

Project Management in Practice: Views from the Trenches

Alan Sixsmith, FEIT, University of Technology, Sydney, Australia,
alan.sixsmith@uts.edu.au

Grant Mooney, FEIT, University of Technology, Sydney, Australia,
grant.mooney@uts.edu.au

Christine Freeburn, FEIT, University of Technology, Sydney, Australia,
christine.freeburn@uts.edu.au

Abstract

Software development has endured radical change with the introduction of agile methods for creating software solutions. This change has prompted new considerations of how software creation should be managed. While agile methods have changed software development processes, it would be premature to assume that has also induced modification in higher-level project management processes. Software development lifecycles (SDLC) and project management lifecycles (PLC), while associated, are not the same thing and it is still unclear to what degree the overarching project management tasks, tools or techniques must change or adapt to meet the needs of undertaking successful agile projects.

This exploratory pilot study investigated agile methods used to manage software projects and was conducted via an online survey and restricted to a specific sample audience with significant project experience and with background in both traditional and agile development methods. The results indicate that traditional project management phases and techniques are adapted to fit with agile. However, as the discipline evolves the potential exists for a pure agile project management framework to surface - one that can be applied to better suit the needs of the management of agile projects as well as projects beyond the realm of software development.

Keywords:

Agile Practices, Traditional Methods, Project Life Cycle, IT projects

Introduction

Agile software development methods have been said to more consistently achieve on time, on budget and high quality project deliverables compared to those forthcoming from more traditional approaches to software development (The Standish Group, 2012). There are different reasons often put forward by those supporting this view: continuous stakeholder feedback; ability to more easily manage changes to requirements; emphasis on working software as the major metric of success...and so on. In order to accommodate a shift to these agile frameworks, long-time project managers have had to modify skill-sets built up from their previous experience with more traditional linear models.

This exploratory pilot study aims to better understand the key challenges posed to traditional project managers when they find themselves having to undertake projects using agile methods of development. What are the *real* similarities between traditional project management methods and ones used in an agile environment? What are differences? To what degree are skills transportable and what has to be learned anew? Are agile methods of project management a true innovation compared to what has gone before or is it simply a natural restatement of traditional project management knowledge with a different emphasis?

The study draws upon the views of those familiar with both project styles. It seeks to provide insights to the questions noted above as revealed by those down in the trenches and holding

responsibility for successfully leading teams through changing organisational contexts using an evolving discipline – namely the project managers themselves.

The objectives of this study are:

- To undertake a pilot study of project management within the context of agile development across various organisations and industry sectors
- To draw initial conclusions as to whether or not agile project management is distinctly different from traditional project management

Its significance lies in the importance to organisations of being able to not only successfully and quickly obtain working software from project outcomes, but to do so and with a project force finding itself increasingly in transition from traditional well-delineated management methods to more open and flexible contemporary approaches – but where the same degree of project understanding and depth of mission experience has not yet been acquired.

Literature Review

Traditional project management (TPM) as described by the Project Management Institute (PMI) is achieved by the application and integration of five broad process groups, containing specific methods to accomplish each phase. These five process groups *initiating, planning, executing, controlling and closing* provide a frame of reference for all project managers regardless of domain. Each stage of the project life cycle is completed before the next stage can begin, budgeting is simple and straightforward based upon derivatives of early planning and each step is bound by timeframes that produce deliverables specific to the activities undertaken for the customer.

Importantly, the PMI argues that the project life cycle is independent of, and does not dictate, the type of methodology chosen to be used claiming, “a life cycle can be documented within a methodology” (PMI Inc. 2013 p. 38). Hence the TPM life cycle is understood to be broad, and thus should be used in conjunction with the methodology chosen, as each project will always be different and require a unique way to achieve its outcomes.

Software development has historically been nested within this framework as a very linear process – using models such as the waterfall model, spiral model and the v-model to name a few. Internally, these processes rely heavily on planning, documentation, and rigorous requirements analysis in the very early stages and later changes to specifications can prove to be huge burdens as cost increases exponentially as you progress through the span of the project. The requirements specified at the outset and agreed on by the customer thus act as the contractual agreement for work to be performed.

The most common traditional practice for software development is still the waterfall model which has been widely used but is not without drawbacks. Each step is heavily dependent on documentation, strictly relies on completion of a previous phase before proceeding to the next and allows little room for later modifications.

In contrast, the introduction and evolution of agile development has arguably been seen to address the challenges of traditional software development - such as market demand for creation flexibility - whilst maintaining an on time and on budget product that is of a high quality. Centred around twelve key principles outlined in the “Manifesto for Agile Software Development” (Agile Alliance, 2013) Agile software development emphasises the importance of a continuous stream of deliverables, or iterations, with close interaction and strong relationships with stakeholders. The focus is not so much on defining and planning in the early stages of the project but more toward customer interaction, trust and communication between team members melded with the flexibility to easily change project requirements. Specifically, the philosophy of agile development prioritises:

- Individual and processes over processes and tools
- Working software over comprehensive documentation

- Customer collaboration over contract negotiation
- Responding to change over following a plan

Furthermore, agile approaches to software development are generally classified as iterative/adaptive, as opposed to predictive, a method often ascribed to as waterfall. The PMI (2013, p. 38) notes that “in an adaptive life cycle, the product is developed over multiple iterations and detailed scope is defined for each iteration only as the iteration begins”. So, between predictive and adaptive/iterative we apparently have two fundamentally different lifecycles for orchestrating software development. But are they different *project management* approaches?

The PMI poses that the processes within a generic project life cycle will merely be repeated in an iterative or adaptive model. Marchewka (2012) supports this stating that the well-trod steps of the traditional project life cycle can also be used with an agile environment but with the execution phase following its own (i.e. agile) methodology, repeating itself until execution of the project is complete and the project moves to its final phases.

Others are happy to agree this. Griffiths ” (2004 p.1) suggests that the internal processes for running an agile project are fundamentally different than what has gone before as “agile methods goes beyond development team activities and changes the way sponsors, users, and other stakeholders are engaged”. Williams (2012) also supports this claim with his survey reviewing the traits project participants perceive to be “essential” for a project to be classed as agile. Processes ranked highest include:

- Continuous integration – rather than integration as a final step
- Short iterations – as opposed to one development stage
- “Done” criteria – as opposed to phases or milestones
- Automated tests run with each build
- Automated unit testing
- Iteration reviews/demos – emphasis on client interaction
- Embracing changing requirements – rather than full requirements at project onset

These steps - identified as crucial to any agile-style project - are not emphasised, considered or even encouraged with traditional project management.

In addition, employing traditional project management techniques within an agile environment is described by Boehm (2002 p. 64) as bringing a “crushing weight of corporate bureaucracy” to what should otherwise be a relatively streamlined process. In a context of rapid change and evolving technology the diligence of traditional project phasing would prompt extensive planning, documentation and task detailing whereas the regard of agile is for the importance of the customer and inter-team relationships over any amount of documentation.

Augustine, et al. (2005) describe a clear delineation between traditional and agile project management methods and discuss how historical predictive methods are too rigid for the agile environment. They describe a management framework for an agile project with six key practices:

1. Organic teams of from seven to nine members
2. Guiding vision
3. Simple rules
4. Free and open access to information
5. Light touch management style
6. Adaptive leadership

They further define an agile manager as one who “understands the effects of the mutual interactions among a project’s various parts and steers them in the direction of continuous learning and adaptation” (Augustine et al. 2005 p. 87). As a result of their study they prescribe what they called

an agile focussed “leadership-collaboration” project management model as apart from traditional project management frameworks.

Still other authors argue that it is possible to hybridise agile approaches by drawing upon conventional foundations. For example, Suman and Uikey (2012) propose an agile project management framework based upon identifying common ground with more traditional project processes. In another case, Baird and Riggins (2012) conducted a study at North Dakota State University to trialling a hybrid project management methodology that used both planning (traditional) and sprinting (agile). They encouraged teams to use sprints and perform traditional methods, such as planning and scheduling, within each sprint. The fusion of both traditional and agile project management methods showed some merit within the case study. The researchers concluded that TPM should be used as a catalyst and framework to get the project underway and to close it down but with selected departures from it into agile management practices during the actual running of the lifecycle. Similarly, Santos, et al. (2011 p. 700) support that agile project management can “incorporate distinguished principles from traditional approaches” into management of software development. However, their associated study into agile projects encompassing 109 respondents around the world also confirmed that users of agile methodologies do, in fact, employ steps and processes not accommodated by traditional project management (e.g. product backlogs, iteration planning, retrospectives, user stories, client partnering).

These cases suggest that while agile and traditional styles might share some overlap and common ancestry – and benefits - they are sufficiently different to warrant very careful reflection if hybridisation is to be seriously considered as a project management option.

Researchers elsewhere look at the topic from yet other angles. Xukan and Zhang Bao-feng (2012) formed a system analysis model based on the idea that requirements are prone to change throughout the lifecycle of a project. As one of their outcomes the authors state, “traditional methods such as waterfall model, a spiral model, a prototype model and others more and more aren’t able to adapt to rapidly changing requirements and market conditions” (2012 p. 986). Alternatively Sliger (2008), in looking at project management from the view of stakeholders, comments on the differences in involvement and regard according to the tenets of the two project styles. He notes that with TPM, stakeholders have a heavy contribution during project initiation but once agreement has been reached are mostly no longer required until closing and evaluation – in agile, stakeholder involvement is welcomed and facilitated throughout the lifecycle via continuous feedback loops. The area of stakeholder participation (and correspondingly, project change and risk) is philosophically incongruent between the two frameworks, requiring very different organisational approaches.

So, there appears quite a bit of variance over whether traditional and agile project management can and/or should be regarded as distinctly and philosophically incompatible project management approaches, able to selectively co-exist as software development lifecycles under standard project framework or whether some happy medium of hybrid is possible. These are interesting questions certainly worth further research.

Methodology

This exploratory pilot study research endeavours to investigate project management processes and techniques used throughout the lifecycle of an agile project. It further attempts to find the problem areas of management when employing agile processes in projects.

An online questionnaire was chosen as the data collection tool to allow ease of access for participants. Several questioning styles were used in the online questionnaire including free-form short answer, scale, multiple choice and ranking questions. The questions were categorized into areas including demographics (but industry and experience related) and contextual understanding of agile development and agile processes.

After the initial survey development, it was then sent out to external parties that would not be responding to the survey, but have sound knowledge in the topic area, for feedback and recommendations for changes to the survey. This testing was undertaken by three people (2

academics with extensive project management experience and 1 practising project manager). All feedback was reviewed and appropriate changes applied to refine and enhance questions.

Being a pilot study, a small sample of individuals with quite specific professional roles was targeted to ensure the results were both representative of our target population as well as adding precision. The web-based survey was distributed to several practitioner organisations and via a LinkedIn group within our University’s Alumni community and left open for 4 weeks. The survey gained 32 respondents with experience in both traditional and agile project management. All responses were recorded and stored within the web-based online survey tool which also had the ability to calculate preliminary statistics and keep track of number of survey participants, response and survey completion rates.

The analysis of our results used both qualitative and quantitative techniques to account for the different types of questions that were made available in the survey. Two qualitative techniques, namely hermeneutic interpretation and thematic analysis, were employed for the free form questions responses. Butler (1998) posits hermeneutics as the interpretation of meaning while Myers (2006) notes hermeneutics aims to understand the text as a whole but also view and interpret the text in its various parts. By understanding the text researchers can then identify common themes and consolidate these themes through the use of thematic analysis (Morse & Richards 2002).

Results & Discussion

A preliminary set of questions aimed to keep participants anonymous whilst still being able to understand industry experience and experience with agile practices. All participants had familiarity IT projects as shown in Table 1. Most had between 5 to 20 years experience with 50% having 10 to 20 years of technology project background.

Table 1: IT Experience of Respondents

IT Experience	No. of Respondents
1 to 5 years	3
5 to 10 years	9
10 to 20 years	16
20 years or more	4

Two other experience related measures were captured. Firstly, Table 2 shows backgrounds of participants in regard to either management or technical as a primary orientation. There was a relatively uniform spread amongst the participants: 35% stating their background was traditional project management; 28% for each of agile project management and agile software development. 20 out of 32 indicated they were currently in a project management role some others combined leadership with hands-on development as circumstances dictated.

Table 2: Background of Respondents

Primary Area	No. of Respondents
Technical – Traditional Developer	3
Technical – Agile Developer	9
Management – Traditional	9
Management – Agile	11

Secondly, table 3 shows respondent experience with agile practices. 25 of 32 indicated their experience to be more than 1 year - while 8 indicated their agile experience to be more than 5 years suggesting they were very early participants in the trend.

Table 3: Years of Experience Applying Agile Practices

Experience in Applying Agile Practices	No. of Respondents
Less than 1 year	7
1 to 5 years	17
5 to 10 years	8

These demographics showed experienced IT professionals who have both traditional and agile methods experience making this a good exploratory sample for study.

Agile Attitudes

The survey aimed to understand the agile experiences of the participants within both their past and current roles and hence to gain a deeper understanding of participants attitudes towards managing agile projects.

When participants were asked if they practice agile or traditional *development* techniques 50% confirmed they practice agile when producing software products, just under 50% saying they use a combination of both methods and 1 respondent using traditional methods only. Overall, this is an interesting but expected result, suggesting that selective use of both options (or elements of them) at is alive and well at the systems development lifecycle (SDLC) level.

In determining the type of *management* paradigm used for these software projects 50% of respondents believed they did not employ traditional project management techniques to govern their projects but used (their view of) agile project management. Alternatively, a quarter of respondents (26%) believed they used traditional project management techniques. The remaining 24% indicated that they either did not know, that the project management lifecycle (PLC) approach was unclassifiable or that it was not relevant (i.e. not differentiated from the SDLC). The top three agile development methods employed by respondents were the commonly known Scrum, Test Driven Development (TDD) and Lean Software Development.

An interesting observation from survey participants' was the belief of adaptation of traditional project management (TPM) to fit into the agile process. This is a key area of discussion in regards to a definition of agile project management. It is interesting that project managers in this field hold the view that the management of agile projects is an adaptation of TPM. It could be expected that TPM would form the foundation for project management ventures but to what extent have traditional methods been able to facilitate agile development? 24% of respondents indicated that they never or rarely adapted traditional management techniques to better suit the agile process. Intriguingly, this result appears to be contradictory to the result in the former paragraph where 50% of respondents had stated that they use an agile project management technique. The majority of respondents (60%) appear neutral in this area, stating that they "sometimes" adapt TPM processes for the purposes of agile development, while the remainder believed they often adapt TPM to fit agile projects.

The common themes identified from the responses to free form questions focused on the ability of agile processes to provide flexibility, faster delivery of working software to the customer and hence to market, team collaboration and the importance of customer and stakeholder relationship and continuous feedback. In particular one respondent surmised the difference in that agile is "iterative, progressive development where requirements are not clear upfront and modelling/prototyping is required to provide the business with more clarity on what they want...with agile scope is variable, but time and cost is fixed [whereas] the traditional model is more staged, each phase must be completed before moving to the next. Requirements must be documented before development is to commence, scope and requirements must be clear." At the other end of the scale however, 2 respondents for this question said that there were no differences between the management of an agile or a traditional project.

Independent of which type of methodology (TPM or APM) is used, respondents were asked to nominate their top 5 most important elements of a project. Through our literature review we selected the most common 16 themes that were of importance and have listed these as options in the question – of which each respondent selected without ranking their top 5 from the list. Interestingly, but expectedly, the most commonly selected elements were based upon different types of communication; that is between teams, within teams, with stakeholders, and with customers. The top 5 common elements are listed below:

1. Communication between team members
2. Active participation of stakeholders
3. Communication with the customer
4. Efficient change management processes
5. Ability to adapt to changing user requirements in later stages of the project

At the bottom of the list of the most important elements of software development projects, were items themed toward documentation, scheduling and delivery of software products within the budget, as listed below:

1. Staff training
2. Scheduling and strict time management
3. Delivery of software product on budget
4. Documentation

In stark contrast with TPM, these elements are known as some of the most important factors of a traditional project, and form the basis for planning and scheduling each step of the traditional product development lifecycle. The question was not asked in the context of agile development or traditional development, however the responses received, gave an insight into the potential changes needed for the management of a project.

Understanding Agile Processes

The following section was aimed at confirming the mindsets of participants in relation to our literature on agile and traditional project management and development methods. In particular, we focused on the values and principles of agile software development sourced from the Manifesto for Agile Software Development (Agile Alliance, 2013) and the beliefs of project managers in practice – whether or not these ideals are aligned with what happens in reality, and if hence they are achievable ideas.

The agile manifesto details a number of core values which have been published in the belief that, although there is merit in the traditional method they have found that there is more value in the way agile development can approach the same problem. When participants were asked what they valued more highly than the other, it was unanimous that agile values were more highly ranked than traditional paradigms as shown in Figure 1 below.

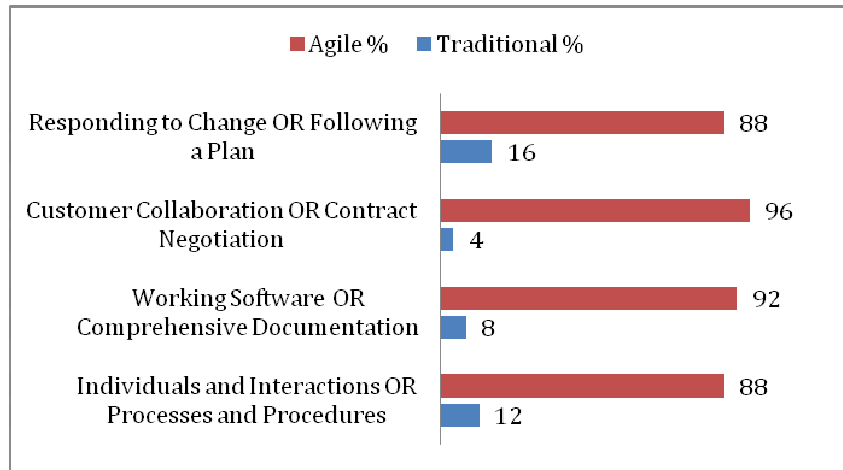


Figure 1: Agile Values vs Traditional Paradigm of respondents

Complementary to the four values conveyed by the Agile Alliance are also the twelve “Principles behind the Agile Manifesto” to describe precisely what it means to be ‘agile’. Study participants were given these principles in order to discover their understanding of what agile management means and what they believe are the most significant and representative agile principles and selected their top 5 from the twelve principles provided.

Table 4: Top Agile Principles as selected by study respondents

Top Agile Principles Identified		% of participants
1	Our highest priority is to satisfy the customer through early and continuous delivery of valuable software	87
2	Business people and developers must work together daily throughout the project	65
3	Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage	52
4	Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale	52
5	Working software is the primary measure of progress	52

In a follow up question to this, 96% of our respondents said they would prefer to manage a project by being flexible throughout the lifecycle of a project, as opposed to adhering to a strict plan. This is particularly interesting given that flexibility throughout the entire project lifecycle is not entirely possible when utilising traditional project management, however it is one of the main factors that realised the need for the agile paradigm.

When asked if and how developing with agile benefited customers, 25% of respondents stated that both traditional and agile methods have their merits, and the benefit of the customers would be dependent not on the method but on the *type* of project. One response stated that “where people are capable enough it is best to allow them to do their own thing in a supportive and collaborative environment...no one methodology will be the solution to everything”. However, the most dominant themes to come out of the benefits the respondents felt agile had toward customers were that the customers had the ability to learn and the flexibility to change requirements during development, instead of having to over specify the project during the initial defining phases. Further to this, another quarter of our respondents believed that through agile processes a higher quality product would be delivered to the customer much faster.

In providing a response to the way in which agile development has changed the way projects are managed a spectrum of responses was generated. The most dominant theme identified was being able to 'manage continuous customer feedback throughout the lifecycle of the project'. Two other clear themes to emerge were 'management of projects are now less formal and burdensome' and 'teams have become self-managed entities'. One respondent aptly recognised that "...instead of developers working off a specification, there is continuous feedback; change in priorities and decision making occurs as late as possible". This would infer that change management processes would need to change for an agile project and allow for up-front specifications to be changed or added to at later stages of the project without a negative impact on the end deliverable, time and monetary budget.

Regarding what participants found to be the easiest and most difficult phase of a project to manage responses were almost unanimous. The easiest phases of a project to manage were the software development iterations and daily rituals (i.e. 'scrums'). This could be expected as such daily iterations have clear processes in place in to achieve phase goals. The most difficult area to manage entailed the higher level project management phasing and structuring for agile undertakings, particularly project initiation and overall governance strategy. These phases overlap common ground with TPM and in particular PMBOK. Since the agile process provides limited documentation regarding commencement, managers misunderstand the extent of planning required before beginning development and/or prototyping stages. One respondent noted that strategy was most difficult because "strategy...is heavily tied to the often amorphous company strategy, budgeting, resource allocation, competitor analysis and so on...even the most Agile team has to confront often a very un-agile business or decision making process". This area of linking project benefit to measureable organisation value was also identified as most time consuming and least understood part of an agile project lifecycle.

For issues that re-occur during managing agile projects, the main ones highlighted by respondents were 'commitment to agile by the business', 'change management', and 'co-located team collaboration'. What is evident from these responses is that it has been difficult for organisations to understand - and hence commit - to the parameters required by agile development. One respondent put this down to a "poor understand[ing] of Agile techniques...resistance from some areas...stakeholders thinking that Agile means requirements can be delivered late or not at all, rather than being delivered then changed later." It appears that businesses are not fully aware of what agile means. They view it is a software development process only and do not understand that agile development requires its own, or at least an adaptation of, overreaching management processes. This could also account for change management as being an oft-indicated area of project difficulty within agile frameworks - although not making the top of the list, still many respondents commented that stakeholders can be non-collaborative and inflexible in a process meant to specifically facilitate them. Without clear-cut and well-documented guidelines in place, change management within agile development endeavours can become a focus of conflict and delay.

Finally, when asked if they would prefer to return to traditional development instead of agile development, the majority of participants clearly indicated they would not want to go back. Only one person indicated they preferred traditional methods. The remaining 29% indicated that it did not bother them either way.

Limitations & Future Research

There are several of limitations of this study which also lead to future research. As this was an exploratory or pilot study conducted within a short timeframe and within a small network of participants future research could expand upon this study with an aim to reach a wider audience, gathering more data, and potentially compare traditional and agile project management in with knowledge of actual practice.

Another limitation of this exploratory research was the limited opportunity to ask in-depth questions as the survey was kept short in order to encourage participation. An option would be to carry out a more comprehensive study using one or more practitioner organisations and, via observation and

interview, examine more closely agile project management in-situ with the aim of identifying a project management life cycle specific to agile.

Another area for future research comes from the data obtained in this study. When managing agile projects, 26% of participants stated that they used traditional project management methods, while over half of the respondents indicated they adapted traditional techniques for the purposes of managing an agile development project. Future research therefore could be undertaken to define a lifecycle that incorporates both project management and agile development methods into the one lifecycle. For this to occur the actual project initiation, execution and closure processes within organizations would be investigated in order to then define a more streamlined the project process which would permit agile processes and practices to occur at both the project management and product development levels of the lifecycle.

Conclusions

From the survey results and the ensuing discussion it is apparent that a specific framework or paradigm for managing agile projects is currently non-existent in practice in organisations that have adopted agile processes for software development. The agile process appears to only exist in the actual development of a software product rather than in the overall management of the project.

It was apparent that the line between traditional and agile project management is blurred – it can be very uncertain where one finishes and the other starts; handover is unclear. Agile project processes appear to exist for specific software development activities although, in practice, a hybrid model often gets employed as a result of project leaders cherry-picking approach elements from both traditional and agile product development styles. Project leaders, equipped with traditional project management tools/techniques, choose to apply knowledge as seems appropriate, adapting as required to each stage of an agile project.

Currently, the foundation for managing agile projects comes from the traditional project management mindset and experience. Hence we could conclude that agile project management is a modernisation of traditional project management. However, in the future as agile principles become more embedded into businesses, and experiences of staff become purely agile, the potential exists for another project management evolution from which a purely agile project management framework will surface, one that can be applied to better suit the needs of the management of agile projects and will have the ability to operate beyond the realm of just software development.

References

- Agile Alliance (2013) *The Twelve Principles of Agile Software*,
URL:<http://www.agilealliance.org/the-alliance/the-agile-manifesto/the-twelve-principles-of-agile-software/> accessed on 4th July 2013
- Augustine, S., Payne, B., Secindiver, F., Woodcock, S. (2005), 'Agile Project Management: Steering from the Edges', *Communications of the ACM*, 48(12); pp85-88
- Baird, A., Riggins, F. (2012), 'Planning and Sprinting: Use of a Hybrid Project Management Methodology within a CIS Capstone Course', *Journal of Information Systems Education*, 23(3); pp243-257
- Boehm, B. (2002), 'Get ready for Agile methods, with care', *Computer*, 35(1); pp64-69
- Butler, T. (1998) Towards a hermeneutic method for interpretive research in information systems, *Journal of Information Technology* (13), pp 285-300.
- Griffiths, M. (2004), 'Using Agile alongside PMBOK', *PMI Global Congress Proceedings*, pp1-6
- Marchewka, J.T. (2012), *Information Technology Project Management – Providing Measurable Organisational Value 4th Edition*, John Wiley & Sons Inc., USA
- Morse, J.M. & Richards, L. (2002) *Readme First for a Users Guide to Qualitative Research*, Sage Publications Thousand Oaks, California

- Myers, M.D. (2006) *Qualitative Research in Information Systems*, URL: www.qual.auckland.ac.nz accessed on 25th September 2007
- Project Management Institute (2013), *A Guide to the Project Management Body of Knowledge (PMBOK Guide) Fifth Edition*, Project Management Institute, Inc., USA
- Santos, M.A., Bermejo, P.H.S, Oliveira, M.S., Tonelli, A.O. (2011), 'Agile Practices: An Assessment of Perception of Value of Professionals on the Quality Criteria in Performance of Projects', *Journal of Software Engineering and Applications*, (4), pp700-709
- Sliger, M. (2008), 'Agile project management and the PMBOK Guide', *PMI Global Congress Proceedings*, pp1-6
- Suman, U., Uikey, N. (2012), 'An empirical study to design an effective Agile project management framework', *Proceedings of the CUBE International Information Technology Conference*, pp385-390
- The Standish Group (2012), *The Chaos Manifesto*, The Standish Group International.
- Williams, L. (2012), 'What Agile Teams Think of Agile Principles', *Communications of the ACM*, 55(4); pp71-76
- Wysocki, R.K. (2012), *Effective Project Management*, John Wiley & Sons Inc., USA
- Xukan, X., Zhang Bao-feng, L.J., (2012), 'Management Information System Requirements Analysis Model Based on the Agile Development', *2012 International Conference on Control Engineering and Communication Technology*, pp986-990