Measuring the Effectiveness and Quality of Agile Projects
Martin Kamau Karanja¹
¹Dr Michael Kimwele

Abstract: Since agile methodology is a fairly new technique, there is need to come up with new metrics that focus on the primal spirit of agile should be developed for an effective measure of agile software quality. The challenge of coming up with effective metrics for agile project is that every project team tend to decide what are the metrics to put into consideration when testing completeness and quality in a project.

There is however key metrics that are similar in the quality test applied by each agile method; be it scrum or extreme programming (or any other). The idea is to harmonise and come with a standard way of effective measure of a quality agile project. This paper seeks to identify and highlight some of the agreed metrics that will effectively address this. In the end it is agreed that the final measure of quality and effectiveness should produce a product that is highly functional, minimal Maintenance, Predictable base code that is high quality and sustainable. The software should also be easy to replicate the base code. Since testing is done at every iteration, defects should be caught early and not after the release. The involvement of users and stakeholders require the schedule should be predictable and the process to be transparency to the entire team.

Keywords: Agile, Cost of Software Quality, Scrum, Extreme Programming & Agile Methodology

Introduction

Agile is people-centric, development and testing is performed in an integrated way, self-organizing teams encourage role interchange ability, customer plays a critical role and Project Life-cycle is guided by product features. (A B M Moniruzzaman Dr Syed Akhter Hossain, 2013)

The current challenge is that it is a fairly new methodology in software development and as such there are no standard metrics to test quality. Another challenges arise when the project manager tries to fall back to the old ways of measuring performance. This is where from the planning level, the plan is pre-set on how to evaluate performance and is implemented on the team and testing of the product is only done at the end of the project.

From traditional Quality Assurance we know Software Metrics as a very good approach to measure Software Quality. Using some well-known metrics custom made for agile can give a promising approach to control and ensure the internal Software Quality. The objective of developing matrices to measure and test an agile project is to achieve higher quality and shorter lead times with minimum overhead, frequent deliveries, close teamwork with team and the customer, continuous integration, short feedback loops and frequent changes of the design.

It is then desirable to completely embrace the agile methods for a fair and effective test of the quality of agile project. If married with the traditional model, the output may not be of high quality as would be expected. The traditional methods tend to focus too much on affecting a pre-set plan and sometimes forget the product quality. They also test the product at the very end of the project. Testing a product from the early saves a lot of time and money. (Rawad Abou Assi, Wes Masri, Fadi Zaraket, 2016) That is why agile project will include testing step. This will be done incrementally until the product is complete. We will look at various measures that have been effective in testing the quality of agile projects. The test methods and measures may differ from organization to organization. The parameters to be tested however remain the same.

Discussion

Traditional Development Methods insists on carefully planning every detail before starting a project, and then execute it while carefully tracking and reporting against the plan. The Agile process is an acute departure to this mind-set, it believes in “responding to change over following a plan” as stated in its manifesto. (LO, 2013)

Iterative project development principles have been around for some time. Agile development model has taken it and made it acceptable to project management professionals. Agile therefore is a project management model that is ideal for exploratory projects in which requirements need to
be discovered and new technology tested (Larson, Erik W 2011). It focuses on active collaboration between the project team and customer representatives, breaking projects into small functional pieces, and adapting to changing requirements.

Figure 1 Definition of agility - Copyright Ernert MnKandla PhD Thesis University of the Witwatersrand

Ken Schwaber one of the founders of agile methods, explains agile using the analogy of building a house. Traditionally, a buyer cannot move into a house until it is “done” entirely. In incremental, iterative approach, the house is built room by room. The fittings would be input in the most important rooms including plumbing and electrical. Each time a room is completed, the team including the buyer of the house would assess the progress and make adjustments as the buyer deems fit. The buyer can choose to add rooms from the initial plan or remove some rooms. In the end, the house will be as the customer seems fit.

With this analogy, it is very apparent that traditional plan driven method of development cannot work. Marrying the two also tends to very tedious hence the need to develop quality assurance tools that can be used specifically for agile projects. The challenge with agile project then is to identify, survey, select, appropriate tools for measuring business value add of agile project and the quality of software projects (Mariana de Azevedo Santos1, Paulo Henrique de Souza Bermejo2, Marcelo Silva, 2013). Product success is defined as Information quality, service quality, user satisfaction and net benefit to the business. The parameters proposed come to focus especially because user satisfaction and quality of product is a primary focus in agile methodology.

### Measurement of Quality

According to (AMBLER, 2005) agile quality is as a result of practices such as effective collaborative work, incremental development, and iterative development as implemented through techniques such as refactoring, test-driven development, modeling, and effective communication techniques (Mcbreans, 2003) Defines agile quality assurance as the development of software that can respond to change as the customer requires it to change. This implies that the frequent delivery of tested, working, and customer-approved software at the end of each iteration is an important aspect of agile quality assurance. (Garvin, 1984) Five major approaches to the definition of quality can be identified:

1) **Transcendent** approach of philosophy - quality can be recognized but not defined
2) **Product-based** approach of economics - quality is tied to inherent product characteristics
3) **User-based** approach of economics - quality is fitness for purpose, marketing, and operations management
4) **Manufacturing-based** - quality is conformance to specification
5) **Value-based** approaches of operations management- quality depends on the amount the customer is willing to pay for the product

Agile is defined in sympathy to the agile manifesto by most publishers. It is an iterative and incremental (evolutionary) approach to software development which is performed in a highly collaborative manner by self-organizing teams within an effective governance framework with "just enough" ceremony that produces high quality solutions in a cost effective and timely manner which meets the changing needs of its stakeholders ( A B M Moniruzzaman Dr Syed Akhter Hossain, 2013)

This agile process is an iterative process in which changes can be made according to the customer satisfaction. Agile process new features can be added easily by using multiple iterations (Sheetal Sharma et al, 2012).

Below we have at table with the general parameters considered in a quality software development. Agile has to address each to suit its process. As agreed before, it is a new methodology and as such, it is still evolving. The parameters have been successful in testing quality of the traditional methodologies. They are key to quality assurance. Agile then works within the predefined quality framework and addresses each parameter uniquely to ensure measurability of its quality. (Janus, A., Dumke, R., Schmietendorf, A., & Jager, J., 2012)

<table>
<thead>
<tr>
<th>Agile</th>
<th>Incremental</th>
<th>Iterative</th>
<th>Self - Organizing</th>
<th>Emergent</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Assists with Modification of design and its requirements</td>
<td>Develop separately and integrate later</td>
<td>Lean team that organize internal dynamics</td>
<td>Requirements, tools and techniques emerge in the</td>
</tr>
</tbody>
</table>
Some of the parameters measured for quality output in agile projects are mostly done to encourage investor confidence. Agile projects are forever changing this means it is not possible to know the cost of the whole project beforehand. This does not go well with financiers and business owners. The platform used before agile has always been where a plan is laid out then a budget is agreed on beforehand with very few adjustments along the way. Since projects exist to solve business problems, there has to be a way to measure agile projects. One of the things to have in mind is that every business emphasizes on one parameter over another depending on their outlook. Agile metrics are normally used to trigger either a discussion or a decision. The team has full access to the set goals. They are also part of the decision makers and are able to articulate the way forward to solving a certain problem. This allows ownership to the project and all inputs are put into consideration before the next iteration. The entire team understands what is being measured and are thus motivated. They own the decision therefore work whole heartedly rather than through compulsion. Software metrics can be classified into three categories: product metrics, process metrics, and project metrics. Product metrics describe the characteristics of the product such as size, complexity, design features, performance, and quality level. Process metrics can be used to improve software development and maintenance. Examples include the effectiveness of defect removal during development, the pattern of testing defect arrival, and the response time of the fix process. Project metrics describe the project characteristics and execution. (Kan, 2003) Agile focuses on the product quality by measuring customer satisfaction. The agile software is never 90% completed. It is either done or not done. The term “done” is used to define a ready to ship increment functionality. For the software to be “done” it should have High valuable functionality,

- Low Maintenance functionalities
- Predictable base code that is high quality, predictable and sustainable
- Replicable easy to enhance code base.
- Defects to be caught early and not after the release.
- The schedule should be predictable
- Transparency to the entire team

The table 2. above are the most authors agree on the common characteristics of agile.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iterative</td>
<td>The main objective of agile software processes is satisfaction of customers, so it focuses on single requirement with multiple iterations.</td>
</tr>
<tr>
<td>Modularity</td>
<td>Decomposes the complete system into manageable modules. Run the modules in parallel.</td>
</tr>
<tr>
<td>Time Boxing</td>
<td>Divide the project into predictable short time stints. That are incremental with time.</td>
</tr>
<tr>
<td>Incremental</td>
<td>Developed in parallel and quick cycles; when the increment is completed and tested, it can be integrated into the system.</td>
</tr>
<tr>
<td>Adaptive</td>
<td>Due to the iterative nature of agile process new risks may occurs. The adaptive characteristic of agile process allows adapting the processes to attack the new risks and allows changes in the real time requirements.</td>
</tr>
<tr>
<td>Collaborative</td>
<td>Customers and developers should be work together in the development process and should provide feedback in a regular and frequent fashion.</td>
</tr>
<tr>
<td>People Oriented</td>
<td>Focus on people over technology. When empowered people productivity and performance increase.</td>
</tr>
</tbody>
</table>
Software quality consists of two levels: intrinsic product quality and customer satisfaction. (Kan, 2003)

To estimate quality in a software can be difficult especially because it is based on intangible characteristics like looks and feels. That is why most engineers default to bugs and defect count. They are easily identified especially in low quality software. (Boerman, M. P., Lubsen, Z., Tamburri, D. A., & Visser, J., 2015) Having said that, for an agile project that is people centric, there must be a guide to measuring quality. As such, the team and the software must be measured so that at the end, the product can be said to be of high quality; the primary focus of agile project.

**Conclusion**

Quality agile project should have a software which is easily understood so that a good developer can tell how and when events took place. On increment in any functionality, there should be no ripple effect due to poorly designed dependencies. The code should be legible and easy to understand even without the developer or data and variable definition. Test check should be automated as much as possible for functionality check. The standards defined and followed should be well known. The entire team should understand all that is happening and should participate in decision making. The investor should see the business value and experience returns on investment. (Gravett, L. S., & Caldwell, S. A., 2016)

<table>
<thead>
<tr>
<th>Metric</th>
<th>Parameter</th>
<th>Unit of measure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team motivation</td>
<td>How motivated the entire team is.</td>
<td>Linear scale lowest 1 highest 5</td>
</tr>
<tr>
<td>Return on Investment</td>
<td>The value of the completed work against investment cost</td>
<td>Value point v/s story point</td>
</tr>
<tr>
<td>Velocity</td>
<td>Relative measure of work done per sprint</td>
<td>Measure of velocity per iteration</td>
</tr>
<tr>
<td>Accuracy</td>
<td>How accurate the time work estimate is</td>
<td>Estimate Amount of effort/actual amount of effort</td>
</tr>
<tr>
<td>Efficiency</td>
<td>How team utilize their time to accomplish the agreed workload</td>
<td>Actual work time/Calendar time</td>
</tr>
</tbody>
</table>

**Table 3. a proposed metrics for agile quality measure**

**Summary**

Agile project measures are only a challenge if they are developed with the traditional project monitoring methods. This is because the goal of each model is very different and therefore the parameters used to measure effectiveness and quality is very different. Once this concept is acceptable and agile project parameters are pre-set with agility in mind, the project success is a guarantee and product acceptability by the stakeholders is much higher in comparison to the output of the traditional project models. The entire team of stake holders, developers, management, testers, customers and all work hand in hand all aiming towards one goal; output a quality highly functional software.

The measures used to output a quality agile project are a variant depending on the business view. However the parameters tend to revolve the same orbit. At the end of the day what is required is a high quality product with maximum business output. The key measures are more focused to a motivated team rather than a plan driven team. The focus is more on the product rather than the planned goal. This is more satisfying to the customer more than anything else. Granted, it is a fairly new school of thought which takes time to be integrated by mature software development companies and even businesses that require software for their information system. The companies that have adopted agile systems entirely do not want to go any other way.

**References**


