2014 47th Hawaii International Conference on System Science

State-of-the-Art: A Systematic Literature Review on Agile Information Systems Development

Markus Hummel
Goethe University
hummel@wiwi.uni-frankfurt.de

Abstract

Principles of agile information systems development (ISD) have attracted the interest of practice as well as research. The goal of this literature review is to validate, update and extend previous reviews in terms of the general state of research on agile ISD. Besides including categories such as the employed research methods and data collection techniques, the importance of theory is highlighted by evaluating the theoretical foundations and contributions of former studies. Since agile ISD is rooted in the IS as well as software engineering discipline, important outlets of both disciplines are included in the search process, resulting in 482 investigated papers. The findings show that quantitative studies and the theoretical underpinnings of agile ISD are lacking. Extreme Programming is still the most researched agile ISD method, and more efforts on Scrum are needed. In consequence, multiple research gaps that need further research attention are identified.

1. Introduction

Agile methods for software and information systems development (ISD) such as Scrum [1] or Extreme Programming (XP) [2] are very popular in industry. Those methods complement the iterative approach to ISD [1, 3, 4] and have been suggested as a way to react quickly to changing requirements by emphasizing small release cycles and through continuous integration of the customer [5-7]. In contrast to traditional methods, flexibility and autonomy is considered important, the overall project is not planned and scheduled upfront, and the development process is split in small iterations, while encouraging constant feedback of the customer [5, 6]. Consequently, agile ISD methods appear to incorporate many lessons learned about ISD during the past [7, 8].

Following the guidelines of Webster & Watson [9] and Peterson et al. [10], this literature review provides insights into the general state of research on agile ISD in terms of research approaches, methods, data collection techniques, and focus of the studies. The state of theory in the field of agile ISD is also evaluated by looking at theoretical contributions, employed theories and definitions of agility. The goal of this review is to identify research areas that deserve future attention of the research community. Consequently, the following two-part research question is investigated: *What is the state of research on agile ISD, and in consequence, what are the implications for future studies?*

The remainder of this paper is structured as follows. First, related literature reviews on agile ISD are briefly discussed, followed by the review design including details on the data collection process. As a next step, the results of the literature review are presented. In the last section, the findings are summarized and implications for future research are presented, including several research gaps that entail opportunities for future work.

2. Related work

The first reviews were provided by Abrahamsson et al. [11], Cohen et al. [12], and Erickson et al. [13]. These reviews focus on the employed agile practices and methods in industry. For example, Erickson et al. [13] review the state of research on XP and agile modeling. The study finds that most empirical research is concerned with the XP practice pair programming, and other practices are neglected. A more holistic review of agile ISD is provided by Dybå & Dingsøyr [14] who investigate studies up to and including 2005. The authors classify research on agile ISD in four main themes: introduction and adoption, human and social factors, perceptions on agile methods, and comparative studies. They find that there is a need for more rigorous, high quality empirical studies. Another finding is that there is a need for conducting research based on other methods

978-1-4799-2504-9/14 $31.00 © 2014 IEEE
DOI 10.1109/HICSS.2014.579

besides XP because empirical evidence on popular methods such as Scrum is missing.

Most recently, Dingsøyr et al. [15] provide an overview of the theoretical perspectives that are employed by research on agile ISD, but as the authors state themselves, the search results are limited because only the topic of studies were searched, and the search strings were based on a previously defined keyword list of twenty theoretical perspectives, including lightweight theoretical perspectives such as knowledge management and personality. The study concludes in a call for a more theory-based research approach in the field of agile ISD.

3. Review design

3.1. Search categories

Table 1 describes the search categories and the contributions of this review. The search categories are adapted from previous literature reviews [14, 15, 18, 19].

Table 1. Definition of search categories

Search Category		Description	Contribution
General Characteristics			
	Research Approach	In order to classify research approaches, qualitative research and quantitative research is distinguished. A third category named conceptualization is used in order to classify purely conceptual papers that do not include the evaluation of empirical data.	Since research efforts on agile ISD increased significantly in recent years [15], there is a clear need for an updated as well as extended systematic literature review on the characteristics of research on agile ISD.
	Research Method	An inductive search for employed research methods (e.g., case study research) and employed data collection methods (e.g., interviews) is conducted.	
	Data Collection Methods		
	Unit of Analysis	The following units of analysis are distinguished: group, individual, organization, method and tool.	
Content of Studies			
	Focus	An inductive search for the focus of the studies on agile ISD is conducted by manually scanning each paper.	This review refines and updates the categorization of Dybå & Dingsøyr [14].
	Agile Methods and Practices	The literature is searched for the employed agile methods. The individual practices of the most investigated agile method XP are also examined.	Previous systematic reviews [13, 14], which only include studies up to 2005, are updated.
Theory			
	Theoretical Foundations	An open search for theoretical foundations is conducted. This is achieved by scanning the whole studies and not only the topics. Lightweight theoretical perspectives were discerned in order to investigate the application of strong theories and frameworks as defined in Whetten [16].	This review extends previous theory assessments [15, 17].
	Theoretical Contributions	As a measure for theory contribution, this review examines whether any hypotheses are either developed or tested.	
	Definition of Agility	This review examines inductively which definitions of agility are used in the literature on agile ISD.	Insight into whether research-based definitions are actually employed in the literature is provided.

3.2. Data collection

The sources of this literature review are presented in Table 2. Since agile ISD is fundamentally about developing IS, the eight IS journals of the Senior Scholars' Basket of Journals[1] are taken into account, as well as the proceedings of the International Conference on Information Systems (ICIS). The most popular journals and conferences in the software engineering discipline that deal with agile ISD (according to [15]) are also included. The latest included journal issues are the March 2013 issues, and the conference proceedings are included up to and including 2012.

Table 2. Literature review sources

.	Search Results	Selected Papers
IS Outlets:	**80**	**38**
EJIS[2]	17	10
ISJ	8	6
ISR	8	8
JIT	12	0
JMIS	3	2
JSIS	0	0
JAIS	2	0
MISQ	5	2
ICIS	25	10
SE Outlets:	**755**	**444**
ESE	17	14
IEEE ToSE	10	6
IET Software	6	5
IST	31	27
JSS	31	24
JSA	5	1
Agile	92	85
ESEM	25	11
EuroMicro	37	21
EuroSPI	18	5
ICGSE	31	13
IFIP	143	14
ICSE	69	15
LESS	5	5
PROFES	61	35
XP	174	153
Total:	**835**	**482**

[1]http://home.aisnet.org/displaycommon.cfm?an=1&subarticlenbr=346

[2] Please consult Appendix 1 for the full names of the outlets.

Within each outlet, all papers that deal with agile ISD are searched (see Appendix 2 for the search string). Based on the initial search results, the titles and abstracts of the papers were scrutinized and papers that did not focus on agile ISD were excluded. In addition, only full research papers were included in the review. Experience reports, editorials, short papers, and education-related outlets (e.g., CSEE&T conference) were excluded from the analysis in order to focus on the main findings of the literature that are based on rigorous scientific research. Practitioner-oriented magazines (e.g., IEEE Software or IEEE Computer) were also omitted because those papers usually do not give detailed information on the research design which is the focus of this study. This search process resulted in 482 selected papers.

4. Results

4.1. General characteristics

4.1.1. Research approach. In terms of the *research approach*, a tendency towards qualitative research (47.5%) is identifiable. Quantitative research (34%) and conceptualization (23%)[3] have not attracted the attention of the research community to the same extent. In rare cases, both qualitative and quantitative techniques are combined in a single contribution, for example in Lee & Xia [20]. The high number of papers employing a conceptual research approach is caused by many conference papers that deal with tools and testing approaches (e.g., [21, 22]).

4.1.2. Research method. Table 3 presents the results of the category *research method*, including the number of papers, percentages, and examples. The dominant qualitative research method is the case study design which accounts for 38.5% of the contributions. The second most frequently employed research method is experiment which accounts for 13% of the studies. Only 8% of the papers carry out questionnaire-based surveys. The high number of experiments is mostly caused by experiments that investigate different facets of pair programming. Experiments are also conducted concerning the unit testing practice. Research methods such as ethnography (3.5%), action research (2.5%), or grounded theory (5%) do not entail a large extent of the research efforts.

[3] In the following, percentages do not necessarily always add up to 100% because papers can contribute several times in the same category, for example, a paper can employ a qualitative as well as quantitative research approach.

Table 3. Research methods

	#	%	Examples
Case Study	185	38,5%	[6, 23, 24]
Experiment	62	13.0%	[25-27]
Survey	39	8.0%	[28]
Ethnography	16	3.5%	[29]
Simulation	18	4.0%	[30]
Literature Review	25	5.0%	[18]
Grounded Theory	22	5.0%	[31]
Action Research	12	2.5%	[32]
Focus Group	6	1.0%	[33]

4.1.3. Data collection methods. Table 4 illustrates the primary *data collection methods* of the research papers. The salient technique for collecting data is interviews (34%). Log files (20.5%) and questionnaires (15%) are also used, but there is a strong imbalance towards interviews. Log files and questionnaires are mainly analyzed as results of experiments. 4.5% of the contributions do not state their data collection method, and the classification does not apply for 22% of the papers, mostly conceptual.

Table 4. Primary data collection methods

	#	%	Examples
Interviews	163	34.0%	[34, 35]
Log Files	99	20.5%	[36]
Questionnaire	73	15.0%	[28]
Field Notes	34	7.0%	[37]
Literature	25	5.0%	[18]
Video Recordings	7	1.5%	[38]
Audio Recordings	4	1.0%	[39]

4.1.4. Unit of analysis. The results show that the most frequent *unit of analysis* is the group / team / project level (51.5%), followed by the method (26%) and tool (7.5%) level. For example, McAvoy & Butler [37] use the unit of analysis 'team' in order to examine the decision making process in agile teams, whereas Qumer & Henderson-Sellers [40] employ the unit of analysis 'method' for assessing the degree of agility of several agile methods. Agile tools and their usage are a less investigated unit of analysis (e.g., [41]). The individual (6%) and organizational (5%) level are the least investigated. First empirical findings are provided by Hong et al. [42] who investigate individual user acceptance of agile information systems, and Lyytinen & Rose [43] who examine ISD agility on the organizational level.

4.2. Content of studies

4.2.1. Focus. The inductive approach for identifying the *focus* of the studies resulted in 19 different categories, as presented in Table 5. Most research on agile ISD is concerned with the evaluation of agile practices, as well as with the adaption, combination, or extension of agile ISD methods.

Due to space restrictions, a detailed discussion of each category is not attempted here. Please consult the author for more information concerning the categories and the full list of papers in each category.

Table 5. Focus

	#	%	Examples
Evaluation of Agile Practices	96	20.0%	[28]
Adaption / Combination / Extension	55	11.5%	[44]
Tool Support	49	10.0%	[41]
Team Characteristics	49	10.0%	[36]
Adoption	44	9.0%	[45]
Testing	40	8.5%	[27]
Agile vs. Plan-based	39	8.0%	[46]
Customer Perspective	29	6.0%	[47]
Measurement	29	6.0%	[48]
Communication	17	3.5%	[35]
Definition of Agility	17	3.5%	[18]
Collaboration	11	2.5%	[49]
Hybrid Approach	10	2.0%	[30]
Success Factors	7	1.5%	[50]
Release Scheduling	7	1.5%	[51]
Coordination	6	1.0%	[52]
Agile vs. Lean	6	1.0%	[53]
Organizational Culture	5	1.0%	[54]
Requirements Engineering	4	1.0%	[34]

Consequently, the guidelines of Petersen et al. [10] are followed in order to create a systematic map for the focus of the studies and the employed research methods (cf. Figure 1). The map shows that first qualitative findings on many topics already exist, but confirmatory studies in almost all research areas are lacking.

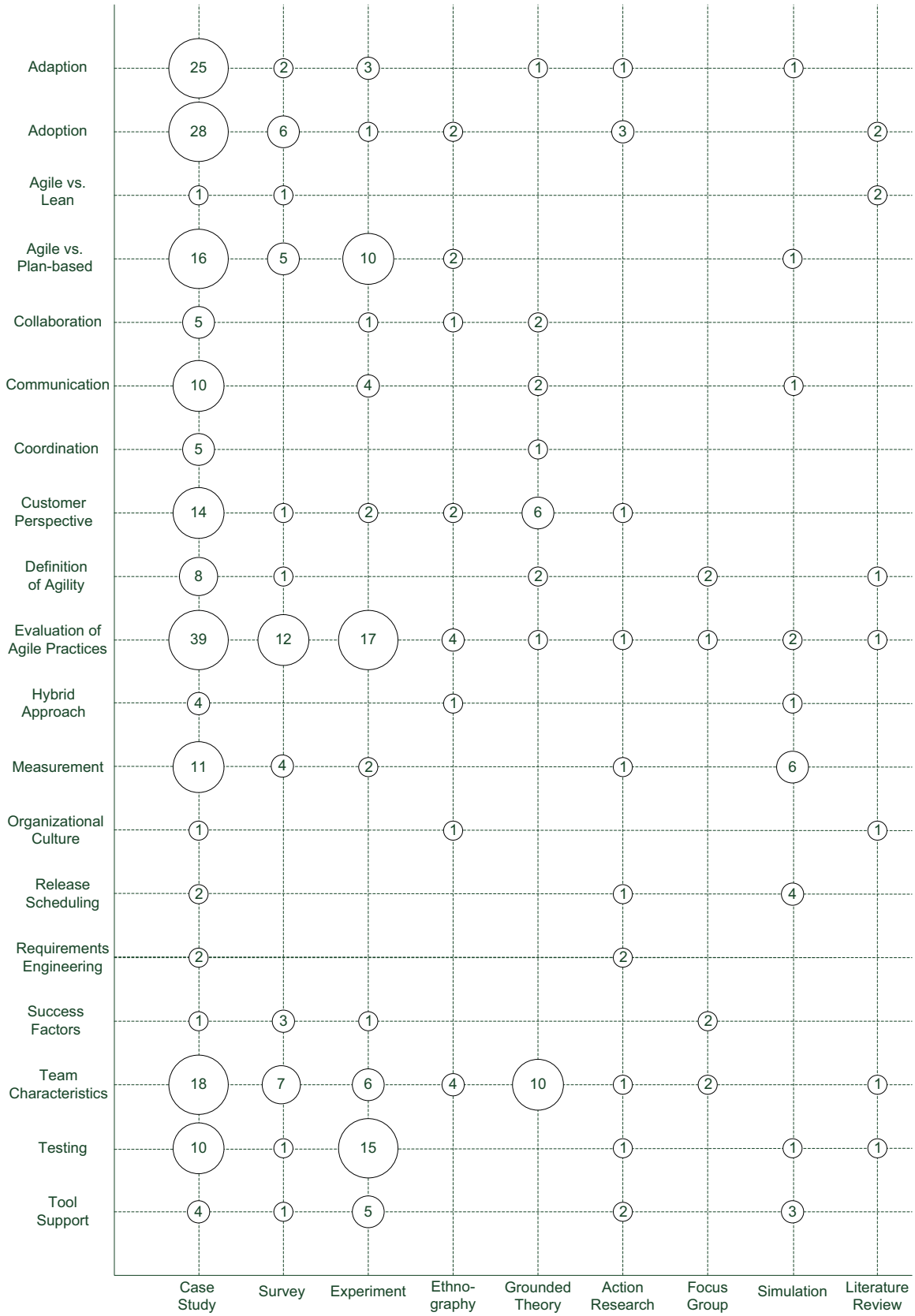

Figure 1. Systematic map for the focus of studies and employed research methods

4.2.2. Agile methods and practices. The findings report that the most frequently reported *agile method* is Extreme Programming (XP) (49%), followed by Scrum (22%). One third of the research is dealing with agile ISD in general. Research on other ISD methods related to agile methods is scarce.

Table 6 illustrates the extent to which the research community has investigated the *practices* of the most investigated agile method XP. The results show that pair programming (35.5%) is the most frequently researched XP practice. Other practices such as unit testing (including test-driven development and automated acceptance testing) (30.5%) and refactoring (25.5%) are nearly as popular as pair programming, but research on those practices is mostly concerned with conceptual papers that deal with the development of models and tools for testing purposes (e.g., [22]). The three practices that are the least examined are simple design (10%), system metaphor and sustainable pace (7.5% respectively).

Table 6. XP practices

	#	%	Examples
Pair Programming	172	35.5%	[36]
Unit Testing	147	30.5%	[27]
Refactoring	122	25.5%	[29]
Planning Game	90	18.5%	[35]
Continuous Integration	78	16.0%	[23]
On-Site Customer	75	15.5%	[47]
Collective Code Ownership	71	14.5%	[5, 55]
Coding Standards	52	11.0%	[55]
Small Releases	52	11.0%	[45]
Simple Design	47	10.0%	[40]
Sustainable Pace	37	7.5%	[23, 32]
System Metaphor	36	7.5%	[32, 40]

4.3. Theory

4.3.1. Theoretical foundations. 57 different theories and frameworks were identified which are employed as *theoretical foundations* of studies on agile ISD. Table 7 provides an overview of the three most frequently applied theories and how they were used in the studies. Besides the theories illustrated in Table 7, two studies respectively use the theories adaptive structuration theory, home ground theory, transactive memory systems, and the distributed cognition for teamwork. The rest of the theoretical foundations are only used in one single paper. The largest part of the literature, accounting for 60% of the papers, relies solely on related work without a strong theoretical background. 25.5% of the literature does not even consult related work.

Table 7. Applied theories in agile ISD

Complex Adaptive Systems Theory (CAS)	
	Vidgen & Wang [6] use CAS theory in order to develop an agile organizing framework that identifies enablers and inhibitors of agility, as well as emergent capabilities of agile teams.
	Kautz & Zumpe [56] use the key concept 'edge of chaos' from CAS theory in order to investigate the beneficial balance between stability and instability of a concrete project.
	Meso & Jain [57] use CAS theory in order to understand how agile practices influence the adaption to changing business needs [15].
	Sacho & Walter [58] use CAS theory to describe the differences between software design and other designs [15].
Control Theory	
	Harris et al. [59] apply control theory to understand when flexibility in software development teams is needed.
	Maruping et al. [28] use control theory in order to investigate agile methodology use and requirements change.
	Cram & Brohman [60] propose a new typology of ISD control on the basis of control theory.
Coordination Theory	
	Strode et al. [52] use coordination theory to analyze the dependencies and associated coordination mechanisms in agile ISD projects.
	Li & Maedche [61] use coordination theory to explain coordination strategies in distributed agile ISD projects.
	Pikkarainen et al. [35] use coordination theory in order to investigate the communication of agile ISD teams.

4.3.2. Theory contribution. The results show that 67% of the investigated papers do not contribute to theory. Of the remaining papers, 20% are categorized as theory testing and 11.5% as theory building. For example, Hong et al. [42] propose and test several hypotheses in order to examine the drivers of user acceptance in agile ISD, whereas Vidgen & Wang [6] build theory by developing an agile organizing framework which emerged out of qualitative case study-based research.

4.3.3. Definition of agility. The findings show that there is no widely accepted *definition of agile ISD* because a multitude of different definitions can be found in the literature. 17 different definitions of agility, agile ISD and agile methods are inductively identified in this review. Most of those definitions are utilized in only one paper. 55% of the papers do not define the concept of agility at all and 25% describe only some basic properties of agile ISD methodologies, without explicitly defining the concept of agility.

The most commonly used definition of agile ISD is the Agile Manifesto [62] (9.5% or 45 papers), which states four values and twelve principles of agile ISD that were derived from experiences of practitioners. The second most cited definition is the research-based definition of definition of Conboy [18] (six papers), followed by the definition of Abrahamsson et al. [11] (five papers).

5. Discussion and concluding remarks

In this study, a systematic, structured literature review in the field of agile ISD was conducted. The data collection resulted in 482 papers that were published in the most popular outlets on agile ISD. This review extends findings of previous reviews [14, 15, 17] by introducing new perspectives and by including contributions until early 2013.

The results show that the state of research on agile ISD is still nascent because there is an imbalance in terms of the employed research methods towards interview-based case studies. Those qualitative research designs are essential for providing first evidence on important factors and relationships, but confirmatory studies testing the qualitative findings are lacking. More studies are needed that are based on quantitative approaches such as field studies or experiments in order to ensure that the qualitative findings are generalizable.

Furthermore, this review exposed promising research areas by presenting a systematic map on the focus of the studies and the employed research methods. Some of these areas, for example the communication patterns of agile teams, are highly important for the successful implementation of agile practices [6, 62] and first qualitative findings exist [e.g., 35], but research remains scarce. The same proposition holds for other research areas such as agile vs. lean, hybrid approaches and organizational culture. In consequence, we need more studies that address the following research gaps:

- *What are the implications of agile ISD on the coordination, collaboration and communication mechanisms within agile teams?*
- *How are agile ISD and lean software development related?*
- *What is the impact of agile practices on the organizational culture?*
- *How can agile methods and traditional, plan-based methods be combined?*
- *What are the implications of agile ISD on release scheduling and requirements engineering?*
- *What are the success factors underlying agile ISD?*

Despite the popularity of Scrum in industry, most researched is based on XP, more specifically on the pair programming, unit testing and refactoring practices. One possible reason for this emphasis on XP is that studies on pair programming may be set up inexpensively in an academic setting with small teams of students. In terms of unit testing and refactoring, many studies propose tools that may support those practices. Future research should focus on other XP practices such as collective code standards and on-site customer. Furthermore, more research is needed that provides theoretically grounded guidance for industry on the adoption, adaption and success factors of Scrum.

Furthermore, this in-depth and rigorous literature search validates the initial findings of Dingsøyr et al. [15] in terms of the limited theoretical foundations of the studies on agile ISD. Findings of the literature are mostly based on experiences which lack empirical and theoretical support. Although 57 theories were inductively identified, a large part of the literature is not testing or building theory. CAS theory, as well as control theory and coordination theory are found to entail the most research efforts.

The definition of agility remains one of the most salient problems of agile ISD. A universal understanding of what constitutes 'agility' is not observable. Attempts of research-based taxonomies for pinpointing the concept of agility [18, 33] build the basis for a common definition, but most papers still rely on the Agile Manifesto [62] that consists of unverified principles and practices of practitioners which are not suitable as a solid theoretical grounding. The use of research-based definitions of agility should be extended in order to enable a better comparison of studies and to increase the value of research on agile ISD.

This review has several limitations. There is only a selection of journals and conferences of the agile ISD domain included in the review, but the essential insights of the literature are captured by consulting

the most important outlets of agile ISD. Second, some contributions may have been missed due to the search procedure. For example, the focus is on the two most popular agile ISD methods Scrum and XP, whereas other agile ISD methods such as Kanban were not included in the search string. Third, the selection of search categories may not be complete but established categories of related literature reviews are employed.

To sum up, this review serves as foundation for future studies by opening up several research possibilities. This is achieved by identifying several research gaps that need further investigation of the research community. Furthermore, an in-depth literature analysis of the exposed under-researched aspects of agile ISD is promising important insights.

7. References

[1] K. Schwaber, and M. Beedle, Agile Software Development with Scrum, Prentice Hall, Upper Saddle River, NJ, USA, 2002.

[2] K. Beck, Extreme Programming Explained: Embrace Change, Addison-Wesley, Boston, MA, USA, 1999.

[3] M. Poppendieck, and T. Poppendieck, Lean Software Development: An Agile Toolkit, Addison-Wesley Longman, 1st ed., Amsterdam, 2003.

[4] J. Martin, Rapid application development, Macmillan Publishing Co., Inc, New York, NY, USA, 1991.

[5] L. Cao, K. Mohan, X. Peng, and B. Ramesh, "A framework for adapting agile development methodologies", European Journal of Information Systems, 18(4), 2009, pp. 332-343.

[6] R. Vidgen, and X. Wang, "Coevolving Systems and the Organization of Agile Software Development", Information Systems Research, 20(3), 2009, pp. 355-376.

[7] J. Highsmith, and A. Cockburn, "Agile Software Development: The Business of Innovation", IEEE Computer, 34(9), 2001, pp. 120-127.

[8] A. Cockburn, and J. Highsmith, "Agile Software Development: The People Factor", IEEE Computer, 34(11), 2001, pp. 131-133.

[9] J. Webster, and R.T. Watson, "Analyzing the past to prepare for the future: Writing a literature review", MIS Quarterly, 26(2), 2002, pp. 13-23.

[10] K. Peterson, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering", in: International Conference on Evaluation and Assessment in Software Engineering, 2008, pp. 68-77.

[11] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, Agile software development methods: review and analysis, VTT Technical Report, 2002.

[12] D. Cohen, M. Lindvall, and P. Costa, "An introduction to agile methods", in (Zelkowitz, M.V.): Advances in Computers, Elsevier Ltd., 2004, pp. 1-66.

[13] J. Erickson, K. Lyytinen, and S. Keng, "Agile Modeling, Agile Software Development, and Extreme Programming: The State of Research", Journal of Database Management, 16(4), 2005, pp. 88-100.

[14] T. Dybå, and T. Dingsøyr, "Empirical studies of agile software development: A systematic review", Information and Software Technology, 50(9-10), 2008, pp. 833-859.

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

[16] D.A. Whetten, "What Constututes a Theoretical Contribution?" Academy of Management Review, 14(4), 1989, pp. 490 - 495.

[17] D. Batra, D.E. Vandermeer, and K. Dutta, "Extending Agile Principles to Larger, Dynamic Software Projects: A Theoretical Assessment", Journal of Database Management, 22(4), 2011, pp. 73-92.

[18] K. Conboy, "Agility from First Principles: Reconstructing the Concept of Agility in Information Systems Development", Information Systems Research, 20(3), 2009, pp. 329-354.

[19] N. Urbach, S. Smolnik, and G. Riempp, "The State of Research on Information Systems Success", Business & Information Systems Engineering, 1(4), 2009, pp. 315-325.

[20] G. Lee, and W. Xia, "Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data", MIS Quarterly, 34(1), 2010, pp. 87-114.

[21] T.D. Hellmann, A. Hosseini-Khayat, and F. Maurer, "Test-Driven Development of Graphical User Interfaces: A Pilot Evaluation", in (Sillitti, A., Hazzan, O., Bache, E., and Albaladejo, X.): Agile Processes in Software Engineering and Extreme Programming, XP 2011, Springer, Berlin Heidelberg, 2011, pp. 223-237.

[22] E. Kim, J. Na, and S. Ryoo, "Developing a Test Automation Framework for Agile Development and Testing", in (Abrahamsson, P., Marchesi, M., and Maurer, F.): Agile Processes in Software Engineering and Extreme Programming, XP 2009, Springer, Berlin Heidelberg New York, 2009, pp. 8-12.

[23] X. Wang, K. Conboy, and M. Pikkarainen, "Assimilation of agile practices in use", Information Systems Journal, 22(6), 2012, pp. 435-455.

[24] S. Sarker, and S. Sarker, "Exploring Agility in Distributed Information Systems Development Teams: An Interpretive Study in an Offshoring Context", Information Systems Research, 20(3), 2009, pp. 440-461.

[25] V. Balijepally, R. Mahapatra, S. Nerur, and K.H. Price, "ARE TWO HEADS BETTER THAN ONE FOR SOFTWARE DEVELOPMENT? THE PRODUCTIVITY PARADOX OF PAIR PROGRAMMING", MIS Quarterly, 33(1), 2009, pp. 91-118.

[26] T. Bipp, A. Lepper, and D. Schmedding, "Pair programming in software development teams - An empirical study of its benefits", Information and Software Technology, 50(3), 2008, pp. 231-240.

[27] M. Müller, and A. Höfer, "The effect of experience on the test-driven development process", Empirical Software Engineering, 12(6), 2007, pp. 593-615.

[28] L.M. Maruping, V. Venkatesh, and R. Agarwal, "A Control Theory Perspective on Agile Methodology Use and Changing User Requirements", Information Systems Research, 20(3), 2009, pp. 377-399.

[29] H. Sharp, and H. Robinson, "An Ethnographic Study of XP Practice", Empirical Software Engineering, 9(4), 2004, pp. 353-375.

[30] D. Port, and T. Bui, "Simulating mixed agile and plan-based requirements prioritization strategies: proof-of-concept and practical implications", European Journal of Information Systems, 18(4), 2009, pp. 317-331.

[31] S. Adolph, P. Kruchten, and W. Hall, "Reconciling perspectives: A grounded theory of how people manage the process of software development", Journal of Systems and Software, 85(6), 2012, pp. 1269-1286.

[32] A. Fruhling, and G.-J. De Vreede, "Field Experiences with eXtreme Programming: Developing an Emergency Response System", Journal of Management Information Systems, 22(4), 2006, pp. 39-68.

[33] S. Sarker, C.L. Munson, S. Sarker, and S. Chakraborty, "Assessing the relative contribution of the facets of agility to distributed systems development success: an Analytic Hierarchy Process approach", European Journal of Information Systems, 18(4), 2009, pp. 285-299.

[34] B. Ramesh, L. Cao, and R. Baskerville, "Agile requirements engineering practices and challenges: an empirical study", Information Systems Journal, 20, 2010, pp. 449-480.

[35] M. Pikkarainen, J. Haikara, O. Salo, P. Abrahamsson, and J. Still, "The impact of agile practices on communication in software development", Empirical Software Engineering, 13(3), 2008, pp. 303-337.

[36] K.S. Choi, F.P. Deek, and I. Im, "Exploring the underlying aspects of pair programming: The impact of personality", Information and Software Technology, 50(11), 2008, pp. 1114-1126.

[37] J. Mcavoy, and T. Butler, "The role of project management in ineffective decision making within Agile software development projects", European Journal of Information Systems, 18(4), 2009, pp. 372-383.

[38] J. Brown, G. Lindgaard, and R. Biddle, "Stories, Sketches, and Lists: Developers and Interaction Designers Interacting Through Artefacts", in (Melnik, G., Kruchten, P., and Poppendieck, M.): AGILE 2008, IEEE, Toronto, Canada, 2008, pp. 39-50.

[39] S. Freudenberg, P. Romero, and B. Du Boulay, ""Talking the talk": Is intermediate-level conversation the key to the pair programming success story?" in (Eckstein, J., Maurer, F., Davies, R., Melnik, G., and Pollice, G.): AGILE 2007, IEEE, Washington, DC, USA, 2007, pp. 84-91.

[40] A. Qumer, and B. Henderson-Sellers, "An evaluation of the degree of agility in six agile methods and its applicability for method engineering", Information & Software Technology, 50(4), 2008, pp. 280-295.

[41] G. Azizyan, M.K. Magarian, and M. Kajko-Matsson, "Survey of Agile Tool Usage and Needs", in: AGILE 2011, IEEE, Salt Lake City, UT, USA, 2011, pp. 29-38.

[42] W. Hong, J. Thong, L. Chasalow, and G. Dhillon, "User Acceptance of Agile Information Systems: A Model and Empirical Test", Journal of Management Information Systems, 28(1), 2011, pp. 235-272.

[43] K. Lyytinen, and G.M. Rose, "Information system development agility as organizational learning", European Journal of Information Systems, 15(2), 2006, pp. 183-199.

[44] L. Cao, K. Mohan, B. Ramesh, and S. Sarkar, "Adapting funding processes for agile IT projects: an empirical investigation", European Journal of Information Systems, 22(2), 2013, pp. 191-205.

[45] K. Petersen, and C. Wohlin, "The effect of moving from a plan-driven to an incremental software development approach with agile practices", Empirical Software Engineering, 15(6), 2010, pp. 654-693.

[46] M. Laanti, O. Salo, and P. Abrahamsson, "Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation", Information & Software Technology, 53(3), 2011, pp. 276-290.

[47] K. Conboy, and L. Morgan, "Beyond the customer: Opening the agile systems development process", Information and Software Technology, 53(5), 2011, pp. 535-542.

[48] V. Mahnič, and T. Hovelja, "On using planning poker for estimating user stories", Journal of Systems and Software, 85(9), 2012, pp. 2086-2095.

[49] R. Hoda, J. Noble, and S. Marshall, "The impact of inadequate customer collaboration on self-organizing Agile teams", Information & Software Technology, 53(5), 2011, pp. 521-534.

[50] S.C. Misra, V. Kumar, and U. Kumar, "Identifying some important success factors in adopting agile software development practices", Journal of Systems and Software, 82(11), 2009, pp. 1869-1890.

[51] Á. Szőke, "Conceptual scheduling model and optimized release scheduling for agile environments", Information & Software Technology, 53(6), 2011, pp. 574-591.

[52] D.E. Strode, S.L. Huff, B. Hope, and S. Link, "Coordination in co-located agile software development projects", Journal of Systems and Software, 85(6), 2012, pp. 1222-1238.

[53] X. Wang, K. Conboy, and O. Cawley, ""Leagile" software development: An experience report analysis of the application of lean approaches in agile software development", Journal of Systems and Software, 85(6), 2012, pp. 1287-1299.

[54] J. Iivari, and N. Iivari, "The relationship between organizational culture and the deployment of agile methods", Information & Software Technology, 53(5), 2011, pp. 509-520.

[55] L.M. Maruping, X. Zhang, and V. Venkatesh, "Role of collective ownership and coding standards in coordinating expertise in software project teams", European Journal of Information Systems, 18(4), 2009, pp. 355-371.

[56] K. Kautz, and S. Zumpe, "Just Enough Structure at the Edge of Chaos: Agile Information System Development in Practice", in (Abrahamsson, P., Baskerville, R., Conboy, K., Fitzgerald, B., Morgan, L., and Wang, X.): Agile Processes in Software Engineering and Extreme Programming, XP 2008, Springer, Berlin Heidelberg, 2008, pp. 137-146.

[57] P. Meso, and R. Jain, "Agile Software Development: Adaptive Systems Principles And Best Practices", Information Systems Management, (Summer 2006), 2006, pp. 19-30.

[58] D. Socha, and S. Walter, "Is designing software different from designing other things?" International Journal of Engineering Education, 22(3), 2006, pp. 540-550.

[59] M.L. Harris, R.W. Collins, and A.R. Hevner, "Control of Flexible Software Development Under Uncertainty", Information Systems Research, 20(3), 2009, pp. 400-419.

[60] W.A. Cram, and M.K. Brohman, "Beyond Modes: A New Typology Of ISD Control", in: ICIS 2010 Proceedings, 2010.

[61] Y. Li, and A. Maedche, "Formulating Effective Coordination Strategies In Aagile Global Software Development Teams", in: ICIS 2012 Proceedings, 2012.

[62] K. Beck, M. Beedle, A. Van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R.C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, http://www.agilemanifesto.org/, 2001, accessed 28th of March, 2013.

Appendix 1. Abbreviations

EJIS = European Journal of Information Systems; ISJ = Information Systems Journal; ISR = Information Systems Research; JIT = Journal of Information Technology; JMIS = Journal of Management Information Systems; JSIS = Journal of Strategic Information Systems; JAIS = Journal of the Association for Information Systems; MISQ = Management Information Systems Quarterly; ICIS = International Conference on Information Systems; ESE = Empirical Software Engineering; ToSE = Transactions on Software Engineering; IST = Information and Software Technology; JSS = Journal of Systems and Software; JSA = Journal of Systems Architecture; ESEM = Empirical Software Engineering and Measurement; EuroSPI = European System & Software Process Improvement and Innovation; ICGSE = International Conference on Global Software Engineering; IFIP = International Federation for Information Processing; ICSE = International Conference on Software Engineering; LESS = Lean Enterprise Software and Systems; PROFES = Product Focused Software Process Improvement.

Appendix 2. Search String

PublicationTitle:“Name” AND ((Title:"Agile" NOT Title:"Manufacturing") OR (Title:"Agility" NOT Title:"Manufacturing") OR Title:"Extreme Programming" OR Title:”Scrum" OR Title:"Pair Programming" OR Abstract:"Extreme Programming" OR Abstract:"Scrum" OR Abstract:"Pair Programming" OR (Abstract:"Agility" AND Abstract:"Development") OR (Abstract:"Agile" AND Abstract:"Development") OR Keywords:"Flexible Information Systems Development" OR Keywords:"Agile Software Development" OR Keywords:"Agile Information Systems Development"). This generic search string was adapted to the specific database in which the sought-after outlet is published in full because not all databases offer the same possibilities for searching.