15356

Reflective Project Practice

Individual assignment

Reflective Practices in Agile development of the On-Demand Virtual Lab

Name: Phi Bang Nguyen
Student number: 97079610

Submission date: 25/06/2012
Word Count: 2705

Contents

Reflective Practices in Agile development of the On-Demand Virtual Lab

Abstract

The focus for this paper is to investigate **reflective** practices in an **Agile** software development project – The On-Demand Virtual Lab. It aims to understand how Agile development has been used in a **technically complex 'proof-of-concept'** project.

This paper uses a **systems thinking** approach to understand the components of this On-Demand Virtual Lab. Both using a hard systems approach to understand the technical issues, as well as a soft systems approach to understand the personal issues.

The investigation found that there were weaknesses in understanding the complexity and length of this project. There was a lack of support from management, as well as a lack of knowledge transfer.

Finally, the paper presents two reflective tools, known as **Agile-Jazz** and **Agile-ECG** that have been demonstrated to be beneficial for teams involved in complex projects. Agile-Jazz is an enhanced management structure, which brings stakeholders together to reflect and understand the problems, and seek solutions together. Agile-ECG classifies the reflection into Emotions, Cognition and Growth – providing a convention to reflection and aiding discussion within the team.

Introduction

Can you remember your best or worst day of high school? Can you still remember your emotions at the time? Have you thought about what you've learnt from that day? Or how it has changed you?

Maybe you can remember what you were doing on September 11, 2001?

The process of reflection gives us insights that we do not have at the time. Neuroscience explains how memories are strengthened during emotional stress (McGaugh 2003). A moment's reflection will persuade most people that events like these - events that were emotional when they occurred - provide many of their most vivid, most detailed, most compelling autobiographical memories (Reisberg, Hertel & ebrary Inc. 2004).

This paper looks at the benefits of adopting active, ongoing reflection during projects, in particular in agile software development. Agile development reflects a method that promotes "iterative and incremental development" (definition from Wikipedia, 2012). The "Agile Manifesto" includes a principle that loosely describes a reflective process:

"At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly." (Agile Manifesto 2012)

This paper will aim to enhance the agile methodology with tools and techniques that help the individual and the team describe their reflection, creating a baseline for communication and synthesis of ideas.

The project – On-Demand Virtual Lab

In 2007 a web-based project was started as a research investigation. Virtualisation and cloud computing was gaining interest and this project was to be a new virtual environment for customers. The concept involved creating an Online (On-Demand) Virtual laboratory, known as the On-Demand Virtual Lab or ODVLab.

The system was developed using incremental development methodologies, specifically "agile development".

The aim of the project was to create a configurable computer environment that was accessible to all users from anywhere in the world. The computer environment had the simple requirement of a web browser and the ability to install a small program to launch the Virtual computer.

The driver behind this project was that a typical corporate IT environment is rather restrictive. Users are often not allowed to install programs, let alone reconfigure the computer. Test environments are expensive and require network infrastructure, technical support as well as physical space. IT projects often have to satisfy a few key stakeholders (Wateridge 1998), which means that their needs must be satisfied even if they are not aligned with the corporate strategy.

The ODVLab system creates small 'projects' where users can create multiple machines that are pre-configured, connected to each other, and have unrestricted administrative control. Since they are virtual and isolated from the real network, there are no security concerns. Users can create, recreate and remove virtual computers as needed.

The ODVLab became an implemented production system 12 months later. Supported by the IT group it then received funding to ensure that resources were available for potential users.

Reflections on the ODVLab development.

Using a **systems thinking** approach to reflection, this project was reviewed to understand how it evolved through the process of development - from a research proof-of-concept to a fully-developed production system.

Using hard-systems thinking, the individual components of the development were reviewed. The **Anatomy tool** (Remington & Pollack 2007) breaks down parts of the greater system to show the roles of each part as well as the links between each part. Each component does not need to know what is 'inside' the adjacent components, it just needs to know what the functions are, and what information or interaction is required to be transmitted between each component.

Since the project started as a research project, it was not given the resources of a formal project. It was a proof-of-concept that eventually proved that it was Desirable, Achievable and Viable; a key component in project justification. (Great Britain. Office of Government Commerce. 2002),

This project was technically complex, using techniques that were unknown or untried and for which there were no precedents (Remington & Pollack 2007). In project management the project must reach an end or it is deemed a failure yet with many research and development projects, there is no set path or firm milestones and no rules to heed.

The application of agile methodologies allowed the initial development to complete components of the task without needing to understand the entire concept. It allowed freedom to change scope without affecting work that had been done.

A soft-systems approach was used to understand the complexity of the project and the interactions between the people. The SECI model (Nonaka & Takeuchi 1995) was used to demonstrate how tacit knowledge was converted to explicit knowledge through Socialisation, Externalisation, Combination, Internalisation.

As a research project in a small team, it suffered from a lack of greater knowledge and resources to complete the project to the desired level of finish and quality. In this development phase, some ideas would lead to dead ends or new methods would be found that would supersede old ones, therefore tacit and implicit knowledge was not transferred to explicit knowledge, and hence successes during development were not often recorded. Knowledge sharing and learning during the development was limited to development notes and occasional team meetings.

Technological hurdles in the development would eventually delay the project, extending the already loose deadline. With many development projects the timeframe to completion is often unknown, or underestimated. As a humorous quote attributed to Tom Cargill (Bell Labs) illustrates; "*The first 90 percent of the code accounts for the first 90 percent of the development time. The remaining 10 percent of the code accounts for the other 90 percent of the development time*".

Application of reflection in agile development

How can agile projects be improved by incorporating continual reflection? The postliminary reflection of the ODVLab project showed that there could be improvements in reflective learning and knowledge transfer.

Boyd and Fales (1983) define reflection as “the process of creating and clarifying the meaning of experiences (past or present) in terms of self”, it is a process that is not just done as a past memory, but during projects as well.

The reflection that occurs after the action is described by Schön (1983) as ‘reflection on-action’, and reflection afterwards as ‘reflection in-action’, professionals who are able to ‘think what they are doing while they are doing it’.

Other authors have described their experiences with adding reflection to agile development. Lamoreux (2005) wrote that her team tended to discount regular reflection meetings and that engineers wanted to avoid anything that might be seen as "touchyfeely." Whereas Talby et al (2006) implemented reflection as part of a “Iteration Summary Meeting” and found it to be highly effective and increased team satisfaction.

So how does a team look to implement ‘Reflection in-action’? When is it seen as a burden rather than a valuable addition to the team? Looking at the team’s motivation and personality may give insight into the drivers for reflection within a project.

Motivation

In psychology theory, Maslow (1954) describes the needs of an individual in his “Hierarchy of needs”, where a person looks to fulfil their basic needs before achieving more high level needs. We can interpret this as a need for the team member to be able to fulfil innate needs before being able to seek higher goals such as self-actualisation. Team members need to feel secure to explore thought exercises whilst still trying to meet deadlines.

Sigmund Freud (1923) defines the human psyche in three parts; the Id, child-like pleasure without consequences; the Ego, censorship and reality testing; and the Super-Ego, to identify with conscious thought. The Super-Ego inhibits the Id's impulses while persuading the Ego to substitute moralistic goals for realistic ones and to strive for perfection (Ryckman 2000). It is this part of our psyche that looks for further reflection and understanding.

Herzberg’s Motivation-Hygiene theory (1987) may be a more intuitive description of job satisfaction. Herzberg demonstrates how job satisfaction is dictated by positive events and negative events that have an influence on satisfaction or dissatisfaction. For example, poor company policy and administration can lead to more dissatisfaction than a low salary. And achievement and recognition plays more of a role in satisfaction than status.

Understanding these motivations may help promote ‘Reflection in-action’.

Personality

Some people are more inclined to use reflection in their learning and development. Understanding personality types may influence how they use reflection, or their willingness. The Myer-Briggs Type Indicators describe four dichotomies; Extraversion vs Introversion; Sensing vs Intuition; Thinking vs Feeling; Judging vs Perception.

People who prefer Extraversion draw energy from action: they tend to act, then reflect, then act further. If they are inactive, their motivation tends to decline. To rebuild their energy, extraverts need breaks from time spent in reflection. Conversely, those who prefer introversion expend energy through action: they prefer to reflect, then act, then reflect again. To rebuild their energy, introverts need quiet time alone, away from activity. (Wikipedia 2012)

Reflective Tools for Agile projects - AGILE-JAZZ

The Agile methodology gives developers and software engineers a framework to work towards. However a Project manager may not have the technical understanding to follow these methodologies and may be compelled to take a higher level approach to manage the complexities from above and incorporate regular reflection.

The Agile-Jazz tool is derived from Remington and Pollack’s (2007) Jazz management tool. The Jazz tool is a management approach for technically complex projects that are typical of research and development projects. The project team is given the freedom to explore their problem, using reflection to understand the problems, and brainstorm solutions. The analogy is a group of Jazz musicians playing together without needing structure to create music. But upon closer inspection we understand that they need to communicate through beats, through phrasing and timing to achieve something harmonious rather than just noise.

How to use the Agile-Jazz tool

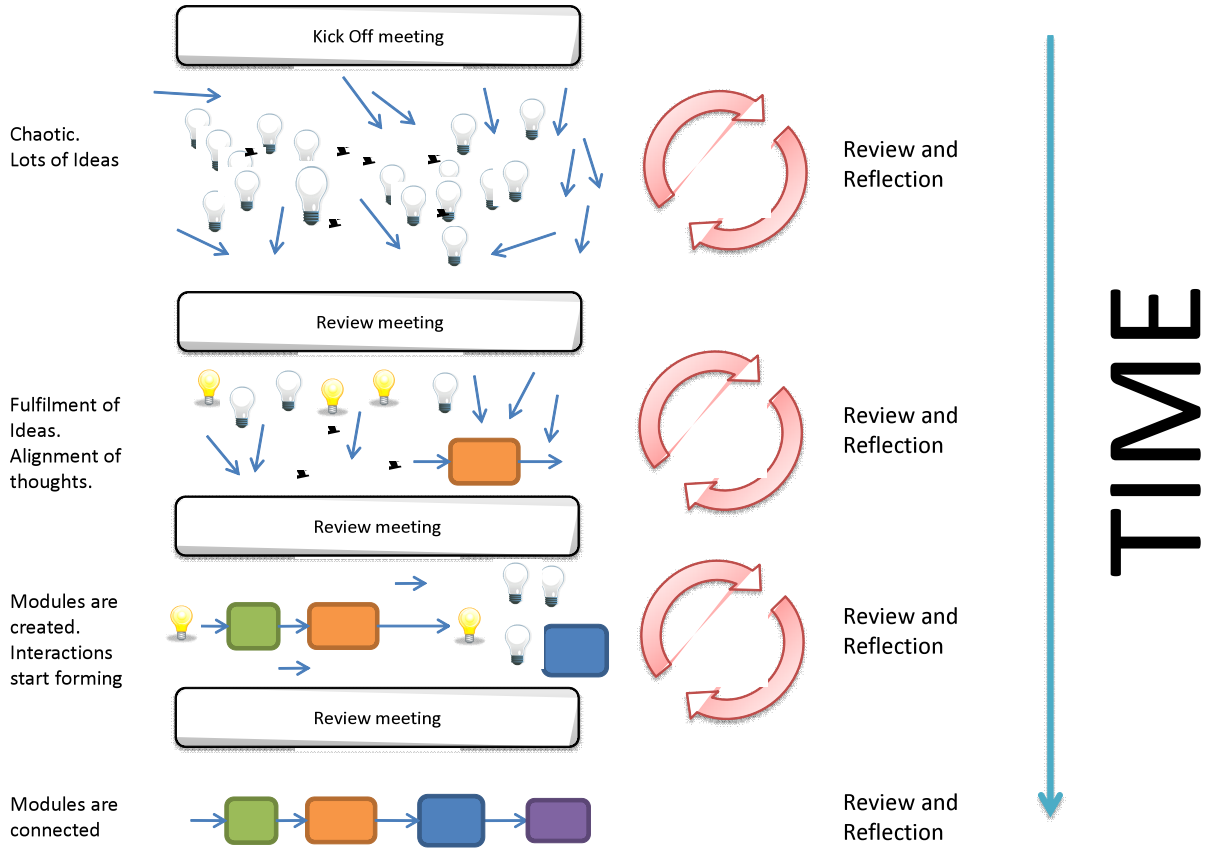
After the initial kick-off meeting, team members are brought together to discuss their ideas, goals, reflections and understanding. These meetings and informal communication can be tied in with the requirement to have a “Story”, “Package”, or “Module” complete for the Agile methodologies.

At each stage, goals are aligned, methods and reflections are discussed and irrelevant parts are removed.

The Review meetings are part of the Jazz tool which allows the entire project team to discuss their direction and realign modules that have been developed as part of the Agile development cycle.

Through each repetition of the Review meeting, the goals are aligned and the irrelevant parts are removed.

The diagram below shows how Remington and Pollack's Jazz tool can be incorporated with modern Software Development paradigms. This merged set of tools has been dubbed the **Agile-Jazz tool.**



Reflective Tools for Agile projects - AGILE-ECG

To aid understanding during the Agile-Jazz reflection, a system of classifying each person's reflection has been derived. The Agile-ECG tool seeks to describe Emotions, Cognition and Growth during each phase of the Agile-Jazz review (Using the ECG method, described by Mark Neasby, The Australian Centre for Value Management Pty Ltd)

The analogy here is the Electrocardiogram that monitors a patient's heartbeat – it is a heartbeat that is continuous throughout the project.

The ECG model looks at answering these questions to give greater insight into one's reflection:

- Emotions.
  What did you feel? Why did you feel this way? Would you want to feel this again? McGaugh's (2003) work shows how strong emotions enforce memories. Traumatic experiences like car accidents are often imprinted into your psyche, often with a sense of fear or anxiety.
- Cognition.
  What were you thinking about? What did you understand? In the past century, influential thinkers such as Dewey, Piaget, and Vygotsky have argued that knowledge and control of one's own cognitive system play a key role in cognitive development. For example, Piaget (1976) argued that being aware of and reflecting on one's cognition is one of the defining characteristics of the most advanced stages of cognitive development." (White & Shimoda 1999)
- Growth.
  What have you realised? What have you learnt? What will you change? Understanding what occurred is a powerful way of avoiding pitfalls in the future.

The ECG method has been used to demonstrate an understanding of reflection in various projects. It uses intuitive prompts to open common discussion. For example, one team member once said "I felt confident and excited with this part (of the project)", which prompted another team member to say, "Oh, I actually felt confused and nervous". Two different emotions that may have led to different interpretations of the meeting.

The other benefit is that it assists with knowledge transfer. Opening a discussion based on personal experience allows the transfer of tacit knowledge into explicit knowledge. Nonaka's SECI model demonstrates how socialisation and externalisation – in this case a reflection discussion – promotes the knowledge transfer by emphasising and articulating.

Retrospection of the ODVLab project

In hindsight, the ODVLab project may have achieved more by applying Agile-Jazz and Agile-ECG tools. The Agile-Jazz tool would have allowed management and team members to discuss their understandings and reflections. Creating a "Ba" (Nonaka & Nishiguchi 2001), a "Fertile ground" or "a shared context in cognition and action" where users have a basis for sharing of knowledge and experience.

The ODVLab would have benefited from the extra management/stakeholder input that the Agile-Jazz tool emphasises. Agile development on its own has close affiliation with the client. However Agile-Jazz brings the clients into the creation of ideas. Further enhancing the 'agile' behaviour to adapt to change, have flexible scope, and achieve modules of development.

The Agile-Jazz tool also helps avoid the feeling that reflection was a waste of time and too "touchy-feely" (Lamoreux 2005) by bringing in management as well as stakeholders.

The Agile-ECG model would have given the team extra understanding of each other's emotions, understanding and growth. At the very least, it would aid communication and bring the team closer together. And at best it would improve the development by understanding how others have avoided pitfalls, how other people understand each other and work together, and how they can help each other grow and learn together.

Conclusion

This paper followed a research proof-of-concept project that was developed using Agile methods.

It reflected on the approach taken and lessons learnt during the period of that project. It found that there were weaknesses in understanding the technological complexity. The project lacked the ability to define scope and time, and ultimately quality. The team also had little knowledge transfer and direction.

The paper proposes two tools which extend the Agile methodology to alleviate some of the problems that this project faced.

Agile-Jazz is a management tool that brings in stakeholders together as a reflective exercise to understand and share problems and solutions. It further emphasises the close relationship between client and developer that is vital in Agile development. Changes in scope are discussed from the start and the project remains 'agile'.

Agile-ECG is a classifying tool that extends Agile-Jazz so that the developers and the clients have a common language to reflect on the project. This is done as "Reflection in-action".

References

Boyd, E.M. & Fales, A.W. 1983, 'Reflective Learning Key to Learning from Experience', *Journal of Humanistic Psychology*, vol. 23, no. 2, pp. 99-117.

Great Britain. Office of Government Commerce. 2002, *Managing successful projects with PRINCE2*, 3rd edn, Stationery Office, London.

Herzberg, F. 2008, *One more time : how do you motivate employees?*, Harvard Business Press, Boston, Mass.

Lamoreux, M. 2005, 'Improving Agile Team Learning by Improving Team Reflections', paper presented to the *Agile Development Conference*.

McGaugh, J.L. 2003, *Memory and emotion : the making of lasting memories*, Weidenfeld & Nicolson, London.

Nonaka, I.o. & Nishiguchi, T. 2001, *Knowledge emergence : social, technical, and evolutionary dimensions of knowledge creation*, Oxford University Press, New York.

Nonaka, I.o. & Takeuchi, H. 1995, *The knowledge-creating company : how Japanese companies create the dynamics of innovation*, Oxford University Press, New York.

Reisberg, D., Hertel, P. & ebrary Inc. 2004, *Memory and emotion*, Oxford University Press, <http://site.ebrary.com/lib/utslibrary/Doc?id=10085293>.

Remington, K. & Pollack, J. 2007, *Tools for Complex Projects*, Gower Publishing Ltd, Hampshire, England.

Ryckman, R.M. 2000, *Theories of personality*, 7th edn, Wadsworth Pub. Co., Australia ; Belmont, CA.

Talby, D., Hazzan, O., Dubinsky, Y. & Keren, A. 2006, 'Reflections on Reflection in Agile Software Development', paper presented to the *AGILE '06 Proceedings of the conference on AGILE 2006*.

Wateridge, J. 1998, 'How can IS/IT projects be measured for success? ', *International Journal of Project Management*, vol. Volume 16, no. Issue 1, pp. 58 - 63.

Wikipedia 2012, *Myers-Briggs Type Indicator*, 2012, <http://en.wikipedia.org/wiki/Myers-Briggs_Type_Indicator>.