UNIVERSITY OF TECHNOLOGY SYDNEY

Individual Assignment

Name: Rami Shakour

Student number: 11886114

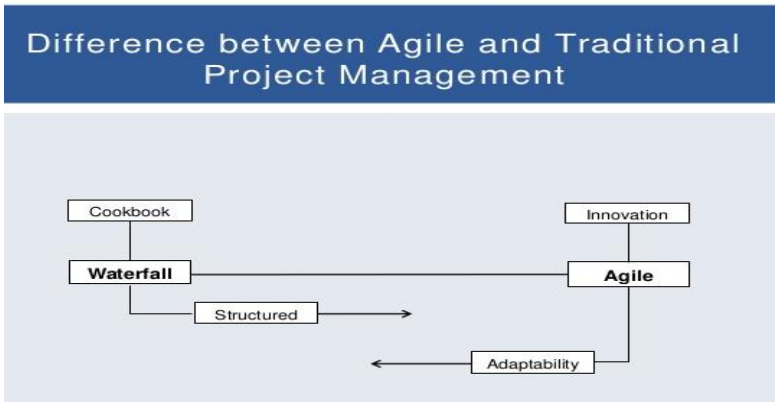
Agile Innovation in Project Management

Abstract

Today, we can generalize that the projects have grown in complexity and it is very common for the project managers struggling to bring their work efficiently. Recognizing these challenges, difficulties, changes and technological advances, project managers try to use standards that can best meet the goals of success and deliver them. Adopting a model for new product development project management is critical to their performance, particularly at the end of high-tech products. The application of project management in business gives better results, quality and reduction costs in the product development. The innovative products are embedded in dynamic business environments, characterized the difficulty in predicting the future, uncertainties and major challenges. However, the current assignment has presented the importance and methodologies of Agile Innovation in project management. The introductory part has given a brief overview and comparison between classical project management and agile project management. In the discussion part, agile methodology is discussed in detail along with the principles and values.

Introduction

For many years, the project management was developed on a solid platform capable of supporting different types of projects, in various environments. Organizations have embraced Classic Project Management that was proposed and implemented for improvements and adaptations, creating their own models and expanding the classic platform for project management. However, like any platform, Classic Project Management presents constraints and approaches to the threshold of its coverage, these restrictions become more visible and their performance, less effective. The development of classic platform continues, but sometimes it is more difficult than designing a new idea. In this context, Chin (2004, p. 1-3) proposed the creation of a new model that was not simply an expansion of Classic Project Management. Then the concept of Agile Project Management arises, as a new project management platform, which is applicable to volatile and challenging environment and subject to frequent changes in the prescriptive and standardized process that no longer fits (Chin, 2004 p 1-3; Highsmith, 2009, p 5). According to Highsmith (2009) and Chin (2004), Agile Project Management discards the anticipatory posture and is strongly based on the prior planning of actions and activities and seeks to develop the vision of the future and the ability exploration.



Discussion

The Agile Project Management can be seen as a management framework response to the growing pressure for constant innovation, fierce competition, the need to reduce development cycles and implementation of new products or services and the need to adapt to a very dynamic business environment. Highsmith (2009, p.16) defines the Agile Project Management as "[...] a set of values, principles and practices

that help the project team to deliver products or services value in a challenging environment." The values and principles describe why the Agile Project Management and practices describe how to accomplish it. Despite ratifying the need to deliver reliable products to customers within time constraints and costs, common to Classic Project Management, Highsmith (2009, p. 6) mentions that the Agile Project Management can be considered "more than an attitude than a process more environment that methodology."The main values of this new project management approach addresses both the need for creating and delivering agile products, adaptable and value, as the need for project development teams with the same characteristics (Highsmith, 2004). The four core values of Agile Project Management are:

- The responses to change are more important than following a plan;
- The delivery of products is above documentation delivery;
- Prioritization of customer collaboration over contract negotiation;
- Individuals and their interactions are more important than processes and tools.

Highsmith (2009) further stated that there are five main goals for a good exploration process, which ultimately form the basis for the Agile Project Management:

1. Continuous innovation: delivering products that meet customer requirements and add value to the business. The idea of innovation is associated with an organizational environment whose culture encourages self-management and self-discipline;
2. Adaptability of the product: the products developed must not only provide customer value in the present, but also be adaptable to the changing needs of the future;
3. Delivery times reduced: focus, precise targeting and technical capacity of the staff are essential to reduce product development times to maximize the use of market opportunities and a better return on investment;
4. Ability to adapt the process and the people: Training teams whose members are comfortable with change and not act as obstacles but

accepts challenges of a dynamic environment; establishing processes to respond quickly to business changes;

5. Reliable results: delivering customer value products, ensuring the operation, growth and increased company profitability.

Considering the first value, Highsmith (2009) states that the exploratory projects, targeted at Agile Project Management, are characterized by processes that emphasize a vision of the future and exploitation, rather than the detailed planning and their implementation. Chin (2004) mentioned that it is not just to absorb minor changes of scope, time or cost, but rather to give complete openness to change plans, requirements, technology and architecture as the project progresses. Highsmith (2009) underlies his view, to present the case of a company that erroneously refused to change the plan initially traced to a software development project, budgeted at approximately US $ 125 million, which it led to a large and costly disaster. To Highsmith (Ibid.) "[...] Plan the work and execute the plan" is not an appropriate slogan to a wide variety of projects, especially for new product development projects (or software) or those related to any type of technological innovation.

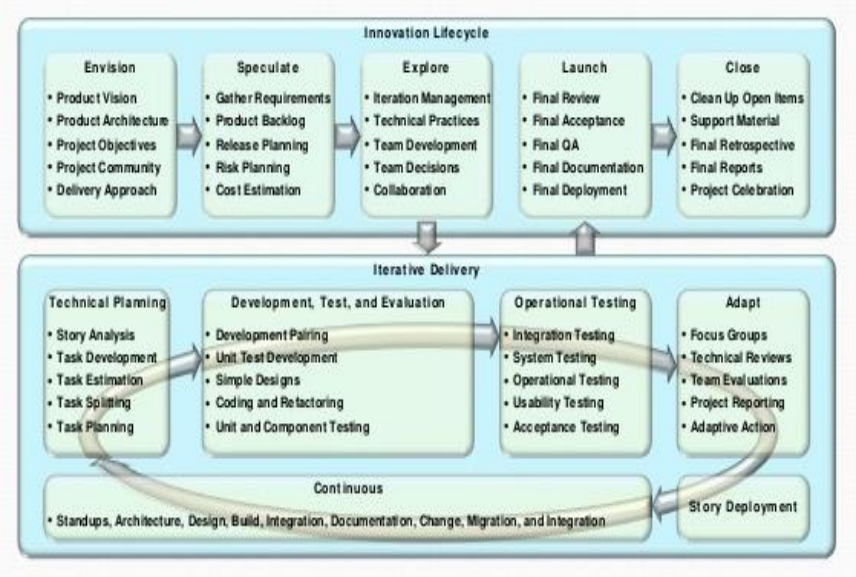
Addressing the second value by establishing a product delivery priority over the preparation of a comprehensive documentation, Highsmith (2004, p. 11-12) argues that this is not to be little the importance of documentation, but to assume that the results can only be evaluated by the project team and the client when something concrete is presented, or when there is a product (or software) functioning. Highsmith (Ibid.) supports the maintenance of a minimum and sufficient level of documentation to support the process of communication and collaboration, foster the transfer of knowledge, preserve the historical information, the basis for the improvement of products and processes and sometimes, meet legal requirements.

The third value, in turn, is assuming the establishment of a partnership between the client and the project team, in which each has specific roles and well defined responsibilities, being charged for it. This partnership should be marked by strong collaboration and not by contract disputes (Highsmith, 2009, p. 13). While acknowledging the existence of different project stakeholders, Highsmith (Ibid.) mentions that the client should "sovereign rule" and has the following definition of customer: "[...] an individual or group of individuals who use the products or services to generate value to the business.” Considering the final value, Highsmith (2009, p. 13-14) stated that the processes serve as guides or tracks and tools are used to

improve efficiency, but without people with technical and behavioral skills appropriate, any process or tool produces some results. The agile movement puts high value on the individual and recognizes his self-organizational skills, self-discipline and competence.

Focusing on the success from the viewpoint of Agile Project Management, Gido, & Clements (2014) mentioned that success is driven by people and their interactions rather than by processes and structures. Skills and individual skills, an interaction between technically qualified individuals or teams and the team's ability to learn and apply the acquired knowledge are crucial to the success or failure of a project and are closely linked to meeting customer expectations. Since people are usually driven by an internal set of values, to develop speed depends on the perfect alignment between the environment and this value of each individual system. This is why the Agile Project Management is strongly based on values and also the reason why it is often impossible the application of Agile Project Management for certain organizations or teams (such as with the adoption of the Development of Agile Methods Software).

In addition to the values, the Agile Project Management has a set of guiding principles that direct its implementation. According to Cervone (2011), the principle-centered leadership is one of the most important characteristics of high-performing teams. On the other hand, Highsmith (2003) sets out six principles, divided into two categories, which help project teams to determine how to proceed, that is, help them to define which practices are most appropriate, to generate other practices when necessary, or even, to evaluate some that may arise and implement them swiftly. Although each principle is useful if applied alone, the system formed by them creates an environment that encourages and produces more effective results (Highsmith, Ibid.).



The basic characteristics of the projects managed with agile methodologies are:

Uncertainty: The address indicates the strategic need to be covered (without elaborating), providing maximum freedom to the team.

Self-Organizing Teams: There are no specialized roles

Autonomy: Freedom for decision-making.

Self-Improvement: Periodically the product being developed is evaluated.

Self-Enrichment: Knowledge transfer.

Overlapping Development Phases: These phases do not exist as such, but tasks / activities are conducted according to the changing needs throughout the project. In fact, it is often not possible to make a detailed technical design before starting to develop and see some results. Moreover, the traditional stages made by different people is not conducive to teamwork and can generate more disadvantages than advantages (e.g. in a phase delay affects the whole project).

Subtle Control: Local control points to properly monitor without limiting the freedom and creativity of the team.

The investment in projects is at boom throughout the world. It is estimated that the demand for professionals in projects will increase 30% by 2020, representing an increase of 1.1 million new jobs per year. Companies with greater maturity in project management generate greater returns for shareholders. Against the tide of these factors, the last The Chaos Manifesto of 2013 showed that only 39% of projects achieve their full objectives and 74% of projects that do not reach the goals have term problems. Last PM Survey report, also of 2013, 61% of companies reported having

problems deadlines in its projects "always" or "most of the time."In recent years the agile project management is increasingly present in books, blogs, academic articles etc., making it a striking element in the discussions of professionals, as well as an alternative for companies to improve their results in projects. There are several alternative methodologies and agile approaches to project management in various industries, and when we are talking about IT, Scrum is the framework used in 54% of cases according to data from the Annual State of Agile Development Survey. Following the keynote of the continuing evolution of IT market several different studies and approaches for using Scrum on the market aimed at maximizing the positive results achieved by the method. It can be seen that many companies are looking for ways to minimize their losses on projects and, in many cases; the implementation of agile models of project management has been a bet on the market to solve this problem. Currently, the most popular agile methodology for project management is Scrum (Cohn, 2010). It comes as a counterpoint to PMBOK and PRINCE2, being used for both software development and for other product. Moreover, they also have specific methodologies for software developments are intended as alternatives to standards such as ISO / IEC 15504, ISO / IEC 12207 and CMMI. For instance:

Dynamic Systems Development Method (DSDM): Oldest Agile methodology and approaches more traditional methods, its implementation would even allow a level 2 CMMI maturity under.

Extreme Programming (XP): The most radical and popular agile methodology. XP focuses on the life cycle of software development.

Agile Modeling: Methodology for modeling and generating documentation that is aligned with the principles of agile development and can be used as a substitute for standard UML.

Feature Driven Development (FDD) software development methodology aimed at creating value for the customer.

While traditional software development methodologies such as CMMI or ISO 15504 have processes that cover all the needs of the information systems, the listed agile methodologies include complementary areas together. For this reason, it is possible to apply several of them together (e.g. Scrum, XP and Agile Modeling) (Cohn, 2010). The appearance and constant adoption of agile methods for managing projects have jeopardized the function and even the scope of the role of project

manager. "We do not need any project manager" is a common phrase heard often by software developers (Dingsøyr, Balijepally & Moe, 2012). The typical advantages of a good project manager is to educate them, when it comes to traditional projects undoubtedly an effective project manager adds a crucial added value throughout the project, achieving improved predictability, perception and probability of success. The nature of agile projects makes the cost and time usually stay fixed in a "time boxing" through each iteration. These methodologies can be used only in certain specific types of projects: software development projects, in which human resources teams are small, highly qualified, co-located and in constant contact with the customer. It is also assumed that the level of criticality is relatively low, so that there is a high tolerance for the progressive development of the results (Dingsøyr, Balijepally & Moe, 2012). Thus, for an organization and a team that are prepared to accept this approach can be very efficient use of these agile methodologies. Given these assumptions, it seems that a project manager, who is experienced and accustomed to managing large initiatives, would be ill-equipped to add value to the agile projects.

Architect of an Agile Team

Article was written by Avram (2014) focusing on the architect of an agile team. The writer has discusses the work of Tom Hollander, who is a solution architect at Microsoft Australia; he presented some of the suggestions for solution architect of an agile team. Following are the suggestions:

Just enough-Initial design of the Project: Only for very simple projects where there are a certain initial amount of design (01 to 02 weeks) is absolutely necessary according to the application type (web applications, intelligent systems, mobile applications or batch), which are the basic functionality requirements for applications short, medium and long term. The purpose of a design during early project is to decide: which technology to use (or ASP.NET etc), which the application type (2 or 3 layers, or service-oriented applications), access to bank data (using Stored Procedures, or some framework, LINQ). That is, the whole thing that can be enclosed for a reference in a short document.

Begin with a Vertical Slice: It refers to begin the project with a little part of functionality, an access login page, for instance, using several technologies selected in the previous phase, Project Home Design. This will show whether the chosen design solutions are correct and that all of them can work together, representing a pattern to

be followed by developers to implement new codes. So if the initial design decisions become inadequate, it is the right time to change.

Immediate response (Just-in-time) Design at Every Iteration: In the middle of the fourth week of iteration, the project manager, the product manager and project architect are required to talk about the requirements to compose the next iteration, in order to be sure that everyone agrees with these requirements, and also to define the priority of each requirement, and if everyone agrees. This discussion extends during the week of the current iteration. During the last week of the iteration, the architect goes through the necessary changes for the next iteration, taking design decisions necessary for the team to work on the requirements the subsequent week. If the changes are altogether different from last ones, the architect makes some prototypes, encodes something close to expectations, and writes some diagrams, putting everything in a background paper of up to 5 pages. This does not mean in detail the solutions to the developer, but to make the new requirements are in line with expectations.

Trust on your Team: This directly relates to the relationship between the architect and the developer. But the architect cannot exceed its role, not to make all the decisions alone, in order not to become boring for developers. The architect needs to be a mentor for the team at the same time, solving difficult problems that may create difficulties for the team. The architect has to be in contact with each developer every day, to know what they are up to, and facilitate them if they are facing any problem of coding. This is especially necessary when it comes to developers who do not like to ask for help and try to solve their own problems, and may even spend a whole week solving.

Write the Code: The architect has to understand the code in order to have a better idea of the effect of their decisions. It can also check, where and when a refactoring is required. When an architect also implements a code, you gain more credibility the team. According to Hollander that this is not dissolution of assignments, as the architect remains an architect and the fact is that it does not necessarily implements a code as well as a developer.

Involve in Everything: It is very good for the architect to be concerned in all the meetings, in relation to the project's proposed solution, code review, development and planning requirements etc. Because an individual can take a broader perspective and an obvious idea of what is going on and facilitate the product manager to stop

them in considering wrong decisions at an initial stage, informing them about the possible outcomes.

Impel a Quality Culture: A flourishing workforce that everyone wants to join is formed on a culture of quality: no one finds a poorly designed code, an important flaw in the project is not lost, and all team players are honest and open and look for the best result of the whole team. First, the architect has to create some rules in the establishment that would not transform over time, if a team member is not satisfied with them. An example of a rule is the choice to write unit tests, another is to have a code review before declaring it as ready, including codes implemented by the architect. Thus, if the code is not accepted by the reviewer, who can be anyone from the team, the code is finalized.

Identify When Changes Are Required: The architect should be extremely flexible and set to bring changes in design when it is required, it possibly will be that a rapid solution is not sufficient, or new requirements require a different approach.

Protect the Team from outside Interference: Generally this is the work of a project manager / Scrum Master; the architect can protect and get mingled with the team from external interferences that normally distract the focal point of the team and require more time to perform tasks.

Write Documents: In order to remember the comprehensive designs, for instance Data Model, documents is a good option for that. Decisions for iteration design should be discussed with other team members at the initial stage of the iterations. However, it is suggested that it should be saved in five pages document for the developers.

Conclusion

Hence, it can be concluded that the use of agile practices is underway in many contexts, regardless of the type of product and the degree of innovation, but it is important to recognize the value of different approaches, practices, tools and management techniques available and understand when to employ right combination for each project. Information technology is constantly changing. Customer's form the new requirements, manufacturers are offering new solutions used infrastructure requires renovation. In such circumstances, the importance of effective management of large changes or project management cannot be overestimated. However, some assume that a good project manager can be successful in any project, regardless of

whether it has practical experience on the area or not. Others believe that it cannot manage a project if he/she does not have the experience on the appropriate area.Therefore; agile methodologies have a diametrically opposite approach to predictive methodologies, providing a more suitable platform for certain projects such as software development approach. However, it is important not to assume the end and all that is bad for one side or another. Although, the use of agile methodologies is interesting in order to understand the tools and predictive techniques. The convergence between the two models can lead to efficient and effective management. The value of this method needs to be explored in the coming years and that has great potential to prove important aspects of the theory of agile management.

References

Abel Avram (2014), 10 Suggestions for the Architect of an Agile Team, Available at: http://www.infoq.com/news/2010/09/Tips-Architect-Agile-Team

Cervone, H. F. (2011). Understanding agile project management methods using Scrum. OCLC Systems & Services: International digital library perspectives, 27(1), 18-22.

Chaos Manifesto (2013) Think Big, Act Small, Available at: https://larlet.fr/static/david/stream/ChaosManifesto2013.pdf

Chin, G. (2004). Agile project management: how to succeed in the face of changing project requirements. AMACOM Div American Mgmt Assn.

Cohn, M. (2010). Succeeding with agile: software development using Scrum. Pearson Education.

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, 85(6), 1213-1221.

Gido, J., & Clements, J. (2014). Successful project management. Cengage Learning.

Highsmith, J. (2003). Agile project management: Principles and tools. Cutter consortium, 4, 1-37.

Highsmith, J. (2009). Agile project management: creating innovative products. Pearson Education.

Project Management Survey Report (2013), Strategies to capture business value