Applicability of Agile Methodology in Software Development – Scrum as a reference

SILVA, Francisco Eronildo da [1]

PINTO, Aurílio Guimarães [2]

FREITAS, Caio Guimarães [3]

ALMEIDA, Cristiany Caliri de [4]

RIBEIRO, Dallas dos Santos [5]

LEITE, Francisco Canindé da Silva [6]

OLIVIERA, Geveson de Souza [7]

MORAIS, Gilvanete Melo de [8]

PERES, Paulo Júnior de Jesus [9]

Summary

This study aims to analyze the speed that the agile methodologies give the software development process, showing how the companies use these methods as a way to reduce time and effort in software development, taking as reference the SCRUM methodology. The agile methodology part of the premise on which the results should be reached quickly without compromising the quality of the final product (software), accordingly the SCRUM is a methodology that aims to improve the planning of software projects whose premise is break the product into smaller pieces and so deliver the functionality without client wait too long to view them. Among the authors searched for the Constitution of this work,

conceptual highlights Arag**on Fernandes (2014), Somerville (2007), Roger Pressman (2011), Kim Pries (2010) and Ken Schwaber (2014).** Th**e most imp**ortant conclusions are that the use of agile methodologies can give rapid software development, showing more effectiveness, dynamism and offering advantages to companies that adopt the methodology, these facts, demonstrated in this work.

Key words: Agile, Scrum, governance, Software Engineering.

1. Introduction

The agile began to have an emphasis on the 80. The site DEVMEDIA says: "agile methodologies have been around for years, since the 80, but some information go through distortion, which made it difficult in the beginning the use of methodologies. Therefore, developers have come to understand the agile methodology as something that can, in other words, we can develop without documentation, no default and carelessly. This is not true, the agile methodologies can bring success to the project, and are used in the industry. " This work demonstrates that the renowned authors-based agile kill the Organization and the necessary care that the software project need to be efficient and effective, and every day, you get the software development market, being used in industries and public agencies who hire outsourced factories to develop their systems.

This study li**mits itse**lf to show the benefits agile brings to the software development. As an example if you have the modus operandi of the CTIS, software company that operates in the domestic market and adopting the common software development methodology, but that's depending on the client, opting for the agile methodology.

The Superintendency of the Manaus free trade zone-SUFRAMA, which, through bidding, hired the service of CTIS and opted for the move to agile methodology.

SUFRAMA is a federal authority responsible for managing tax incentives. Most of their systems are critical and operate with limitations, so bid the software factory service to work with agile methods. The option for Agile methods in reason of fast delivery and quality because the traditional engessava the process methodology.

The agile method dispenses with a stack of paper, focuses on the quality of the product, because what matters to the client is the software working.

Is this not go into the details of the change in methodology adopted by SUFRAMA, and Yes, illustrate that various industries and Government agencies, are opting for change of your software development model for Agile methodology.

Despite being an option for software development, agile methodology, which offers various benefits to development certainly may not be suitable for all projects. For a better understanding of the subject, you should review some of the history of agile development.

Th**e general obje**ctive of this work is to do a systematic review of the agile methodology. The SCRUM method will be the object of this study, where they will be listed the negatives and positives.

This research **is justified** by the need that the software industry has to carry out deliveries of products with the least possible time to their customers, without losing sight of the quality, economy and satisfaction.

T**he methodol**ogy of this study is descriptive and explanatory research, having as the bibliographical data collection.

2. Development

This study begins by reviewing the concept of Software that was originally proposed in 1968, at a conference organized to discuss what was then called the "software crisis". The software crisis resulted indirectly in the introduction of a new computer hardware based on integrated circuits. The application of integrated circuits made of computer applications, considered not feasible, viable proposals. The resulting software was larger and more complex than previous systems software (SOMMERVILLE, 2007).

Software Engineering is a branch of engineering, whose focus is to develop within appropriate cost high quality software systems. Software Engineering is a layered technology, involving tools, methods, process and focus on quality. (SOMERVILLE, 2007). Any engineering approach (including software) must be grounded in an organizational commitment to quality.

Total quality management Six Sigma1 (GYGI; DECARLO; WILLIAMS, 2008) and similar philosophies they promote a culture of continuous improvement processes, and it is this culture that, in the end, leads to the development of increasingly effective approaches in Software engineering. The cornerstone that supports the software engineering is the focus on quality (PRESSMAN, 2011).

With respect to the history of agile development, the same began in 2001, with the "Manifesto for Agile software development", which was signed by Kent Beck, an American software engineer, creator of E*xtreme Programming* a*nd Test-Dr*iven, and sixteen more renowned developers. Details of this fact can be verified at the address https://www.agilealliance.org/agile101/the-agile-manifesto/.

The manifesto emphasizes individuals and interactions over processes and tools; the software running more than a complete documentation, collaboration with the customer instead of negotiations of contracts and responses to change over following a plan.

This does not take away the importance of the documentation or processes, and in no way relate the ineffectiveness of tools, means, however, that the delivery of the software is more valued, as declaring Pressman:

"The agile software engineering blends philosophy with a set of principles of development. The philosophy advocates the customer satisfaction and the delivery of prior incremental; small project teams and highly motivated; informal methods; minimum software engineering artifacts and, above all, simplicity in general development. The principles of developing prioritize delivery more than analysis and design (although these activities are not discouraged); also prioritize the active and continuous communication between developers and customers ". (Pressman, 2011)

A project involves people and changes, especially when it comes to deliveries. In this way the agile

methodologies working with highly motivated teams, because they are linked directly to the process, involved in every part of it, feeling the responsibility upon himself the success of work and knowing that you have the possibility to support the changes during the entire process of development.

In agile development, you can't make a plan complete with everything that we should perform to then start the development, without contact with the client, instead, developed incrementally, i.e. the product is done gradually and consistently delivered this way, any necessary change requested by the client or seen by project members, at any time during development, not full damage will result and the change can be performed without major damage, because the project is under development and has not been completed.

The initial increments of the system can provide a high priority feature, so that customers may soon get value for your system development. Customers can view the requirements in practice and specify changes to be incorporated in subsequent releases of the system.

In this way, the client can make use of system resources more quickly. What would take months to be delivered in weeks, and can verify bugs and specify new changes or improvements, not needing to get to the end of the development to see the problems.

This methodology also provides a greater knowledge of the system to the team, because it will understand the business to develop it, making it, with greater speed and accuracy, and in case of errors, the team will lose less time for analysis, and may fix the error quickly.

According to Aguinaldo Aragon Fernandes (2014), initially, the Scrum was designed with sharp focus in the delivery of software development projects in a complex environment. However, it has been increasingly applied in product development projects of other natures ".

Aragon also States that:

"The Scrum consists of iterative and incremental method for managing complex projects, whose objective is to ensure faster delivery and maximize adherence to customer requirements, cooperation between team members and the productivity of each participant. Is one of the methods of so-called "agile" more widespread in the it market, (Aragon Fernandes 2014).

Agile methodologies can be applied in resolutions of complex problems, such as software development informs Schwaber:

"Complex project situations occur when the complexity of the intermediate activities does not allow a defined process controlled, you can generate a loop products in acceptable levels of quality. The complexity of a project is directly proportional to the complexity of the requirements of the customers and the technology involved, ale to be largely dependent on the characteristics of each team member, taking into account the diversity of skills, knowledge, attitudes etc, which can be found in any group of people ". Schwaber (2004)

In situations like this, Schwaber recommends using the empirical process control concept, which uses mechanisms such as self-organization and sense of urgency with the following pillars:

"Visibility: all aspects that affect the desired results should be visible to control processes.

Inspection: the various aspects of the process should go through regular inspections in order to detect unacceptable variations.

Adaptation: the process or its intermediate products must be adjusted after inspection to minimize future deviations more severe, case characteristics and results are outside the acceptable limits and jeopardize the acceptance of the final product. Schwaber (2004)

The Scrum is structured in a set of practices conducted by teams in specific roles, organized into a stream of activities or fixed-duration events, totally controlled with artifacts and well-defined rules, which aims to obtain usable products at intervals short of time.

Every Scrum practices are based on a "skeleton" represented by successive iterations of development activities, (each one generating an increase of product), inspected and adjusted daily by members of his own staff, and oriented for a list of initial requirements.

At the beginning of each iteration, the team reviews what should be done and determines what viable functionality to be delivered at the end of the iteration. The team is free to use your best effort in the remainder of the iteration and features at the end the final product built.
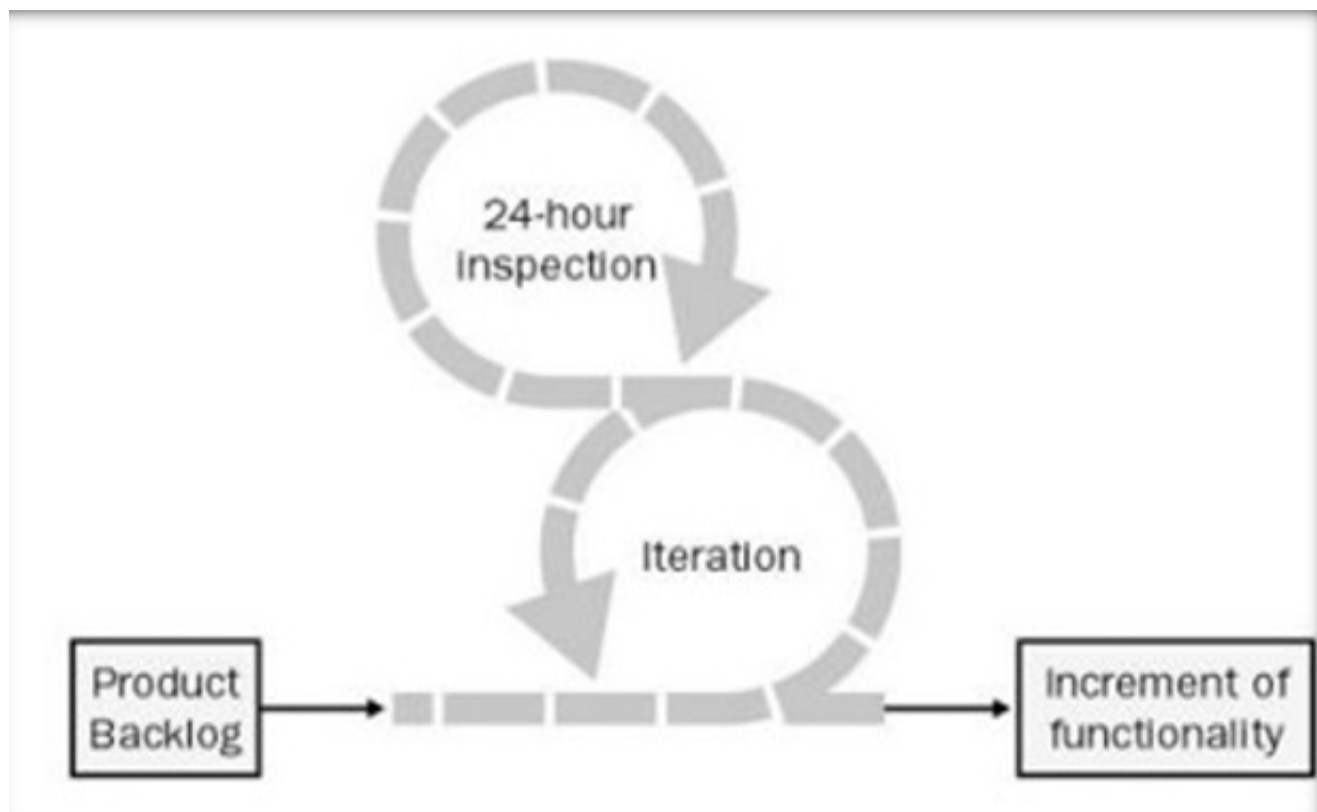
The figure below shows the flow of Scrum:

Figure 1: the flow of the Scrum – adapted from Schwaber (2004)

On a Scrum project, all the responsibilities are divided between three roles:

ProductOwner: person responsible for managing the product Backlog (ensuring that is visible to all), generate and disseminate the project requirements, as well as the plan for successive deliveries, prioritizing the results that will bring greater added value to the project.

Scrum Master: responsible for implementing the Scrum method, as well as teach you to everyone involved in the projects and ensure that all follow the rules and practices.

Scrum team: development group collectively responsible for the success of each iteration and the project as a whole, must be composed of multi-disciplinary people capable of self-organization and self-management.

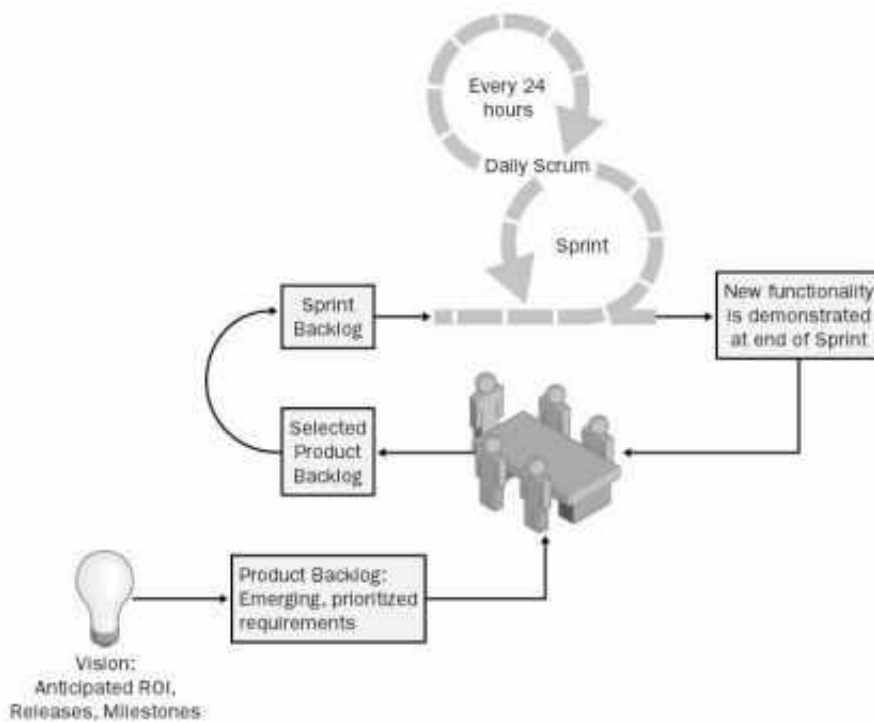The process advocated by Scrum covers the following elements as shown below:

Figure 2: elements of Scrum – adapted from Schwaber (2004)

T**he vis**ion: must be prepared by ProductOwner, including releases and plan the product delivery milestones every Sprint, in order to maximize the return on investment of the product development project.

T**he Backlog of produc**t: also prepared by ProductOwner, contains a list of the functional and nonfunctional requirements, prioritized and divided into releases (Sprints).

T**he Sprint planning meeting: the p**roject is divided in Sprints lasting thirty calendar days each, to be performed one after the other, without interruption. The planning is done in a meeting with the participation of the Scrum Team and by ProductOwner, in two periods of 4 hours each:

In the first hour is set the scope ("what") will be treated by Sprint, through the selection of the product Backlog requirements that will be placed on the Sprint Backlog.

In 4 hours late, the actual planning of the tasks to be performed in the Sprint, ("how") and the official start of the Sprint, at which time starts to run the period of 30 days.

Sprint: the own product development iteration, which has a fixed duration. A Sprint includes their planning meetings, review and retrospective.

The **Daily Scru**m: daily meeting of fifteen minutes, where each team member answers the following questions:

- What I did on the project since the last Daily Scrum?
- What I'm planning to do until the next Daily Scrum?
- Is there any restriction or impediment to that I honor my commitment of the current Sprint and/or the project?

In addition, the team synchronizes all activities and program meetings necessary for the continuation of the project.

Detailing a little more parties views so far.

The Sprint review meeting: in 4 hours, the Scrum Team presents to the ProductOwner (and other stakeholders) the work generated in Sprint and determines among them what to do in the next Sprint.

The Sprint Retrospective meeting: in 3 hours, the Scrum Master encourages the team members to review the Scrum development process your birth practices and Scrum process model, in order to make it more effective and rewarding for the next Sprint.

According to Schwaber (2004), the Sprint planning meetings, Daily Scrum, review and retrospective of the Sprint are, together, the empirical inspection practices and adaptation of the Scrum.

There are two categories of artifacts in the context of the Scrum: the Backlog tables and graphics that show the work that still lack until the end (named BurndownCharts).

The Backlogs are tables: product Backlog consists of a "live" document developed and maintained by ProductWoner which, by definition, is never full (since there are always improvements to be implemented in a product until it is finally removed from circulation). Contains a list of all the changes that will be made in the product for future versions (features, functions, technologies, adaptations, enhancements, fixes, etc.). such requirements are ordered by priority and detailed in terms of description attributes, complex factors/adjustments and estimates (of effort and term) along the future Sprints.

The Sprint Backlog: defines the tasks that the Scrum Team must perform to create product increments (from the product Backlog) while running a Sprint. The your details should be enough for you to be accompanied at meetings of the Scrum Dario, in tasks that last between four and 16 hours.

Each task should be documented, at least in terms of your responsible, the status (not started, in progress, completed) and the amount of hours of remaining work every day of the Sprint.

The BurndownCharts show, graphically, the amount of total work (other efforts) over time, reflecting your correlation with the progress of the project teams in reducing your work. Can be used in the context of the product Backlog (including all Sprints) or inside each of the Sprints (Sprint Burndow).

The Scrum, as previously mentioned, was originally created for use in software projects in complex environments, i.e. where the requirements change with some frequency, that may have your scope or your work breakdown structure or WBS EAP project organized and structured into packages of incremental, consistent and usable artifacts, to be delivered to the client in successive periods of fifteen to thirty day each.

At first this concept is perfectly applicable to any project or programme whose objective is the development of products or services of another nature, or even that involves improvement initiatives through the use of methodologies with Lean, Six Sigma, etc. In short, the Scrum is a recommended approach, which has shown strong applicability, for projects that require the combination of skills and knowledge focused to a team and involving collaborative efforts.

Second Pries & Quigley (2010) there are ways to adapt the Scrum for application in various types of programs and complex projects, such as:

Combining with traditional methods of project management: can connect concepts and artifacts, such as WBS (work breakdown structure) and product Backlog, earned value analysis, the BurndowsCharts and the Communication Plan, control of meetings Sprints (planning, daily, review, retrospectives) etc.

Managing complex programs: adoption of a Scrum to Scrum, where the Backlog of product can be broken down into sub-backlogs, each being consumed by your respective Team Scrum.

Expertise in functional areas serving various projects (for example, teams of testing or quality assurance): in the product Baclog, can come in various designs and tasks in the Backlog at a Spint, those tasks that fit within thirty days.

Combined with the technology in the form of "cascade": you can split the schedule in fixed duration model, in order to synchronize, for example, a sequence of Sprint with a milestone (milestone) foreseen in the project, as well as had the activities of verification and validation of the form evolution in each Sprint.

Combining with the Six Sigma approach: you can wrap each of the phases of the DMAIC methodology (Define, Measure, Analyze, Improve, Control) in a Sprint, running one after another.

Schwaber (2004) mentions the possibility of using the Scrum in a context of price and contract duration prefixed. In these cases, the Backlog of product can be used not only to demonstrate to the customer that the requirements were understood, but also the priority of each to value generation was understood. The most relevant requirements can be selected for the first few Sprints and the increments in functionality reliably every follow-up meeting.

It is important to stress that the adoption of Scrum for an organization should be done judiciously and that there are many challenges to be faced.

Let's connect some of the points that, if not well managed, can jeopardize the effectiveness of Scrum methodology:

It is essential to have a well-functioning team working group, since the success of the work depends on intensive effort on the skills that each one has as a differential;

It is important that each Member of the Scrum has a strong sense of self management.

Ensure that team members are allocated to only one project.

You must ensure the commitment of all involved (especially from those who represent the client).

You must ensure that the Backlogs are well documented, so there are no misunderstandings between those involved.

There may be some difficulties to "atomize" the tasks to be placed in each line of the backlogs, as well as to establish the dependencies between them, which can affect the planning and the good progress in the implementation of the Sprints.

You must ensure that all meetings (planning, daily, review, retrospective) of the Sprints are carried out and that the fixed times are effectively fulfilled, at the risk of damaging the sense of discipline, which is crucial to the success of the method.

Conclusion

Before the quotes from various authors, is notorious for agility and speed that the agile methodologies, in particular the SCRUM, chosen to illustrate the other in this work, give businesses that hire software factories.

Among the advantages, we can highlight: the greater agility in the control and management of the progress of work, emphasis on teamwork and focus on rapid results, shared responsibility with the group causing a greater sense of commitment, faster deliveries and efficiency, shorten feedback emphasizing communication and increasing customer satisfaction by having the software delivery time reduced, without losing the quality and increased the company's profit.

References

FERNANDES, **A. A**.; **ABREU,** V.F.: Deploying the governance, 4th ed., São Paulo, SP: Brasport Books and multimedia publishing company Ltda., 2014.

PRESSMAN, r. 2011 Software Engineering A professional approach. Translation Ariovaldo Griesi, Mario Moro Fecchio. 7th ed.-São Paulo, SP: MGH Editora Ltda, 2011.

PRIES, Kim H., QUIGLEI, Jon M. Scrum Project Management. CRC Press, 2010

SOMMERVILLE, i. Software engineering. 8. Ed. São Paulo, Pearson Addison Wesley, 2007.

SCHWABER, Ken. Agile Project Mangement with Scrum. Microsoft Press, 2014.

The agile manifesto. Available in: <https: www.agilealliance.org/agile101/the-agile-manifesto/="">. Accessed on 21 October 2016.</https:>

Manifesto for Agile Software development. Available in: <http: www.manifestoagil.com.br/="">.</http:> Accessed on 21 October 2016.

An overview of agile methodology. Available in: <http: www.devmedia.com.br/uma="" visao-geral-sobre-metodologia-agil/27944/="">.</http:> Accessed on 21 October 2016.

[1] Graduated in computer science, it acts as a public server on SUFRAMA, as Administrative Analyst-you.

[2] Graduated in computer science, it acts as a public server on SUFRAMA, as Administrative Analyst-you.

[3] Graduated in computer science, it acts as a public server on SUFRAMA, as Administrative Analyst-you.

[4] Graduated in business administration, acts as public servant on SUFRAMA, as administrator.

[5] Graduated in computer science, it acts as a public server on SUFRAMA, as Administrative Analyst-you.

[6] Graduated in computer science, it acts as a public server on SUFRAMA, as Administrative Analyst-you.

[7] Graduated in computer science, acts as server SUFRAMA, as Administrative Analyst-you.

[8] Graduated in economics, acts as public servant on SUFRAMA, as an economist.

[9] Graduated in computer science, it acts as a public server on SUFRAMA, as Administrative Analyst-you.

PUBLIQUE SEU ARTIGO CIENTÍFICO EM:
https://www.nucleodoconhecimento.com.br/enviar-artigo-cientifico-para-submissao