Investigating Agile Adaptation for Project Development

Rajesh H. Kulkarni[1], P.Padmanabham[2], Manasi Harshe[3], K.K.Baseer[4], Pallavi Patil[5]
[1]Department of Computer Engineering, JSPM, NTC S.P.P University, India
[2]Department of Computer Science and Engineering.BIET, J.N.T.University, India
[3]Department of Computer Engineering, BSCOER, S.P.P University of Pune, India
[4]Department of Computer Science, JNIAS, JNTUA (India)
[5]Department of Computer Engineeering JSPM, NTC S.P.P University, India

Article Info

Article history:

Received Dec 9, 2016
Revised Mar 27, 2017
Accepted Apr 11, 2017

Keyword:

Agile process model
Project management
Rapid application development
Software Engineering

ABSTRACT

Agile methodologies adaptation in software companies is an accepted norm.Rapid application development and efficient deliverables being the premise. The changing business needs, cost effectiveness and timely delivery are catered to by agile methods. Different software development models are in the literature and also are being used by the industry. Few companies have adopted agile, few are gearing up for and few are in transition. We investigated agile presence in a software company. Project management is an evolving art with innovative methods being added up. The aim of this paper is to tease about project management challenges, agile, agile transition in companies and a comparison of conventional software engineering practices with agile process model. This paper takes stock of the current status of quality in software projects and to add on quality process improvement strategies. The results presented here are after a qualitative interview study with one cross function team using streamline development framework in agile development.

Corresponding Author:

Rajesh H. Kulkarni,
Department of Computer Engineering,
JSPM, NTC S. P. P. University, India.
Email: 2015rajeshkulkarni@gmail.com

1. INTRODUCTION

There exist several Software Development Lifecycle (SDLC) Models, but they are rarely followed by organizations for the real project implementations as they lack suitability [1]. A project is a sequence of unique, complex, and connected activities that have one goal or purpose and that must be completed by a specific time, within budget, and according to specification [2], [3]. A project comprises a number of activities that must be completed in some specified order, or sequence. An activity is a defined chunk of work [4]. The sequence of the activities is based on technical requirements, not on management prerogatives.Although the project manager treats the specification as fixed, the reality of the situation is that any number of factors can cause the specification to change.Agile Manifesto triggered Agile approaches [5]. Agile process model (ASP) [6-9] are adaptive, flexible, coherent and faster [10].

After the publication of Agile manifesto, approach of Agile Management has increased in the software development or IT projects. It is necessary to use agile practices and principles in every project that faces uncertainty. Many popular agile practices did exist in traditional projects even before the Agile manifesto. However, we cannot say that certain agile practices cannot be utilized for projects that are still carried out in the traditional way. Certain agile practices can still be utilised for projects, which are following traditional way of implementation [3], [5], [1].

The agile methods emphasises the concurrent execution of the traditionally successive tasks of a project and the constant coordination of participants. The essence of the approach involves constant updating of the execution, and detailed planning of smaller cycles (iterations) to implement a project according to the current results, lessons learned, new ideas etc. The focus on the user is very important and the project team therefore usually involves a representative of end users who regularly checks the partial results of the project. The agile software development evolved in 1990s as a flexible means of handling changing requirements [6]. [11-13] speak about agility and agile methods such as scrum and XP as well the major strength of agile being communication.The reports of Standish Group as mentioned in[14], [15] highlight the reason of high success rate of agile being effective communication and rapid development.

2. LITERATURE REVIEW

Here, the focus of discussion is agile techniques, challenges involved, comparison with traditional methods, pros and cons, adaptability. These approaches give a major description about the previously available best methods with regard to their features along with the knowledge gained after the application of those approaches in the industrial field. This section summarizes agile process model in a brief manner. Research is needed to provide new approaches and to harmonize the existing methods. Clearly, agile concepts will continue to migrate into traditional organizations (and vice versa) through planned or clandestine vectors.In 2002, in [16] a technical report at VTT on agile software shows procedures as well as the functions involved. The adaptability and effectiveness is highlighted in [15-19]. Yet, only a smaller range of empirically validated studies satisfy these features. A comparative study on agile methodologies [20] has stated that the empirical support is needed.Relationship between agile development and the Capability Maturity Model (CMM) is elaborated in [21] and expressed the possibility of amalgamation of agile with conventional methods. XP and agile modeling were explored in [22-24].

In [25] human and social factors affecting agile methods are discussed. In [26] XP model was analysed for better quality. In [27] Scrum influences on development team are verified. Migration from traditional to agile development is explored in [28]. In [29], a group of methods for co-located teams of varying sizes and criticality through Clear, Yellow, Orange, Red, Blue was used for representation purposes. The largely agile method, known as Crystal Clear, show interest on communication in small teams that are involved in software development and it is of less significance. Clear development can be accomplished using seven traits, namely, requisites for the technical environment, regular delivery, osmotic communication, personal protection, reflective enhancement, concentration and trouble-free access to expert users.

A-Pro-PD model development was checked in [30], [31]. A combination of model-driven and agile development was dealt in [32] and it gives more importance to the initial project model, splitting of the work in terms of features and offering an iterative design for each and every feature. All the iterations associated with each and every feature involves two phases, namely, the design and the development.

AVModel design is elaborated in [33]. Project plannings with feedback control loops is elaborated in [24]. Lean methods were implemented in Toyota production system [23] where speed, adaptability and power were a plus. Agile concept implementation in a project using DCR graphs was discussed in [34] alongwith the advantages and disadvantages.Good programming practices such as pair programming, refactoring and testing were recommended in [30]. Evaluation of effectiveness of development team members was critiqued in [11], [35], [36] using shared mental models.

As per [37]: "Many Software Development Lifecycle Models (SDLCs) exist. Evolution of various models was result of the industry need. Any SDLC process model should be based on the best industrial practices.The traditional SDLC process models are not sufficient to cater to current needs. These models are not suitable for real time projects”.

As per [38]: "To develop quality software on a predictable schedule, the requirements must be established and maintained with reasonable stability throughout the development cycle. Changes will have to be made, but they must be managed and introduced in an orderly way. Hence, change management is a critical part of any SDLC model". As per [39]: “Agile processes focus on immediately delivering functionality, while traditional methods focus on optimizing development over a longer period. The traditional methods longer life cycles require adjustments to the agile processes”.

[39] is the base for Agile factors, web questionnaire methodology and likert analysis. The research approach and method are same as [39] in a different environment.

As per [40]: “The lack of direct contact between the development team and the customers could encumber the process of specifying requirements for the future. In turn, handoffs among functions can cause delays and increasing risks of information being misunderstood. Level of details is varying depending on representatives between customer and developer. As a result, the developed system is frequently not satisfactory or even lead to project failure”.

As per [41]: “The process model must focus on identifying the errors in the same or closest phase of the SDLC process to avoid or reduce the redo-work and cost. Most of the existing traditional SDLC process models don't involve management team directly with the development team. Hence, the project management team does not have direct communication with the development and associated members. The management just remains as a silent intermediate communication body. Thus, proper management observation and control is hidden in the development process. As a result, the development process lacks proper management supervision and controls. In addition, the project has to suffer from resource shortage, risk handling, coordination and many other conflicts and problems".

As per [32]: "Managing variability in subsystems and teams has proven difficult. If both agile and traditional teams are developing software for the same product, they can develop radically different artifacts that might not integrate easily. Without some means of coordination, an agile team's domain assumptions, GUIs, or commercial off-the-shelf choices could vary significantly from other developers' counterpart assumptions".

3. RESEARCH METHOD

A survey has been carried out through printed questionnaire. Here, we tried to know how Agile Software Development practices affect the performance of software development projects, what are the important changes that are required and the challenges involved in adopting these agile practices. This questionnaire was given to software practitioners such as director, senior manager, manager, architect, team leader, developer, tester, who has experience practicing software development using both plan-driven (traditional) and agile approaches and methodologies.

The information collected from the survey questions enabled us to enhance our understanding of the critical changes required and the challenges/risks involved in introducing agile software development practices in organizations practicing plan-driven (traditional) software development methodologies. The likert statistical analysis points are depicted i TABLE I. The questionnaire was conducted and categorised on the basis of following agile factors derived from [39]:

a. Specification
b. Project Schedule
c. Team Work
d. Client Collaboration
e. Challenges/Risk Involved

A questionnaire was distributed to two teams working on software projects using agile methodologies of multinational Indian software company.The number of respondents was 14 though the target was 25. The questionnaire results shed light on the agile practices being followed. The team members were very much synchronized and worked in unison.The questionnaire responses and results reflected the same with few different voices.We used a two level Likert scale and the responses were collected from the respondents.

As in [39] groupwise questionnaire contents and questions are as followed:

a. Specifications of Project & Product
   1. Is agile project management applicable in the context of product development?
   2. Can certain agile practices be utilised for projects that are still carried out in traditional manner?
   3. Does project specifications include an assessment of the functions of the product?
   4. Are specifications prepared jointly by the client and the project team?
   5. Should more emphasize be laid on face-to-face communication for conveying information/specifications to and within the development team?
   6. Are changing requirements welcome, even late during project/product development?
   7. Are the team members always willing to continuously learn from one another and open for training through mentoring and professionally guided discussions?

b. Project Schedule
   1. Was the project divided into short (same length) cycles/ iterations during which the team was focused on individual functions (or set of functions) of the product
   2. Do you try to make important project decisions rapidly within short timeframes?
   3. Does your software development team rely on internalized, informal, undocumented plans (as against formal documented plans)?
   4. Do you believe in short, iterative, test-driven, and people-centric development?
   5. Would you shift from lifecycle-based development to feature-driven evolutionary and iterative development?
   6. Do you measure and track progress based on working software?
   7. Do you practice simple designs, processes, and approaches in your software development methodologies?

8. Is agile project management applicable in the context of product development?
9. Can certain agile practices be utilised for projects that are still carried out in the traditional manner?

c. Role of Teamwork
1. The members in the team should be geographically co-located.
2. Work should be in small teams (no more than 4-5 members) in the projects.
3. Team members should regularly discuss their mistakes and learn from them
4. In most cases, communication and negotiation in the projects happen between people who are physically close to one another.
5. In most cases, communication and negotiation in the projects happen between people who work in the same (similar) time zone.
6. The business team and developers should work together daily (closely) throughout the project.
7. The development teams should be self-organizing to meet the changing requirements and the newly arising challenges of the business.
8. At regular intervals, the team should reflect on how to become more effective.
9. The team generally should consists of technically competent and experienced people
10. The majority of members of the team should have similar social culture, even though they might be belonging to different nationalities and provinces.
11. The team members should in general always be willing to continuously learn from one another.

d. Client Collaboration
1. Should customers closely collaborate with the development team members?
2. In software development projects, customers are committed to the project, i.e., they are motivated, active, and consider themselves to be responsible elements of the project.
3. In the projects, a very high priority is given to achieve customer satisfaction.
4. Customer/client should propose and evaluated (cost, value added) changes together with the team.
5. Customer/client user should participate in the development of test procedures.
6. Software development project team follows continuous attention to technical excellence and good design for development.
7. Customer satisfaction has given a high priority through early and continuous delivery of valuable software.

e. Challenges/Risks Involved
1. Problems arise within development teams that are geographically distributed and not co-located in agile development.
2. Differences in productivity between team members in agile software development.
3. Challenges of the agile teams in integrating the development processes and subsystems with teams within the same organization practicing traditional development methodologies.
4. Adopting agile methodologies for use in legacy systems, which are more resistant to changes in internal source code.
5. Problems with selecting the appropriate agile methodology and the supporting tools according to organizational needs and characteristics.

4. RESULTS AND DISCUSSION

The responses result from the specifications of the project and the product, project schedule, role of teamwork, client collabortation, challenges/ risk involved and agile success are Figure 1, Figure 2, Figure 3, Figure 4, Figure 5 and Figure 6, respectively. Applicability of agile practices for the current undergoing project was found to be 79%, scope for migrating from traditional practices to agile practices was found to be 43%, assessment of product functions was found to be 36%, 22% of the project specifications were jointly prepared by client and the project team, 86% recommendation for face to face communication was given, changing requirements are anticipated and welcomed by 79% of respondents to the questionnaire and 86% of the respondents have shown the willingness to learn.

100% respondents affirmed that the modules were divided intoshort cycles with focus on functions, rapid decision making was recommended by 93%, 54% rely on formal document plans, iterative, self-drive, people centric development was recommended by 86%, migrating to iterative development was recommended by 71%, progress tracking on working software is done by 71%, simplicity is followed by 78%, 71% say agile is applicable in the context of product development and possibility of migration is suggested by 79%.

Role of Teamwork resulted in the responses as in Figure 3, 93% suggested geographical colocation of team members is needed, smaller team size is recommended by 29%, 93% recommend periodic interaction among team members, 21% suggest physical proximity leads to higher communication, 86% feel similar time zone leads to better communication, 100% suggest business and developer team should work together and as well self organizing teams ready to take challenges.

Client Collaboration resulted in the responses as in Figure 4. Close collaboration, technical excellence, good design, customer satisfaction are high priority with a rating of 100%, customer commitment and custome client equal stakeholdership in test procedures has a rating of 93% while in cost evaluation it is 86%.

Challenges/Risks Involved resulted in the responses as in Figure 5. Appropriate tool selection problem got a 100% rating, differences, challenges of integration; adoption got 93% rating. Figure 6 depicts agile success rate depicted in ambysoft survey 2013.

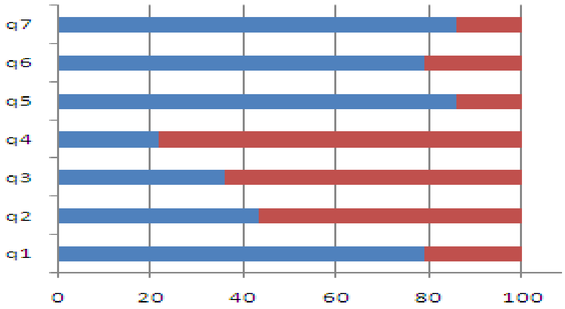

Figure 1. Specifications of Project and Product

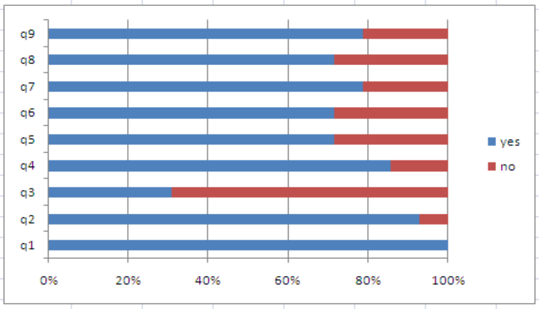

Figure 2. Project Schedule

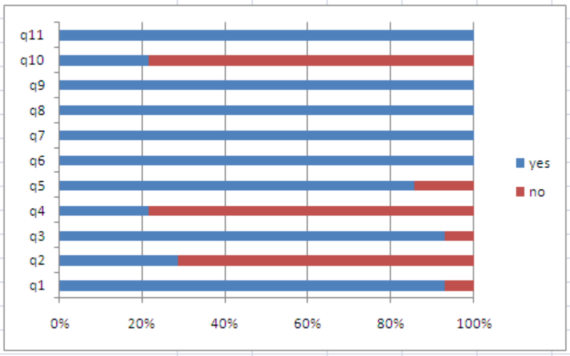

Figure 3. Role of Teamwork

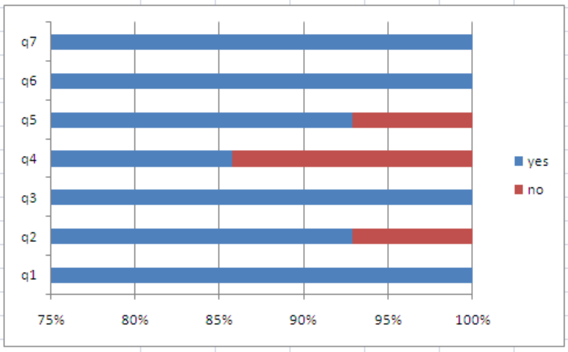

Figure 4. Client Collaboration

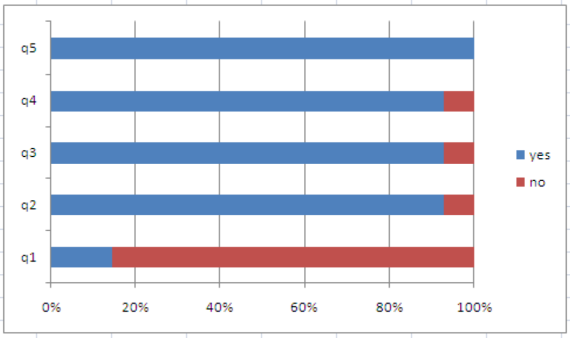

Figure 5. Challenges/Risks Involved

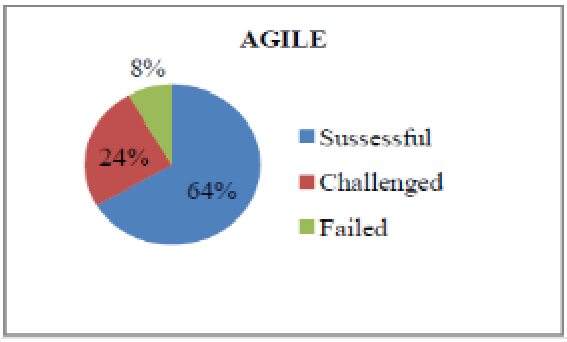

Figure 6. Agile Success Chart (Ambysoft)

The statistics of the likert analytics has been presented in Table 1. The statistical report on specifications on project and product, project schedule, role of teamwork, client collabortation, challenges/ risk involved and agile success are discussed using Table 1. The statistics reveal that the response rate remains consistent at 56%, whereas the standard deviation is lesser than 50%. Such reduced standard deviation ensures the consistency of the results as well as the reliability of the software. The mean performance is above 70% that ensures the fulfillment of project requirements.

Table 1. Likert Analysis Statistics on details of the project

Program Name	Specifications of Project & Product					
	Q1	Q2	Q3	Q4	Q5	Q6
Total Points Received	11	6	5	3	12	11
Total Points Possible	28	28	28	28	28	28
Mean	0.78571	0.42857	0.35714	0.214285714	0.85714	0.78571
Standard Deviation	0.426	0.514	0.497	0.426	0.363	0.426
Response Rate	56%					

Program Name	Project Schedule								
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9
Total Points Received	14	13	7	12	10	10	11	10	11
Total Points Possible	28	28	28	28	28	28	28	28	28
Mean	1	0.928571429	0.5	0.857142857	0.71429	0.71429	0.78571	0.71429	0.78571
Standard Deviation	0.000	0.267	0.855	0.363	0.469	0.469	0.426	0.469	0.426
Response Rate	56%	56%	56%	56%	56%	56%	56%	56%	56%

Program Name	Role of Teamwork										
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11
Total Points Received	13	4	13	3	12	14	14	14	14	3	14
Total Points Possible	28	28	28	28	28	28	28	28	28	28	28
Mean	0.928571429	0.28571	0.92857	0.214285714	0.85714	1	1	1	1	0.21429	1
Standard Deviation	0.267	0.469	0.267	0.426	0.363	0.000	0.000	0.000	0.000	0.426	0.000
Response Rate	56%	56%	56%	56%	56%	56%	56%	56%	56%	56%	56%

Program Name	Client Collaboration						
	Q1	Q2	Q3	Q4	Q5	Q6	Q7
Total Points Received	14	13	14	12	13	14	14
Total Points Possible	28	28	28	28	28	28	28
Mean	1	0.928571429	1	0.857142857	0.92857	1	1
Standard Deviation	0.000	0.267	0.000	0.363	0.267	0.000	0.000
Response Rate	56%	56%	56%	56%	56%	56%	56%

Program Name	Challenges / Risks Involved				
	Q1	Q2	Q3	Q4	Q5
Total Points Received	2	13	13	13	14
Total Points Possible	28	28	28	28	28
Mean	0.14286	0.928571429	0.92857	0.928571429	1
Standard Deviation	0.363	0.267	0.267	0.267	0.000
Response Rate	56%	56%	56%	56%	56%

Number of Likert Scale Options	2
Questionnaire Participants	14
Total Possible Participants	25

All responses are based off a Likert scale (Strongly Agree to Strongly Disagree)
Strongly Agree=1, Strongly Disagree=0

5. CONCLUSION

In this paper a critical review of agile practices being adopted in a software company is done. Using questionnaire and likert scale we analysed the impact of various agile factors on team members.Project specifications, teamwork, risks, challenges and other agile factors effect on the team was acquired using a questionnaire and using likert scale it was quantified. Face to face communication, dividing the modules into small iterations, client and developer collaboration, appropriate tool selection was few of the factors which rank high among the team members using agile methodologies for software development.

ACKNOWLEDGMENT

The authors wish to thank Dr.Anirudha Joshi, IDC, IIT Mumbai for all the support and guidance. The authors would like to thank all the industry and academic participants of this analysis.

REFERENCES

[1] Sonali Pathak, Pallavi Saxena, “Hybrid Methodology Involving Scrum and Waterfall Model towards the Software Project Development in Academic Knowledge Centers”, *International Journal of Evaluation and Research in Education (IJERE)*, Vol 1 No 1, 2012 pages 25-32

[2] Robert K. wysocki, “Effective Project Management, Traditional, Agile, Extreme”, Wiley, Sixth Edition

[3] Aljaz Stare, “Agile Project Management – A Future Approach to the Management of Projects?” *Dynamic Relationships Management Journal*, May 2013.

[4] Pratiyush Guleria Guleria, Vikas Sharma, Manish Arora, “Development and Usage of Software as a Service for a Cloud and Non-Cloud Based Environment- An Empirical Study”, *International Journal of Cloud Computing and Services Science (IJ-CLOSER)* Vol 2 No 1, 2013 pages 50-58

[5] www.agilemanifesto.org

[6] MiXio Aoyama, *“Agile Software Process Model”*, in Proceedings of Twenty-First Annual International Conference on Computer Software and Applications, pp. 454 - 459, 1997.

[7] R. McCauley, “Agile Development Methods Poised to Upset Status Quo”, *SIGCSE Bull*, vol. 33, pp. 14–15, 2001.

[8] O’Leary P, Ali Babar M, Thiel S, Richardson I., *“Product Derivation Process and Agile Approaches: Exploring the Integration Potential”,* in Proceedings of Second IFIP Central and East European Conference on Software Engineering Techniques. Wydawnictwo NAKOM: Pozn n, Poland, 2007.

[9] Xiaodan Yu, Stacie Petter, "Understanding Agile Software Development Practices using Shared Mental Models Theory", *Information and Software Technology*, Vol. 56, no. 8, pp. 911–921, August 2014.

[10] Adam Hendra Brata, Deron Liang, Sholeh Hadi Pramono, “Software Development of Automatic Data Collector for Bus Route Planning System”, *International Journal of Electrical and Computer Engineering (IJECE)*, Vol 5 No 1, 2015 pages 150-157

[11] K. Conboy, “Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development”, *Information Systems Research*, Vol. 20, pp. 329–354, 2009.

[12] P. Abrahamsson, O. Salo, J. Ronkainen, J. Warsta, “Agile Software Development Methods: Review and Analysis”, VTT Publications, 2002.

[13] J. Highsmith, A. Cockburn, “Agile Software Development: The Business of Innovation”, *Computer*, Vol. 34, pp. 120–122, 2001.

[14] R.L. Glass, “The Standish Report: Does it really Describe a Software Crisis?”, *Communication ACM*, Vol. 49, pp. 15–16, 2006.

[15] S. Group, 2011 CHAOS Report, 2011.

[16] P. Abrahamsson, O. Salo, J. Ronkainen, J. Warsta, *“Agile Software Development Methods: Review and Analysis”,* VTT Technical report, 2002.

[17] D. Cohen, M. Lindvall, P. Costa, “An Introduction to Agile Methods”, in: M.V. Zelkowitz (Ed.), *Advances in Computers, Advances in Software Engineering*, Vol. 62, Elsevier, Amsterdam, 2004.

[18] J. Erickson, K. Lyytinen, K. Siau, Agile Modeling, “Agile Software Development, and Extreme Programming: The State of Research”, *Journal of Database Management*, Vol. 16, no. 4, pp. 88–100, 2005.

[19] P. Krutchen, “The Rational Unified Process: An Introduction”, third ed., Addison-Wesley, Boston, 2003.

[20] P. Abrahamsson, J. Warsta, M.T. Siponen, J. Ronkainen, “New Directions on Agile Methods: A Comparative Analysis”, in: Proceedings of the 25th International Conference on Software Engineering (ICSE’03), IEEE Press, 2003.

[21] M.C. Paulk, C.V. Weber, B. Curtis, M.B. Chrissis, “The Capability Maturity Model: Guidelines for Improving the Software Process”, Addison-Wesley, Boston, 1995, ISBN 0-201-54664-7.

[22] J. Erickson, K. Lyytinen, K. Siau, Agile Modeling, “Agile software development, and extreme programming: the state of research”, Journal of Database Management, Vol. 16, no. 4, pp. 88–100, 2005.

[23] M. Poppendieck, T. Poppendieck, “Lean Software Development – An Agile Toolkit for Software Development Managers”, Addison-Wesley, Boston, 2003, ISBN 0-321-15078-3.

[24] K. Schwaber, M. Beedle, “Agile Software Development with Scrum”, Prentice Hall, Upper Saddle River, 2001.

[25] T. Dyba, T. Dingsoyr, “Empirical Studies of Agile Software Development: A Systematic Review”, *Information Software Technology*, Vol. 50, pp. 833–859, 2008.

[26] Y. Khramov, *“The Cost of Code Quality”*, in Proceedings of the AGILE Conference, pp. 119–125, 2006.

[27] G. Benefield, *“Rolling out Agile in a large enterprise”,* in: Proceedings of the 41st Annual Hawaii International Conference on System Sciences, IEEE Computer Society, 2008.

[28] F. Ji, T. Sedano, *“Comparing Extreme Programming and Waterfall Project Results, Software Engineering Education and Training (CSEE&T)”*, in Proceedings of 24th IEEE-CS Conference, pp. 482–486, May 2011.

[29] A. Cockburn, “Crystal Clear: A Human-Powered Methodology for Small Teams, Addison-Wesley”, 2004, ISBN 0-201-69947-8.

[30] K. Beck, “Extreme Programming Explained: Embrace Change”, Addison-Wesley, 2000, ISBN 0-201-61641-6.

[31] J. Stapleton, “DSDM: Business Focused Development”, second ed., Pearson Education, 2003, ISBN 978-0321112248.

[32] Aljaz Stare, “Comprehensive Management of Project Changes”, *Economic and Business Review*, Vol. 12, No. 3, 2010, 195–210.

[33] Martin McHugh, OisínCawley, Fergal McCaffery, Ita Richardson, and Xiaofeng Wang, *“An Agile V-Model for Medical Device Software Development to Overcome the Challenges with PlanDriven Software Development*

Lifecycles", in proceedings of 5th International Workshop on In Software Engineering in Health Care (SEHC), Pp. 12-19, 2013.

[34] Soren Debois, Thomas Hildebrandt and Morten Marquard, *"A Case for Declarative Process Modelling: Agile Development of a Grant Application System",* in proceedings of 18th International Conference Workshops and Demonstrations on Enterprise Distributed Object Computing, 2014.

[35] T. Dyba, E. Arisholm, D. Sjøberg, J. Hannay, F. Shull, "Are two heads better than one? On the effectiveness of pair-programming", *IEEE Software*, Vol. 24, no. 6, pp.10–13, 2007.

[36] H. Erdogmus, M. Morisio, M. Torchiano, "On the Effectiveness of the Test-first Approach to Programming", *IEEE Transactions on Software Engineering*, Vol. 31, no. 3, pp. 226–237, 2005.

[37] ArdhenduMandal and S. C. Pal, "Investigating and Analysing the Desired Characteristics of Software Development Lifecycle (SDLC) Models", *International Journal of Software Engineering Research & Practices*, Vol. 2, Issue 4, October, 2012.

[38] Martin Michlmayr, Francis Hunt, David Probert, *"Quality Practices and Problems in Free Software Projects"*, Proceedings of the First International Conference on Open Source Systems Genova, 11th-15th July 2005

[39] Aljaz Stare *"Agile Project Management in Product Development Projects"*, Procedia - Social and Behavioral Sciences, Volume 119, 19 March 2014, Pages 295-304.

[40] Andrei Antanovich, Anastasia Sheyko& Brian Katumba, "Bottlenecks in the Development Life Cycle of a Feature", Bachelor of Science in Software Engineering and Management, Thesis Report No. 2010:012 ISSN: 1651-4769.

[41] Barry Boehm, Richard Turner, "Management Challenges to Implementing Agile Processes in Traditional Development Organizations", Published by the IEEE Computer Society, 0740-7459/05/$20.00 © 2005 IEEE.