics outcome. Consumer can be –business managers, software project managers, programmers, administrators, quality managers etc.

Step 2- Identify business goals

This is the most required step to identify the business goals which the software project and organization needs to achieve. Basili and Rombach have defined a Goal / Question/Metrics paradigm, which can be used to refine goals and design measures based on them.

In GQM basically main focus is on specific needs of the software project and of the development organization. Once the measurement goals are defined according to the management goals then refined into questions and questions are refined into metrics.

Step 3- Identify quantifiable questions

After the business goals for metrics are identified now there is a need to define the quantifiable questions that need to be answered in order to ensure that each goal is being covered by the metric.

Example –if goal was to tracking defect and fixing defects the iterative improvement session questions can be –

-how critical are the defects reports?

-what kind of defects reported?

-are these test cases needs to be updated? Etc

Step -4 Identify Metrics

The next step is to select a metrics that matches with the goals of the business and needed to answer the questions decided upon.

Example questions –

- Are we on track of project goal?

Metrics – Functions completed.

- How many visitors on the webpage are converted into consumers?

Metrics – number of registration / number of visitors * 100

- How many revenue are we making from each visitor?

Metrics – Product registrations / number of visitors

Step -5 Selection of Measurement Model

Software measurement can be Fundamental or Derived. Fundamental deals with the measurement process that takes place in the initial phase of model development while derived takes place later when new measures are needs to be defined based on existing one. Derived metrics are quite complex and are modeled using mathematical combinations.

Define various models of measurement and select the models by making checklist. Verify all the attributes of a model through checklist and select the most appropriate one.

Step -6 Analyses and Reporting Structure

Now there is need to define and develop the reporting of the selected metrics. Within an Organization various report audience may have different goals and requirements of reporting metrics. These reports help them to further analyze and steam like the project process. This step will involves defining report audience, reporting format, use of reports etc.

Step -7 Method to collect data

This step involves defining who collects the data needed by the metric program, what kind of data needs to be collected and how to collect it. Various methods can be used – by survey, test cases, observing behaviors etc.

6. CONCLUSION

Implementing a successful metrics program may involve a lot of incremental changes to the processes and procedures being followed within the organization. This may be time consuming and the gains may only be noticeable after a period of time. As the metrics program implementation may span a couple of years, it may also suffer from a loss of momentum. To avoid such weak spots management must continuously reinforce and convey the benefits of implementing the metrics program to employees. A positive work culture focused on discipline, patience and team work should be promoted by the management. After reviewing various research articles related to software metrics, we have proposed a seven step methodology which may be useful to organizations to start and sustain a measurement program.

REFERENCES

[1] Mr. Haajjdiab, H.; Taleb, A.S (2011); “Agile adoption experience: A case study in U.A.E.”, IEEE, Pg(s): 31-34.

[2] Mr. Tobin, J. Lehman; Akhilesh Sharma (2011) “Software development as a service: Agile experiences”, Annual sr ii global conference, IEEE, Pg(s): 749-758, .

[3] Mr. Kumar, K.; Gupta, P.K.; Upadhgey, D.(2011) “Change oriented adaptive software engineering by using agile methodology’’,3rd International conference , IEEE ,PAGE(s): 11-14.

[4] Mr. Aniche, M.F.; DE Azevedo Silveria, G.(2011) “Increasing learning in an agile environment: Lessons learned in an agile team”, AGILE CONFERENCE, Pg(s):289-295.

[5] Mr. Nagy, A.; Njima M.; Mkrtchyan, L.(2010) “A bayesian based method for agile software development planning and project health monitoring”, 2nd International conference on intelligent and collaborative systems (INCOS), IEEE, Pg(s):192-199..

[6] Mr. Madison, J.(2011) “Agile architecture interaction” IEEE, PAGE(s): 41-48.

[7] Mr. Patcha, K.K.(2009) “Agile edi framework for b2b applications”, International conference, IEEE, PAGE(s): 1-3.

[8] Mr. Maher, P.(2009) “Weaving agile software development techniques into a traditional computer science curriculum”, 6TH International conference, IEEE, PAGE(S): 1687-1688.

[9] Mr. Cohen, S.J; Money, W. H.(2008); “Bridge methods: Complementary steps tegrating agile development tools and methods with formal process methodologies”, Hawaii International conference on system sciences, IEEE, PAGE(S): 460.

[10] Mr. Kunz, M.; Dumke, R.R.; Zenker, N. (2008) “Software metrics for agile software development”, SOFTWARE ENGINEERING, 19TH AUSTRALIAN CONFERENCE, IEEE, PAGE(S):673-678.

[11] Mr. M.C. Gregor, J.D(2008); “Agile software product line – A Working session”, Software product line conference, 12TH INTERNATIONAL CONFERENCE, IEEE, PAGE(S):364.

[12] Mr. Cohan, S. (2007) “Successful integration of agile development technique within DISA (DEFENSE INFORMATION SYSTEM AGENCY)”, IEEE conference, PAGE(S):255- 261.

[13] Mr. Rowley, D.; Lange, M. (2007) “Forming to performing: The evolution of an agile team”, IEEE, PAGE(S): 408-414.

[14] Mr. Smits H(2007) “The impact of scaling on planning activities in an agile software development center”, 40th HAWAII INTERNATIONAL CONFERENCE, PAGE(S): 274.

[15] V.R. Basili, H.D. Rombach (1988), “The TAME Project: Towards Improvement – Oriented Software Environments, IEEE Transactions in Software Engineering.

ABOUT AUTHORS

Garima Rastogi has done M.Tech in Computer Science and MCA. Presently she is pursuing her PhD from Singhania University under the guidance of Dr. M.P. Thapliyal. She is working in the department of Computer Sc as an Assistant Professor at NIET, Greater Noida. She has got more than 5 years of industry and 8 years of teaching experience and published 11 national and international research articles in refereed journals. Also she has attended and presented papers in several conferences.

Dr Thapliyal is PhD in Computer Sc. and working as Reader in Department of Computer Sc. in H.N.B. Garhwal University, Srinagar. He has published several papers in National and International Journals. He has supervised many PhD reaserach scholars in the field of computer sc for several years.

ACKNOWLEDGMENTS

We want to thank Dr. Hemraj Verma who helped me to collect data and suggested us various steps, Dr M.P. Thapliyal who gave me the idea of the topic.