University of Nebraska - Lincoln

DigitalCommons@University of Nebraska - Lincoln

Library Philosophy and Practice (e-journal) Libraries at University of Nebraska-Lincoln

Fall 12-1-2020

Software Development with Scrum: A Bibliometric Analysis and Profile

Peter Kokol
peter.kokol@um.si

Sašo Zagoranski
Semantika, saso.zagoranski@semantika.si

Marko Kokol
Semantika, marko.kokol@usemantika.si

Follow this and additional works at: https://digitalcommons.unl.edu/libphilprac

Part of the Computer Engineering Commons, and the Library and Information Science Commons

Software Development with Scrum: A Bibliometric Analysis and Profile

Peter Kokol, Marko Kokol, Sašo Zagoranski

Faculty of Electrical Engineering and Computer Science, University of Maribor, 2000 Maribor, Slovenia, peter.kokol@um.si

Semantika, 2000 Maribor, Slovenia, marko.kokol@um.si

ABSTRACT

Introduction of the Scrum approach into software engineering has changed the way software is being developed. The Scrum approach emphasizes the active end-user involvement, embracing of change, and /iterative delivery of products. Our study showed that Scrum has different variants or is used in combination with different methods. Some tools not normally used in the conventional software approaches, like gamification, content analysis and grounded theory are also employed. However, Scrum like other software development approach focuses on improvement of software process, software quality, business value, performance, usability and efficiency and at the same time to reduce cost, risk and uncertainty. Contrary to some conventional approaches it also strives to boost soft factors like agility, trust, motivation, responsibility and transparency. The bibliometric synthetic scoping study revealed seven main research themes concerned with the Scrum research.

Keywords: Software development, Scrum, Bibliometrics, Bibliometric profile, Thematic analysis, Content analysis.

Keywords: Software development, Scrum, Bibliometrics, Thematic analysis, Content analysis.

INTRODUCTION

Introduction of the agile approach especially Scrum into software engineering has changed the way software is being developed. The Scrum approach emphasizes the active end-user involvement, embracing of change, and evolutionary/iterative delivery of products. Among agile methods Scrum seems to be employed in most countries and by most of software development companies [1]. Scrum is not only used in software development. It is implemented as a strategy teamwork and also as a teaching tool in educational institutions [2], and in agile maturity model research [3]. Historically Scrum research has its early roots in software engineering publications introducing innovative solutions to management, and development of software products [4]–[6]. Later historical roots were concerned with Scrum development process [7], software development management with Scrum [8] and empirical research of Agile software development [9].

Over the course of the conceptual evolution, the research on Scrum has attracted considerable attention in almost one and half thousand publications. The research literature has evolved in importance, quantity and complexity. However very few attempts have been made to holistically synthetize the research findings. The few published reviews were devoted to specific Scrum topics like the language used of Scrum communications [10], testing practices in software quality [11] and a bibliometrics study on analysing the research literature of Scrum impact on the productivity and efficiency of software development [12]. . Consequently, there is an extensive body of literature reporting on Scrum research, that has yet to be characterized. To close this gap we performed a study aimed to scope and chart the existing literature on the Scrum research, taking into account Scrum unique characteristics.

Scoping reviews have become increasingly popular as a form of knowledge synthesis [13]. They have great utility for synthesizing research evidence and are often used to categorize or group existing literature in a given field in terms of its nature, features, and volume. Scoping also takes into account the availability of resources. In the fast developing field like software engineering where the knowledge develops extremely rapidly and massively the knowledge synthesis reports have to be produced in a timely manner. To speed the scoping process even further and to enable the analysis of all research publications regarding Scrum, distant reading in combination with bibliometric mapping instead of manual reading of publications [14], [15] was used, thus making scoping process synthetic and semiautomatic. The objective of this synthetic scoping review was to identify and report on the extent literature concerning Scrum research, to map the Scrum research into the landscape of themes and second to identify which methods, tolls, roles, artefacts and objectives emerged in the scope of determined themes.

RESEARCH METHODOLOGY

Scoping reviews can be as a type of research synthesis that aims to 'map the literature on a particular topic or research area and provide an opportunity to identify key concepts; gaps in the research; and types and sources of evidence to inform practice, policymaking, and research' [16] or more simply as a process of mapping the existing literature or evidence [17]. Scoping and systematic reviews are similar in the manner that they both use rigorous and transparent methods to identify and analyse relevant literature to answer research questions [18]. However they also differ considerably in several ways. In scoping reviews the research questions are normally broader, inclusion/exclusion criteria can be developed post-hoc, data extraction is not so detailed and qualitative synthesis is done more frequently than quantitative [19]. Scoping reviews deal with broader topics, might include different types of studies, do not evaluate the quality of the studies, create narratives to sum the results, and consider resources limitations like cost and time [20]. Additionally, data extraction is focused on key general themes. Indeed, the knowledge synthesis is usually expressed as a literature overview from in which general themes are identified [21].

Scoping reviews are performed when there are research gaps in specific areas of research, to disseminate the research results, to clarify a complex concept, as a preliminary step before conducting systematic reviews to explore the extent of literature, and, finally, when informative summary of a research in a specific professional area is needed [17], [22]–[25]. Recently, scoping reviews are also employed for knowledge translation [26]. Arksey and O'Maley [17] proposed a six step methodological framework to perform scoping reviews. This framework was later extended by Levac, Colquhoun & O'Brien [23] and Joanna Briggs Institute [27].

In this study we used the scoping framework proposed by Arksey and O'Malley [17] and the suggestions encased by Levac et al. [23]. The framework was triangulated with bibliometric mapping and thematic analysis to make the scoping semi-automated i.e. synthetic [28], [29].In this manner the review proceeded trough following phases: identifying the research question, harvesting relevant publications, charting and mapping the publications metadata using the above triangulation approach, and collating, summarizing, and reporting the results. The optional 'consultation exercise' was not conducted.

Identifying the Research Question

This review was guided by the question: *What research themes emerge in the Scrum scientific literature and which methods, tolls, roles, artefacts and objectives appear in the context of the themes*.

Harvesting Relevant Publications

After experiment with various search stings the corpus was finally harvested from the Scopus database on 24th of February 2020, using the search string *Scrum AND agile AND software*. Our study was focused more on the basic research and not so much of the application of Scrum in other disciplines Tso we limited the search to the subject area *Computer Science*. No further limitations i.e. on date, language, or publication type were set. The advantage synthetic scoping is also to enable that no traditional eligibility criteria need to be set, ineligible terms or author keywords are excluded during the thematic analysis. Hence, they don't appear in charting and mapping activities and don't influence the results.

Charting and Mapping the Publications Metadata

The publications metadata was analysed first to determine the extent of the literature production in the manner to identify most prolific countries, institutions and source titles, and literature production dynamics. For the bibliometric mapping we used the VOSViewer software [30]. VOSviewer has been used to create bibliometric maps (also called science landscapes) in various studies.[31] The VOSviewer software has powerful visualization capabilities, meaning that resulting can be presented in various forms of bibliometric maps, consequently emphasising different aspects of the analysed research literature like term cluster landscapes, collaboration or citation networks, bibliometric coupling and similar. VOSviewer first uses text mining functions to identify relevant text elements than analyses them using an unified approach for both mapping and clustering. The approach is based on a normalized term co − occurrence matrix and a similarity measure which calculates association strength between terms [32]. In this manner, the VOSviewer software merges text elements that are closely associated into clusters. The proximity of the text element can be interpreted as an indication of their similarity. VOSviewer. The popularity of text element is indicated by the size of the element in the map.

Using a customized Thesaurus file, we excluded the common terms like study, significance, sample, baseline, group, experiment and eliminated geographical names and time stamps. The cluster landscapes were used to identify the themes of Scrum research employing thematic analysis [33]. The terms and keywords were used as codes during thematic analysis [34]. First, the term cluster landscape was analysed, and the clusters denoted with appropriate themes. Next the similar analysis was performed for the keyword cluster landscape, and the results of both analyses were collated. During the next step both landscapes were further analysed to determine which methods, tolls, roles, artefacts and objectives were used/targeted in each of the determined themes.

Finally, the most cited papers for the determined methods, tolls, roles, artefacts, and objectives per theme were identified using the citation data from Scopus. The results were collated and summarised in a tabular form. The hot topics were identified by an approach devised by Kokol at al [35].

RESULTS

The search resulted in 1279 papers. Among them there were 949 conference papers, 217 original articles, 60 conference reviews, 33 book chapters, 11 reviews, five books and four other types of papers. The first two papers indexed in Scopus were publish in 2002 and were associated with integrating agile programming in software engineering courses [36] and the use of Extreme programming in web application development [37].

The dynamics of the overall literature production (Figure 1), shows that in the incubation period (2002 − 2005) the trend in the number of publications was positive, but slow. After 2005 the steeper began, lasting till 2016, when a small decrease in productivity could be observed. In 2017 the growth begun again reaching the peak in 2019 with 162 publications. The average growth was 11.1 publications per year. Regarding journals the

more intensive growth in productivity began after 2010, but the trend was irregular. The peak was reached in 2019 with 26 articles. Most of the publications appeared in conference proceedings, which may imply that the Scrum research didn't yet establish a list of core journals to report research results. On the other hand in the fast development field like Scrum based software development, the conferences might be a more appropriate venue for exchange and maturation of ideas.
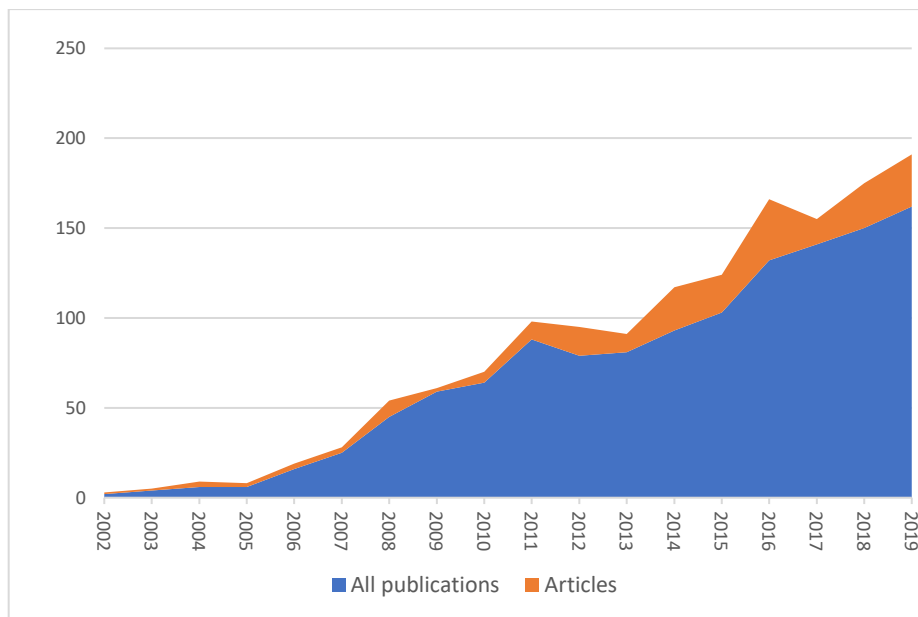

Figure 1 The dynamics of research literature production

Extent of the Literature Production

As stated above, much of the research on Scrum was presented on conferences and published in conference proceedings. The most productive conferences were Lecture Notes In Business Information Processing (n=98), Lecture Notes In Computer Science Including Subseries Lecture Notes In Artificial Intelligence and Lecture Notes In Bioinformatics (n=90), ACM International Conference Proceeding Series (n=72), Communications in Computer And Information Science (n=57), Advances In Intelligent Systems And Computing (n=42), Ceur Workshop Proceedings (n=30), Proceedings International Conference On Software Engineering (n=27), Procedia Computer Science (n=15), Proceedings Frontiers In Education Conference Fie (n=10) and Conference On Human Factors in Computing Systems Proceedings (n=5). Most of the above proceedings are from computers science or related fields. The publisher are internationally recognised entities and the proceedings are highly cited, mostly they have the H Index above 40. The highest H Index was reached by the Lecture Notes In Computer Science with H Index = 324.

The most prolific journals were IEEE Software (n=21), Information And Software Technology (n=20), Journal Of Systems And Software (n=19), Empirical Software Engineering (n=10), Journal Of Software Evolution And Process (n=10), Crosstalk (n=7), International Journal Of Advanced Computer Science And Applications (n=7), Journal Of Theoretical And Applied Information Technology (n=6) and International Journal of Engineering and Advanced Technology (n=5). The journals are mostly from the Computer science or Information Technology categories. In general their Scimago JCR factors (Scopus, Elsevier, Netherlands) are above 0.2, especially for the top productive journals and they are widely recognised as the top journals in software engineering. The less productive journals are not yet so recognized and have lower, however increasing JCR factors and might become core archival journals for the Scrum publications.

The most productive countries were United States (n=169), Brazil (n=124), Germany (n=108), India (n=76) and Finland (n=64), Sweden (n=51), Norway (n=47), United Kingdom (n=47), Netherlands (n=46) and Spain (n=43). United States is the country producing the most software globally [38]. USA employs the largest number of programmers (cca. four millions) globally and Germany the largest number of programmers in Europe (cca. 850000) [39]. Brazil and India are the best countries in which to outscore software development [40]. USA, Norway, Sweden, Germany and Finland and UK are among the countries where software developers have

highest salaries [41]. Most of the top productive countries in Scrum research are also the most productive countries in the total scientific output i.e. USA is the first, UK third and Germany fifth.

The most productive organisations are Aalto University (n=23), SINTEF Foundation for Scientific and Industrial Research (n=20), Instituto Tecnologico de Aeronautica,(=17), University of Limerick (n=16), SINTEF Digital (n=16), Technical University of Munich (n=15), Lero - The Irish Software Engineering Research Centre (n=14), Fraunhofer Institute for Experimental Software Engineering IESE (n=13), Reykjavik University (n=13), Norges Teknisk-Naturvitenskapelige Universitet (n=12) and Universidade Federal de Pernambuco (n=12). Despite the fact that USA is the most productive country none of the most productive institutions is located there. The most productive institutions are situated in Europe and Brazil. It is interesting to note that three of the most productive institutions are located in Norway. These facts might imply that Scrum research in USA is spread through many institutions while in Europe and Brazil it is mostly done in very strong research centres.

Thematic analysis

Seven clusters emerged in the term cluster landscape (Figure 2), and 10 cluster in the author keywords cluster landscape (Figure 3). During the thematic analysis some of the clusters from the author keywords landscape were fused resulting in seven final themes compiled in the Table 1.Those seven theme are: Improving software processes with education and training; Interacting with users to develop working software, Other software development methods and Scrum, Translation of Scrum development approach, Interactive requirements engineering with user stories, Managing Scrum team in global software development and Scrum activities in every day work. Themes reveal that Scrum research is very wide. The research range from every day software development to the development in global and distributed environments. The research focus is frequently devoted to the topics where Scrum differs from traditional software development approaches. A lot of research is also dedicated to how to incorporate the Scrum into traditional software engineering courses and how to translate Scrum paradigm into practice. The study also revealed that Scrum research is frequently performed in the context of other software development approaches, either to compare Scrum with other approaches or to present frameworks how Scrum can be used in combination with other development approaches to improve software development process.

Quantitatively the most publications were concerned with the research on how first to interact with the users and stakeholders aiming to develop the best possible software system and second to translate the Scrum into practice. This research in general followed the directions set in the Agile manifesto [42]. Less research was done on how to introduce Scrum into the software engineering curricula and courses. The researchers mainly focused on how to teach and train Scrum users outside the traditional educational systems.

During the analysis of which methods, tolls, roles, artefacts and objectives were used/targeted in single themes (Table 2) we encountered that Scrum has many variants (or is used in combination with other methods), It is used in combination with other agile approaches, like Feature driven development [43], Xtreme programming [37] or Crystal Clear [44]. Additionally, it is combined with more traditional and systemic software development approaches like Dynamic System Development Method [45], Rational Unified Process [46] or Lean Development [47]. Some tools not normally used in conventional software approaches, like gamification, content analysis and grounded theory are also employed during software development. However, Scrum like other software development approach focuses on improvement of software process, software quality, business value, performance, usability and efficiency and at the same time to reduce cost, risk and uncertainty. Contrary to some conventional approaches it also strives to boost soft factors like agility, trust, motivation, responsibility and transparency.

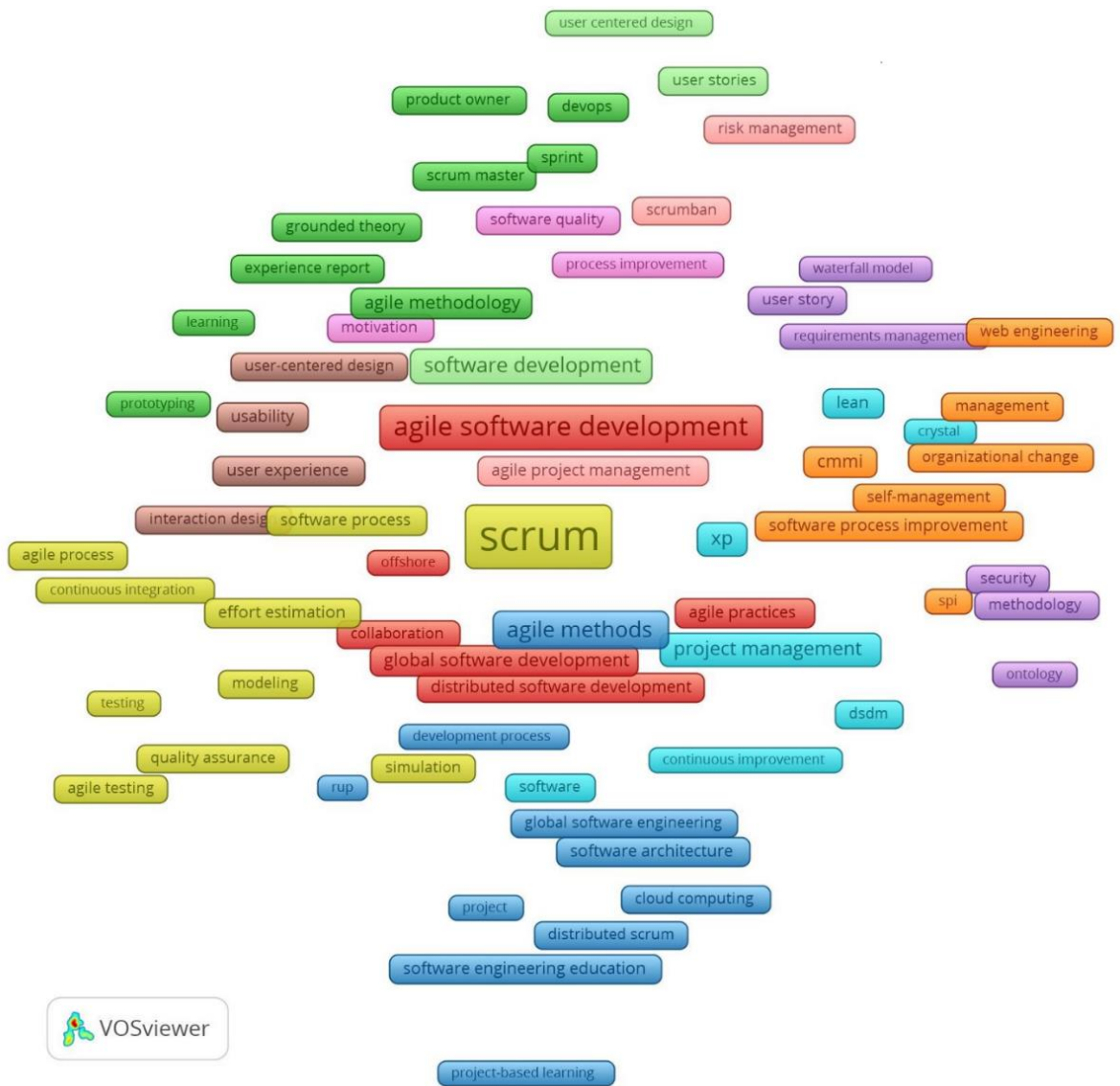*Figure 2. The term cluster landscape of the Scrum research*

Figure 3. The author keyword cluster landscape of the Scrum research

Table 1 Themes of Scrume research

Theme	No of publications	Landscape	Colour	Representative terms/author keywords used as codes in the thematic analysis
I. Improving software processes with education and training	75	Term	Violet	Teaching, education, training, skill, academia, Software process, software engineer, assessment, academia, learning, training, standard, iso iec
		Keyword	Pink	Process improvement software quality, motivation
II. Interacting with users to develop working software	144	Term	Orange	User, persona, usability, user experience, interaction, agile manifesto, working software
		Keyword	Brown	User centred design, interaction design, user experience, usability
III. Other software development methods and Scrum	275	Term	Yellow	Feature driven development; asd (agile software development), Extreme programming, software development methodology
		Keyword	Light blue	Lean, crystal, XP, dsdm
IV. Translation of Scrum development aproach	287	Term	Green	Business value, stakeholder, software development company, continuous integration, enterprise, cmmi, software development lifecycle, mobile applications
		Keyword	Orange	Organizational change, software process improvement, cmmi, self-management, web engineering
V. Interactive requirements engineering with user stories	345	Term	Light Blue	Agile software development team, customer requirement, user story, conflict, customer satisfaction, budget
		Keyword	Green + violet	Agile methodology, software development, requirement management, user centered design, user story(ies), product owner, scrum muster, sprint, prototyping, learning, experience report, grounded theory, security, ontology
VI. Managing Scrum team in global software development	105	Term	Blue	Scrum team, product owner, scrum master, project manager, team member, coordination, global software development
		Keyword	Red + Rose + Blue	Global software development, global software engineering, agile project management, agile software development, agile methods, distributed software development, distributed Scrum, risk management, collaboration, cloud computing, Scrumban, offshore, project based learning
VII. Scrum activities in every day work	130	Term	Red	Agile environment, scrum framework, infrastructure, business, software team, product backlog, client, daily meeting, code, trust, barrier, flexibility
		Keyword	Yellow	Software process, agile process, scrum, effort estimation, modelling, simulation, agile testing, continuous integration

The analysis of the most prolific papers for the methods, tools, roles, artefacts and objectives per theme resulted in 67 publications from the period 2007 to 2019. (Table 2). According to the publishing years, the most state of the art research is done in the themes related to process improvements and user developer interactions. The research seems to be the most established in the theme related to the use of Scrum in every day practice.

Table 2a Methods. Tools, Roles, Objectives and Artefacts with most prolific publications

Theme	Methods	Tools	Roles	Objectives	Artefacts
I	Lean software development [48], Distributed Scrum [49]	Gamification [50], learning standards [51], Scrum framework [52]	Software engineer	Software process improvement [50], applicability [53], software quality [54]	software process [55], software model [56], standard [51]
II		Agile manifesto [57], user centered design [57], interaction design [58]	User [57], Persona [59]	Usability [60]	Working software [58]
III	FDD [61], Crystal Clear [44], DSDM [45], Lean [62], Extreme programming [61], RUP [46], Kanban [1]	Effort estimation [63], global software engineering [64],		Iterative development [65], uncertainty [66]	Software system
IV	Agile techniques [67]	Content analysis [68], knowledge management [69], guidance [70], CMMI [71], risk management [72], automation [47], cloud computing [73], self management [74], web engineering [75]	Project team [76] , software developer [77]	Transparency [78], business value, functionality, dependency [79]	Software
V	Scrum framework [72]	User stories [80], failure prevention [81], user centered design [82], ontology [55], knowledge management [83]	Agile software development team [84]	Security [85], compliance [86], budget [87], customer satisfaction [88]	Software product
VI	SDLC [89], agile methodology [90]	Backlog [91], review [92] , coordination [93], continuous improvement [94], sprint [95], prototyping [96],, empirical studies [97], teamwork [98]	Scrum team [98], product owner [88], Scrum master [99], architect [100], project manager [101], team member [101] expert [102]	Responsibility [103], success factor [81], product quality [104], performance [95], agility [105]	Scrum project [101]
VII	Scrumban, COBIT [106], DevOps [107]	Pair programming [108], agile project management and development [109], Scrum framework [110], simulation, risk management [109], continuous integration [107], modelling [62], agile testing [111], quality assurance [112], effort estimation [113]	Software team [101], employee	Adaptability [89], flexibility [114], trust [115], efficiency [113]	

DISCUSSION

Our study showed that Scrum research is focusing on process improvement, user centred development, development of projects of any size (small to global), how to make development and management flexible and adaptive, roles of developers and users in the development process, how to translate Scrum into traditional software development organisations, how to educate Scrum users and how to use "untraditional methods" like grounded theory, games or content analysis to improve software processes. Those results are confirmed by review papers concerned with the agile development in general *[116]–[118]*. Contrary to our findings some author question the scalability of Scrum to large or very large projects *[119]–[121]*.

One of the more interesting aspects of the Scrum compared to the traditional approaches is the use of gaming in software development. Gaming is used during various activities, especially interesting is the application of gaming in the so called retrospectives – meetings devoted to process improvements [122]. During this activity games are used for data gathering, risk management, team cohesion, timeline assessment and priority of ongoing activities [123]. Another interesting approach to improve the Scrum processes not normally used in the traditional software engineering is the Grounded theory. It is used to support transition from a traditional software development to the Scrum based software development [124], and how an immature Scrum team can become self-organised [125], [126]. Another untraditional tool emerging in Scrum is the Content analysis which is used to better understand users [127].

While the traditional software engineering methods are mostly used as stand-alone, the Scrum is sometimes used in the combination with other development methods. Such hybrids include approaches where a traditional process model serves as (1) a framework in which several fine-grained Scrum practices are plugged in [128] or the traditional waterfall approach is incorporated into the Scrum to "manage chaos and risk" [129]. Other hybrids employ UML in user stories presentations [130] or use lean principles for continuous agile process improvement [131]. A recent more complex hybrid integrates Scrum, Kanban and Waterfall approaches aiming to support the automation of the software systems development [132]. Similar attempts have been made with Scrumban [133] and ScrumFall [134].

Recent publications also compared Scrum with other development approaches. An interesting comparison shows that Scrum, Kanban and Extreme programming requirements election is more effective comparing to other agile approaches, which often still adopted the traditional requirement process [135]. Another study showed that user experience is better if using Kanban than Scrum [136]. A recent study shows that synthesising Lean software development and distributed Scrum led to technical advances in system integration of network software systems [137]. Studies comparing Scrum and Waterfall revelled that using Scrum is preferable when issues should be fixed in a short period of time, accountability should be on very high level, when projects are not too large and when all the details are not known in the beginning of the projects, which all reults in higher Scrum execution success rates [138].

Few universities started including Scrum approach in their curricula [139], hence most of the Scrum education and training is done outside the regular university studies. Additionally, software engineering is a continuous process, which should incorporate needs of the software industry, thus virtual environments can serve as an appropriate training platform [140]. Other interesting training platforms are the gamification [141], [142], LEGO blocks [143], [144], doing real-life projects [145] and using learning platforms [54].

Our study also identified possible future directions in Scrum research. Collaborative games could be used first to foster the active user involvement in requirements engineering to envision new business and technical opportunities and shape the solutions [146] and second to revitalize the retrospective meetings [147]. Inter-team coordination employing a hybrid of agile and traditional planning tools could be used in large scale Scrum projects [148]. Similarly, knowledge management might be used in very large scale Scrum projects for knowledge transfer between teams [149] and to prevent knowledge loss [150]. The challenges of global software development [151] like self organization between remote teams [64] and bridging geographical, socio-cultural and temporal boundaries [152] will be tackled with the use of distributed software development [153] and model driven software architectures [154]. One of the future challenges is also how to trace non-functional requirement and the solution might be the integration of Feature Driven Development and Scrum [155]. User stories are a core approach in Scrum requirements engineering and it seems that future research will also be directed to them. How to estimate the needed resources based on the complexity of the user stories [156], conceptualisation of the consensus procedures to determine when a story is completed [80], reducing the ambiguity caused by using natural language in user stories [55] and deriving user for distributed Scrum [157] might be some of the challenges. Research on use of machine learning support to Scrum development will probably also receive more attention, for example in cost estimation [158], automatic testing [159] and general resources estimation [160].

Challenges and limitations

First, we used just one bibliographic database, namely Scopus the Elsevier's abstract and citation database of peer-reviewed literature. The reason was that Scopus is the largest of that kind of databases containing scientific journals, books and conference proceedings. It delivers a comprehensive overview of the world's research output in the fields of science including technology [161]. To maintain a certain degree of the quality of the corpus publication we didn't include the grey literature, which is mainly not peer-reviewed.

Second, to maintain broadness required for scoping we created a comprehensive database of existing literature on Scrum research, hence the volume of literature was to large to chart the publications in more depth. Consequently, we opted to produce a tabulated map of most prolific publications.

Finally, thematic analysis is a qualitative approach, thus susceptible to bias. While the authors done the study as objectively as possible, other authors might come to different results.

On the other hand our study is a first comprehensive bibliometric study focusing on solely on Scrum research. Additionally, the study was performed with a novel triangulating approach, combining bibliometric mapping and thematic analysis.
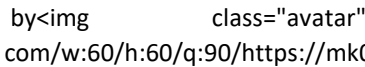
CONCLUSION

The synthetic scoping review presents a holistic view on the Scrum research, as revealed by the bibliometric and bibliometric mapping based analysis of Scrum related publications indexed in the Scopus database. The review can help researchers and practitioners to understand the broader aspects of Scrum` research and its translation to practice. On the other hand, it can inform a novice, interested reader or software development professional without specific knowledge on Scrum to develop a perspective on the most important research themes, methods, tools and benefits of using Scrum. Our scoping review has also produced a comprehensive literature map, which points interested readers to most prolific publications.

REFERENCES

[1]   T. Dingsoeyr, D. Falessi, and K. Power, "Agile Development at Scale: The Next Frontier," *IEEE Software*, vol. 36, no. 2, pp. 30–38, 2019, doi: 10.1109/MS.2018.2884884.

[2]   J. L. Onieva López, "SCRUM as a strategy for collaborative learning through projects. Didactic proposal for implementation in the university classroom," *Profesorado*, vol. 22, no. 2, pp. 467–485, 2018.

[3]   V. Henriques and M. Tanner, "A systematic literature review of agile and maturity model research," *Interdisciplinary Journal of Information, Knowledge, and Management*, vol. 12, pp. 53–73, 2017.

[4]   W. W. Royce, "Managing the Development of Large Software Systems.," in *Proceedings of IEEE WESCON*, 1970, vol. 26.

[5]   B. W. Boehm, "Software Engineering Economics," *IEEE Transactions on Software Engineering*, vol. SE-10, no. 1, pp. 4–21, Jan. 1984, doi: 10.1109/TSE.1984.5010193.

[6]   H. Takeuchi and I. Nonaka, "The New New Product Development Game," *Harvard Business Review*, no. January 1986, Jan. 01, 1986.

[7]   K. Schwaber, "SCRUM Development Process," in *Business Object Design and Implementation*, London, 1997, pp. 117–134, doi: 10.1007/978-1-4471-0947-1_11.

[8]   K. Schwaber, *Agile Project Management with Scrum*. Microsoft Press, 2004.

[9]   T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: A systematic review," *Information and Software Technology*, vol. 50, no. 9, pp. 833–859, Aug. 2008, doi: 10.1016/j.infsof.2008.01.006.

[10]   E. Friess, "Scrum language use in a software engineering firm: An exploratory study," *IEEE Transactions on Professional Communication*, vol. 62, no. 2, pp. 130–147, 2019, doi: 10.1109/TPC.2019.2911461.

[11] C. Gil, J. Diaz, M. Orozco, A. de la Hoz, E. de la Hoz, and R. Morales, "Agile testing practices in software quality: State of the art review," *Journal of Theoretical and Applied Information Technology*, vol. 92, no. 1, pp. 28–36, 2016.

[12] T. G. Lechler and S. Yang, "Exploring the Role of Project Management in the Development of the Academic Agile Software Discourse: A Bibliometric Analysis," *Project Management Journal*, vol. 48, no. 1, pp. 3–18, 2017, doi: 10.1177/875697281704800101.

[13] Z. Munn, M. D. J. Peters, C. Stern, C. Tufanaru, A. McArthur, and E. Aromataris, "Systematic review or scoping review? Guidance for authors when choosing between a systematic or scoping review approach," *BMC Medical Research Methodology*, vol. 18, no. 1, p. 143, Nov. 2018, doi: 10.1186/s12874-018-0611-x.

[14] F. Moretti, *Distant Reading*. London, New York: Verso, 2013.

[15] P. Kokol, K. Saranto, and H. Blažun Vošner, "eHealth and health informatics competences: A systemic analysis of literature production based on bibliometrics," *Kybernetes*, vol. 47, no. 5, pp. 1018–1030, Jan. 2018, doi: 10.1108/K-09-2017-0338.

[16] H. M. Daudt, C. van Mossel, and S. J. Scott, "Enhancing the scoping study methodology: a large, inter-professional team's experience with Arksey and O'Malley's framework," *BMC Medical Research Methodology*, vol. 13, no. 1, p. 48, Mar. 2013, doi: 10.1186/1471-2288-13-48.

[17] H. Arksey and L. O'Malley, "Scoping studies: towards a methodological framework," *International Journal of Social Research Methodology*, vol. 8, no. 1, pp. 19–32, Feb. 2005, doi: 10.1080/1364557032000119616.

[18] M. T. Pham, A. Rajić, J. D. Greig, J. M. Sargeant, A. Papadopoulos, and S. A. McEwen, "A scoping review of scoping reviews: advancing the approach and enhancing the consistency," *Res. Syn. Meth.*, vol. 5, no. 4, pp. 371–385, Dec. 2014, doi: 10.1002/jrsm.1123.

[19] S. E. Brien, D. L. Lorenzetti, S. Lewis, J. Kennedy, and W. A. Ghali, "Overview of a formal scoping review on health system report cards," *Implementation Sci*, vol. 5, no. 1, p. 2, Jan. 2010, doi: 10.1186/1748-5908-5-2.

[20] P. Sarrami-Foroushani, J. Travaglia, D. Debono, R. Clay-Williams, and J. Braithwaite, "Scoping Meta-Review: Introducing a New Methodology," *Clinical and Translational Science*, vol. 8, no. 1, pp. 77–81, 2015, doi: 10.1111/cts.12188.

[21] H. Sucharew and H. Sucharew, "Methods for Research Evidence Synthesis: The Scoping Review Approach," *Journal of Hospital Medicine*, vol. 14, no. 7, Jul. 2019, doi: 10.12788/jhm.3248.

[22] R. Armstrong, B. J. Hall, J. Doyle, and E. Waters, "Cochrane Update. 'Scoping the scope' of a cochrane review.," *Journal of public health (Oxford, England)*, vol. 33, no. 1, pp. 147–150, 2011.

[23] D. Levac, H. Colquhoun, and K. K. O'Brien, "Scoping studies: advancing the methodology," *Implementation Science*, vol. 5, p. 69, Sep. 2010, doi: 10.1186/1748-5908-5-69.

[24] J. Peterson, P. F. Pearce, L. A. Ferguson, and C. A. Langford, "Understanding scoping reviews: Definition, purpose, and process," *Journal of the American Association of Nurse Practitioners*, vol. 29, no. 1, pp. 12–16, 2017, doi: 10.1002/2327-6924.12380.

[25] A. C. Tricco *et al.*, "A scoping review on the conduct and reporting of scoping reviews," *BMC Medical Research Methodology*, vol. 16, no. 1, 2016, doi: 10.1186/s12874-016-0116-4.

[26] A. Schultz *et al.*, "A scoping approach to systematically review published reviews: Adaptations and recommendations," *Research Synthesis Methods*, vol. 9, no. 1, pp. 116–123, 2018, doi: 10.1002/jrsm.1272.

[27] M. Peters, C. Godfrey, H. Khalil, P. Mcinerney, C. Soares, and D. Parker, "2017 Guidance for the Conduct of JBI Scoping Reviews," 2017.

[28] H. Blažun Vošner, D. Železnik, P. Kokol, J. Vošner, and J. Završnik, "Trends in nursing ethics research: Mapping the literature production," *Nurs Ethics*, vol. 24, no. 8, pp. 892–907, Dec. 2017, doi: 10.1177/0969733016654314.

[29] P. Kokol and H. Blažun Vošner, "Historical, descriptive and exploratory analysis of application of bibliometrics in nursing research," *Nursing Outlook*, May 2019, doi: 10.1016/j.outlook.2019.04.009.

[30] N. J. van Eck and L. Waltman, "Visualizing Bibliometric Networks," in *Measuring Scholarly Impact: Methods and Practice*, Y. Ding, R. Rousseau, and D. Wolfram, Eds. Cham: Springer International Publishing, 2014, pp. 285–320.

[31] "VOSviewer - Publications," *VOSviewer*. https://www.vosviewer.com/publications/ (accessed Apr. 15, 2020).

[32] N. J. van Eck, L. Waltman, E. C. M. Noyons, and R. K. Buter, "Automatic term identification for bibliometric mapping," *Scientometrics*, vol. 82, no. 3, pp. 581–596, Mar. 2010, doi: 10.1007/s11192-010-0173-0.

[33] M. Vaismoradi, H. Turunen, and T. Bondas, "Content analysis and thematic analysis: Implications for conducting a qualitative descriptive study," *Nurs Health Sci*, vol. 15, no. 3, pp. 398–405, Sep. 2013, doi: 10.1111/nhs.12048.

[34] P. Kokol, H. Blažun Vošner, and D. Železnik, "Clinical Simulation in Nursing: A Bibliometric Analysis after Its Tenth Anniversary," *Clinical Simulation In Nursing*, vol. 13, no. 4, pp. 161–167, Apr. 2017, doi: 10.1016/j.ecns.2016.11.007.

[35] P. Kokol, J. Završnik, and H. Blažun Vošner, "Bibliographic-Based Identification of Hot Future Research Topics: An Opportunity for Hospital Librarianship," *Journal of Hospital Librarianship*, vol. 0, no. 0, pp. 1–8, Sep. 2018, doi: 10.1080/15323269.2018.1509193.

[36] L. Williams, M. Lutz, G. Hislop, M. McCracken, N. Mead, and J. F. Naveda, "Integrating agile practices into software engineering courses," presented at the Software Engineering Education Conference, Proceedings, 2002, vol. 2002-January, pp. 242–243, doi: 10.1109/CSEE.2002.995222.

[37] F. Maurer and S. Martel, "Extreme programming: Rapid development for web-based applications," *IEEE Internet Computing*, vol. 6, no. 1, pp. 86–90, 2002, doi: 10.1109/4236.989006.

[38] S. Seth, "World's Top 10 Software Companies," *Investopedia*. https://www.investopedia.com/articles/personal-finance/121714/worlds-top-10-software-companies.asp (accessed Nov. 08, 2019).

[39] "Software Developer Statistics 2019: How Many Software Engineers Are in the US and in the World?," *Daxx Software Development Teams*, May 19, 2019. https://www.daxx.com/blog/development-trends/number-software-developers-world (accessed Feb. 07, 2020).

[40] by D. Gunn, "10 Best Countries to Outsource Software Development, Based on Data," *CodeinWP*, May 14, 2019. https://www.codeinwp.com/blog/best-countries-to-outsource-software-development/ (accessed Nov. 08, 2019).

[41] techwriting, "Highest Paying Countries For Software Engineer," *Tech Writing*, Jan. 26, 2020. https://writingtech.wixsite.com/writer/post/highest-paying-countries-for-software-engineer (accessed Apr. 19, 2020).

[42] "Principles behind the Agile Manifesto." https://agilemanifesto.org/principles.html (accessed Apr. 20, 2020).

[43] A. Sharma and R. K. Bawa, "A roadmap for agility estimation and method selection for secure agile development using AHP and ANN," *Advances in Intelligent Systems and Computing*, vol. 542, pp. 237–245, 2018, doi: 10.1007/978-981-10-3223-3_22.

[44]   L. Zamudio, J. A. Aguilar, C. Tripp, and S. Misra, "A requirements engineering techniques review in agile software development methods," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10408 LNCS, pp. 683–698, 2017, doi: 10.1007/978-3-319-62404-4_50.

[45]   A. Delgado and P. Condori, "Comparative Study of Methods to Improve Administrative Processes in a Organization," presented at the 2018 Congreso Internacional de Innovacion y Tendencias en Ingenieria, CONIITI 2018 - Proceedings, 2018, doi: 10.1109/CONIITI.2018.8587055.

[46]   M. Tanveer, "Agile for large scale projects - A hybrid approach," presented at the 2015 National Software Engineering Conference, NSEC 2015, 2016, pp. 14–18, doi: 10.1109/NSEC.2015.7396338.

[47]   A. M. Redmond and A. Zarli, "The concept selection of lean software and system engineering tools for smart cities," presented at the 2017 International Conference on Engineering, Technology and Innovation: Engineering, Technology and Innovation Management Beyond 2020: New Challenges, New Approaches, ICE/ITMC 2017 - Proceedings, 2018, vol. 2018-January, pp. 36–43, doi: 10.1109/ICE.2017.8279866.

[48]   J. R. Neve, K. Godbole, and R. Neve, "Productivity and process improvement using 'Scaled Agile' approaches: An emphasized analysis," presented at the Proceedings of the International Conference on Inventive Computing and Informatics, ICICI 2017, 2018, pp. 793–798, doi: 10.1109/ICICI.2017.8365245.

[49]   M. Paasivaara, "Adopting SAFe to scale agile in a globally distributed organization," presented at the Proceedings - 2017 IEEE 12th International Conference on Global Software Engineering, ICGSE 2017, 2017, pp. 36–40, doi: 10.1109/ICGSE.2017.15.

[50]   B. R. Maxim, R. Kaur, C. Apzynski, D. Edwards, and E. Evans, "An agile software engineering process improvement game," presented at the Proceedings - Frontiers in Education Conference, FIE, 2016, vol. 2016-November, doi: 10.1109/FIE.2016.7757682.

[51]   F. M. Soto-Guerrero, P. A. Quezada-Sarmiento, L. C. Condolo-Herrera, S. Mengual-Andrés, D. I. Moreno-León, and I. L. Rey-Mendoza, "Development of a methodological framework for educational innovation in the context of the tutorial action based on agile methodologies and software engineering knowledge standards," *RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao*, no. E15, pp. 233–242, 2018.

[52]   R. T. De Souza, S. D. Zorzo, and D. A. Da Silva, "Evaluating capstone project through flexible and collaborative use of Scrum framework," presented at the Proceedings - Frontiers in Education Conference, FIE, 2015, vol. 2014, doi: 10.1109/FIE.2015.7344249.

[53]   J. C. Guzmán, G. López, and B. Aymerich, "Agile for newbies: A scrum workshop design and implementation process," *Advances in Intelligent Systems and Computing*, vol. 785, pp. 443–451, 2019, doi: 10.1007/978-3-319-93882-0_42.

[54]   W.-L. Lee, "SCRUM-X: An interactive and experiential learning platform for teaching Scrum," presented at the 7th International Multi-Conference on Complexity, Informatics and Cybernetics, IMCIC 2016 and 7th International Conference on Society and Information Technologies, ICSIT 2016 - Proceedings, 2016, vol. 2, pp. 192–197.

[55]   P. L. de Souza, A. F. do Prado, W. L. de Souza, S. M. dos Santos Forghieri Pereira, and L. F. Pires, "Improving Agile Software Development with Domain Ontologies," *Advances in Intelligent Systems and Computing*, vol. 738, pp. 267–274, 2018, doi: 10.1007/978-3-319-77028-4_37.

[56]   B. Bruegge, S. Krusche, and L. Alperowitz, "Software engineering project courses with industrial clients," *ACM Transactions on Computinig Education*, vol. 15, no. 4, 2015, doi: 10.1145/2732155.

[57]   D. Teka, Y. Dittrich, and M. Kifle, "Adapting lightweight user-centered design with the scrum-based development process," presented at the Proceedings - International Conference on Software Engineering, 2018, pp. 35–42, doi: 10.1145/3195528.3195530.

[58] V. Holzmann and I. Panizel, "Communications management in scrum projects," presented at the 7th European Conference on Information Management and Evaluation, ECIME 2013, 2013, pp. 67–74.

[59] J. Haikara, "Usability in agile software development: Extending the interaction design process with personas approach," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 4536 LNCS, pp. 153–156, 2007.

[60] M. Larusdottir, Å. Cajander, J. Gulliksen, G. Cockton, P. Gregory, and D. Salah, "On the integration of user centred design in agile development," presented at the Proceedings of the NordiCHI 2014: The 8th Nordic Conference on Human-Computer Interaction: Fun, Fast, Foundational, 2014, pp. 817–820, doi: 10.1145/2639189.2654836.

[61] T. Sharma, M. Fragkoulis, and D. Spinellis, "House of Cards: Code Smells in Open-Source C# Repositories," presented at the International Symposium on Empirical Software Engineering and Measurement, 2017, vol. 2017-November, pp. 424–429, doi: 10.1109/ESEM.2017.57.

[62] A. Srivastava, P. K. Kapur, D. Mehrotra, and R. Majumdar, "Modelling fault detection using SRGM in Agile environment and ranking of models," *Journal of Cases on Information Technology*, vol. 21, no. 2, pp. 1–20, 2019, doi: 10.4018/JCIT.2019040101.

[63] M. Arora, S. Verma, and Kavita, "An efficient effort and cost estimation framework for Scrum Based Projects," *International Journal of Engineering and Technology(UAE)*, vol. 7, no. 4.12 Special Issue 12, pp. 52–57, 2018.

[64] R. Vallon, B. J. da Silva Estácio, R. Prikladnicki, and T. Grechenig, "Systematic literature review on agile practices in global software development," *Information and Software Technology*, vol. 96, pp. 161–180, 2018, doi: 10.1016/j.infsof.2017.12.004.

[65] T. V. Ribeiro, C. D. F. Souza, and H. A. T. Leão, "Sidd - scrum iteration driven development: An agile software development and management process based on scrum," presented at the Proceedings of the International Conference on Software Engineering and Knowledge Engineering, SEKE, 2018, vol. 2018-July, pp. 502–505, doi: 10.18293/SEKE2018-102.

[66] R. Popli and N. Chauahn, "Managing uncertainty of story-points in agile software," presented at the 2015 International Conference on Computing for Sustainable Global Development, INDIACom 2015, 2015, pp. 1357–1361.

[67] J. Balakrishnan, "Design of a framework to implement agility at organizational level," in *Emerging Innovations in Agile Software Development*, 2016, pp. 127–140.

[68] M. Alqudah and R. Razali, "An empirical study of Scrumban formation based on the selection of scrum and Kanban practices," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 8, no. 6, pp. 2315–2322, 2018.

[69] G. A. Ruiz *et al.*, "Using meta-ethnography to synthesize research on knowledge management and agile software development methodology," presented at the ACM International Conference Proceeding Series, 2018, pp. 230–239, doi: 10.1145/3275245.3275270.

[70] S. W. Ambler and M. Lines, "The disciplined agile process decision framework," *Lecture Notes in Business Information Processing*, vol. 238, pp. 3–14, 2016, doi: 10.1007/978-3-319-27033-3_1.

[71] B. Yalçıner and K. Dinçer, "Mapping scrum method building elements to CMMI level 2 requirements," presented at the CEUR Workshop Proceedings, 2018, vol. 2201.

[72] M. Mousaei and T. J. Gandomani, "A new project risk management model based on Scrum framework and Prince2 methodology," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 4, pp. 442–449, 2018, doi: 10.14569/IJACSA.2018.090461.

[73] S. Stavru, I. Krasteva, and S. Ilieva, "Challenges for migrating to the service cloud paradigm: An agile perspective," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial*

Intelligence and Lecture Notes in Bioinformatics), vol. 7652 LNCS, pp. 77–91, 2013, doi: 10.1007/978-3-642-38333-5_10.

[74]   G. Gutierrez, J. Garzas, M. T. G. De Lena, and J. M. Moguerza, "Self-Managing: An Empirical Study of the Practice in Agile Teams," *IEEE Software*, vol. 36, no. 1, pp. 23–27, 2019, doi: 10.1109/MS.2018.2874324.

[75]   C. J. Torrecilla-Salinas, T. Guardia, O. De Troyer, M. Mejías, and J. Sedeño, "NDT-Agile: An agile, CMMI-Compatible framework for web engineering," *Communications in Computer and Information Science*, vol. 770, pp. 3–16, 2017, doi: 10.1007/978-3-319-67383-7_1.

[76]   R. Marques, G. Costa, M. Mira Da Silva, and P. Gonçalves, "Gamifying software development scrum projects," presented at the 2017 9th International Conference on Virtual Worlds and Games for Serious Applications, VS-Games 2017 - Proceedings, 2017, pp. 141–144, doi: 10.1109/VS-GAMES.2017.8056584.

[77]   J.-P. Koster and R. Baardman, "Agile needs FBM," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11231 LNCS, pp. 126–134, 2019, doi: 10.1007/978-3-030-11683-5_13.

[78]   J. Lopez-Martinez, R. Juarez-Ramirez, C. Huertas, S. Jimenez, and C. Guerra-Garcia, "Problems in the adoption of agile-scrum methodologies: A systematic literature review," presented at the Proceedings - 2016 4th International Conference in Software Engineering Research and Innovation, CONISOFT 2016, 2016, pp. 141–148, doi: 10.1109/CONISOFT.2016.30.

[79]   A. Zia, W. Arshad, and W. Mahmood, "Preference in using agile development with larger team size," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 7, pp. 116–123, 2018, doi: 10.14569/IJACSA.2018.090716.

[80]   M. Ormsby and C. Busby-Earle, "A Standardized Procedure to Conceptualizing and Completing User Stories," presented at the Proceedings - 2017 International Conference on Computational Science and Computational Intelligence, CSCI 2017, 2018, pp. 934–939, doi: 10.1109/CSCI.2017.162.

[81]   S. Dhir, D. Kumar, and V. B. Singh, "Success and failure factors that impact on project implementation using agile software development methodology," *Advances in Intelligent Systems and Computing*, vol. 731, pp. 647–654, 2019, doi: 10.1007/978-981-10-8848-3_62.

[82]   P. Lucena, A. Braz, A. Chicoria, and L. Tizzei, "IBM design thinking software development framework," *Communications in Computer and Information Science*, vol. 680, pp. 98–109, 2017, doi: 10.1007/978-3-319-55907-0_9.

[83]   J. Sedeño, E.-M. Schön, C. Torrecilla-Salinas, J. Thomaschewski, M. J. Escalona, and M. Mejias, "Modelling agile requirements using context-based persona stories," presented at the WEBIST 2017 - Proceedings of the 13th International Conference on Web Information Systems and Technologies, 2017, pp. 196–203.

[84]   Z. Wang, "Teamworking strategies of scrum team: A multi-agent based simulation," presented at the ACM International Conference Proceeding Series, 2018, pp. 404–408, doi: 10.1145/3297156.3297179.

[85]   L. Benedicenti, A. Messina, and A. Sillitti, "Iagile: Mission critical military software development," presented at the Proceedings - 2017 International Conference on High Performance Computing and Simulation, HPCS 2017, 2017, pp. 545–552, doi: 10.1109/HPCS.2017.87.

[86]   T. Kowark, F. Hager, R. Gehrer, and J. Krüger, "A research plan for the integration of design thinking with large scale software development projects," in *Design Thinking Research: Building Innovation Eco-Systems*, 2014, pp. 183–202.

[87]   S. Andrzeevski, "The main challenges of implementing fixed-price agile projects for corporate customers," presented at the 2010 6th Central and Eastern European Software Engineering Conference, CEE-SECR 2010, 2010, pp. 220–225, doi: 10.1109/CEE-SECR.2010.5783180.

[88]   J. Choma, L. A. M. Zaina, and D. Beraldo, "UserX story: Incorporating UX aspects into user stories elaboration," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial*

Intelligence and Lecture Notes in Bioinformatics), vol. 9731, pp. 131–140, 2016, doi: 10.1007/978-3-319-39510-4_13.

[89]   R. A. Dos Santos, A. F. Zorzo, and S. Marczak, "Towards an SDLC for projects involving Distributed Systems," presented at the ICEIS 2016 - Proceedings of the 18th International Conference on Enterprise Information Systems, 2016, vol. 1, pp. 158–165.

[90]   S. Galvan, M. Mora, R. V. O'Connor, F. Acosta, and F. Alvarez, "A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process," presented at the Procedia Computer Science, 2015, vol. 64, pp. 188–195, doi: 10.1016/j.procs.2015.08.480.

[91]   V. T. Heikkilä, M. Paasivaara, C. Lasssenius, D. Damian, and C. Engblom, "Managing the requirements flow from strategy to release in large-scale agile development: a case study at Ericsson," *Empirical Software Engineering*, vol. 22, no. 6, pp. 2892–2936, 2017, doi: 10.1007/s10664-016-9491-z.

[92]   M. Drury, K. Conboy, and K. Power, "Obstacles to decision making in Agile software development teams," *Journal of Systems and Software*, vol. 85, no. 6, pp. 1239–1254, 2012, doi: 10.1016/j.jss.2012.01.058.

[93]   G. Arseni, "Role of the design authority in large scrum of scrum multi-team-based programs," *Communications in Computer and Information Science*, vol. 422, pp. 181–189, 2016, doi: 10.1007/978-3-319-27896-4_15.

[94]   M. Paasivaara and C. Lassenius, "Communities of practice in a large distributed agile software development organization - Case Ericsson," *Information and Software Technology*, vol. 56, no. 12, pp. 1556–1577, 2014, doi: 10.1016/j.infsof.2014.06.008.

[95]   J. Sutherland, S. Downey, and B. Granvik, "Shock therapy a bootstrap for hyper-productive scrum," presented at the Proceedings - 2009 Agile Conference, AGILE 2009, 2009, pp. 69–73, doi: 10.1109/AGILE.2009.28.

[96]   L. Alperowitz, C. Scheuermann, and N. Von Frankenberg, "From storyboards to code: Visual product backlogs in agile project courses," presented at the CEUR Workshop Proceedings, 2017, vol. 1790, pp. 69–72.

[97]   V.-P. Eloranta, K. Koskimies, and T. Mikkonen, "Exploring ScrumBut - An empirical study of Scrum anti-patterns," *Information and Software Technology*, vol. 74, pp. 194–203, 2016, doi: 10.1016/j.infsof.2015.12.003.

[98]   C. J. Stettina and W. Heijstek, "Five agile factors: Helping self-management to self-reflect," *Communications in Computer and Information Science*, vol. 172, pp. 84–96, 2011, doi: 10.1007/978-3-642-22206-1.

[99]   P. Ralph and P. Shportun, "Scrum abandonment in distributed teams: A revelatory case," presented at the Proceedings - Pacific Asia Conference on Information Systems, PACIS 2013, 2013.

[100]   L. T. Portela and G. Borrego, "Scrumconix: Agile and documented method to AGSD," presented at the Proceedings - 11th IEEE International Conference on Global Software Engineering, ICGSE 2016, 2016, pp. 195–196, doi: 10.1109/ICGSE.2016.39.

[101]   A. Moran, *Managing agile: Strategy, implementation, organisation and people*. 2015.

[102]   S. N. H. Ghazali, S. S. Salirti, I. Inayat, and S. H. Ab Hamid, "A risk poker based testing model for scrum," *Computer Systems Science and Engineering*, vol. 33, no. 3, pp. 169–185, 2018.

[103]   Å. Cajander, M. Larusdottir, and J. Gulliksen, "Existing but not explicit - The user perspective in scrum projects in practice," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 8119 LNCS, no. PART 3, pp. 762–779, 2013, doi: 10.1007/978-3-642-40477-1_52.

[104]   Y. Zhang and S. Patel, "Agile model-driven development in practice," *IEEE Software*, vol. 28, no. 2, pp. 84–91, 2011, doi: 10.1109/MS.2010.85.

[105]   J. Nakatumba-Nabende, B. Kanagwa, R. Hebig, R. Heldal, and E. Knauss, "Hybrid software and systems development in practice: Perspectives from Sweden and Uganda," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10611 LNCS, pp. 413–419, 2017, doi: 10.1007/978-3-319-69926-4_30.

[106]   N. Özkan, "Scrum integrated SDLC processes of Turkiye Finans IT in a COBIT compliant environment," presented at the CEUR Workshop Proceedings, 2017, vol. 1980, pp. 126–134.

[107]   S. S. Samarawickrama and I. Perera, "Continuous scrum: A framework to enhance scrum with DevOps," presented at the 17th International Conference on Advances in ICT for Emerging Regions, ICTer 2017 - Proceedings, 2018, vol. 2018-January, pp. 19–25, doi: 10.1109/ICTER.2017.8257808.

[108]   Z. Wang, "The Impact of Expertise on Pair Programming Productivity in a Scrum Team: A Multi-Agent Simulation," presented at the Proceedings of the IEEE International Conference on Software Engineering and Service Sciences, ICSESS, 2019, vol. 2018-November, pp. 399–402, doi: 10.1109/ICSESS.2018.8663874.

[109]   S. K. Khatri, K. Bahri, and P. Johri, "Best practices for managing risk in adaptive agile process," presented at the Proceedings - 2014 3rd International Conference on Reliability, Infocom Technologies and Optimization: Trends and Future Directions, ICRITO 2014, 2015, doi: 10.1109/ICRITO.2014.7014759.

[110]   J. B. Barlow *et al.*, "Overview and guidance on agile development in large organizations," *Communications of the Association for Information Systems*, vol. 29, no. 1, pp. 25–44, 2011.

[111]   M. Elallaoui, K. Nafil, R. Touahni, and R. Messoussi, "Automated Model Driven Testing Using AndroMDA and UML2 Testing Profile in Scrum Process," presented at the Procedia Computer Science, 2016, vol. 83, pp. 221–228, doi: 10.1016/j.procs.2016.04.119.

[112]   G. Kumar and P. K. Bhatia, "Comparative analysis of software engineering models from traditional to modern methodologies," presented at the International Conference on Advanced Computing and Communication Technologies, ACCT, 2014, pp. 189–196, doi: 10.1109/ACCT.2014.73.

[113]   F. U. Rehman, B. Maqbool, M. Q. Riaz, U. Qamar, and M. Abbas, "Scrum Software Maintenance Model: Efficient Software Maintenance in Agile Methodology," presented at the 21st Saudi Computer Society National Computer Conference, NCC 2018, 2018, doi: 10.1109/NCG.2018.8593152.

[114]   L. Chen, W. Ye, and S. Zhang, "Onboard: A data-driven agile software development collaboration tool," *Jisuanji Yanjiu yu Fazhan/Computer Research and Development*, vol. 53, no. 12, pp. 2753–2767, 2016, doi: 10.7544/issn1000-1239.2016.20160625.

[115]   R. Phalnikar, V. S. Deshpande, and S. D. Joshi, "Applying agile principles for distributed software development," presented at the Proceedings - International Conference on Advanced Computer Control, ICACC 2009, 2009, pp. 535–539, doi: 10.1109/ICACC.2009.93.

[116]   G. S. Matharu, A. Mishra, H. Singh, and P. Upadhyay, "Empirical Study of Agile Software Development Methodologies: A Comparative Analysis." Association for Computing Machinery, Feb. 06, 2015, Accessed: Apr. 19, 2020. [Online]. Available: https://doi.org/10.1145/2693208.2693233.

[117]   R. Mobashar, A. Aamir, G. Rehman Abdul, and H. Ahmed Manzoor, *Human Factors in Global Software Engineering*. IGI Global, 2019.

[118]   H. Saeeda, J. Dong, Y. Wang, and M. A. Abid, "A proposed framework for improved software requirements elicitation process in SCRUM: Implementation by a real-life Norway-based IT project," *Journal of Software: Evolution and Process*, vol. n/a, no. n/a, p. e2247, 2020, doi: 10.1002/smr.2247.

[119]   G. Brataas, G. K. Hanssen, N. Herbst, and A. van Hoorn, "Agile Scalability Engineering: The ScrumScale Method," *IEEE Software*, pp. 1–1, 2019, doi: 10.1109/MS.2019.2923184.

[120]   N. Ozkan and A. K. Tarhan, "Investigating Causes of Scalability Challenges in Agile Software Development from a Design Perspective," in *2019 1st International Informatics and Software Engineering Conference (UBMYK)*, Nov. 2019, pp. 1–6, doi: 10.1109/UBMYK48245.2019.8965633.

[121] D. Rosenberg, B. Boehm, M. Stephens, C. Suscheck, S. R. Dhalipathi, and B. Wang, "Large-Scale Parallel Development," in *Parallel Agile – faster delivery, fewer defects, lower cost*, D. Rosenberg, B. Boehm, M. Stephens, C. Suscheck, S. R. Dhalipathi, and B. Wang, Eds. Cham: Springer International Publishing, 2020, pp. 153–168.

[122] D. Marshburn and J. P. Sieck, "Dont Break the Build: Developing a Scrum Retrospective Game," Jan. 2019, doi: 10.24251/HICSS.2019.838.

[123] M. Jovanovic, A.-L. Mesquida, and A. Mas, "Process Improvement with Retrospective Gaming in Agile Software Development," in *Systems, Software and Services Process Improvement*, Cham, 2015, pp. 287–294, doi: 10.1007/978-3-319-24647-5_23.

[124] P. J. Carew and D. Glynn, "Anti-patterns in Agile Adoption: A Grounded Theory Case Study of One Irish IT Organisation," *Glob J Flex Syst Manag*, vol. 18, no. 4, pp. 275–289, Dec. 2017, doi: 10.1007/s40171-017-0162-8.

[125] J. M. Bass, S. Beecham, M. A. Razzak, C. N. Canna, and J. Noll, "An empirical study of the product owner role in scrum," in *Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings*, Gothenburg, Sweden, May 2018, pp. 123–124, doi: 10.1145/3183440.3195066.

[126] S. V. Spiegler, C. Heinecke, and S. Wagner, "Leadership Gap in Agile Teams: How Teams and Scrum Masters Mature," in *Agile Processes in Software Engineering and Extreme Programming*, Cham, 2019, pp. 37–52, doi: 10.1007/978-3-030-19034-7_3.

[127] R. J. Holden, A. Kulanthaivel, S. Purkayastha, K. M. Goggins, and S. Kripalani, "Know thy eHealth user: Development of biopsychosocial personas from a study of older adults with heart failure," *International Journal of Medical Informatics*, vol. 108, pp. 158–167, Dec. 2017, doi: 10.1016/j.ijmedinf.2017.10.006.

[128] M. Kuhrmann *et al.*, "Hybrid software and system development in practice: Waterfall, scrum, and beyond," presented at the ACM International Conference Proceeding Series, 2017, vol. Part F128767, pp. 30–39, doi: 10.1145/3084100.3084104.

[129] A. Azanha, A. R. T. T. Argoud, J. B. de Camargo Junior, and P. D. Antoniolli, "Agile project management with Scrum: A case study of a Brazilian pharmaceutical company IT project," *International Journal of Managing Projects in Business*, vol. 10, no. 1, pp. 121–142, Jan. 2017, doi: 10.1108/IJMPB-06-2016-0054.

[130] M. Elallaoui, K. Nafil, and R. Touahni, "Automatic Transformation of User Stories into UML Use Case Diagrams using NLP Techniques," presented at the Procedia Computer Science, 2018, vol. 130, pp. 42–49, doi: 10.1016/j.procs.2018.04.010.

[131] 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*, vol. 85, no. 6, pp. 1287–1299, 2012, doi: 10.1016/j.jss.2012.01.061.

[132] K. Bhavsar, S. Gopalan, and V. Shan, "Scrumbanfall: An Agile Integration of Scrum and Kanban with Waterfall in Software Engineering," *ResearchGate*, 2020. https://www.researchgate.net/publication/339042133_Scrumbanfall_An_Agile_Integration_of_Scrum _and_Kanban_with_Waterfall_in_Software_Engineering (accessed Apr. 20, 2020).

[133] A. Banijamali, R. Dawadi, M. O. Ahmad, J. Similä, M. Oivo, and K. Liukkunen, "Empirical Investigation of Scrumban in Global Software Development," in *Model-Driven Engineering and Software Development*, Cham, 2017, pp. 229–248, doi: 10.1007/978-3-319-66302-9_12.

[134] S. Rahim, A. E. Chowdhury, D. Nandi, and M. Rahman, "ScrumFall: A Hybrid Software Process Model," 2018, doi: 10.5815/ijitcs.2018.12.06.

[135] D. Taibi, V. Lenarduzzi, A. Janes, K. Liukkunen, and M. O. Ahmad, "Comparing Requirements Decomposition Within the Scrum, Scrum with Kanban, XP, and Banana Development Processes," in *Agile Processes in Software Engineering and Extreme Programming*, Cham, 2017, pp. 68–83, doi: 10.1007/978-3-319-57633-6_5.

[136] E. L.-C. Law and M. K. Lárusdóttir, "Whose Experience Do We Care About? Analysis of the Fitness of Scrum and Kanban to User Experience," *International Journal of Human–Computer Interaction*, vol. 31, no. 9, pp. 584–602, Sep. 2015, doi: 10.1080/10447318.2015.1065693.

[137] S. Tapp, R. Scott, R. Kemp, and S. Saunders, "Synthesizing Lean Software Development and Distributed Scrum," *1*, vol. 8, no. 1, Aug. 2019, Accessed: Apr. 20, 2020. [Online]. Available: https://secsjr.org/index.php/secs/article/view/232.

[138] B. T. Ramamoorthy and P. Mayilvahanan, "Comparative study on agile scrum over traditional waterfall lifecycle projects," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 11, no. 4, pp. 524–529, 2019.

[139] S. Al-Ratrout, "Impact of using agile methods in software engineering education: A case study," presented at the 2019 6th International Conference on Control, Decision and Information Technologies, CoDIT 2019, 2019, pp. 1986–1991, doi: 10.1109/CoDIT.2019.8820377.

[140] E. Lozada-Martinez, J. E. Naranjo, C. A. Garcia, D. M. Soria, O. R. Toscano, and M. V. Garcia, "SCRUM and Extreme Programming Agile Model Approach for Virtual Training Environment Design," presented at the 2019 IEEE 4th Ecuador Technical Chapters Meeting, ETCM 2019, 2019, doi: 10.1109/ETCM48019.2019.9014882.

[141] L. R. Begosso, L. H. B. Franco, D. S. Da Cunha, and L. C. Begosso, "The use of gamification to support the process of teaching scrum," presented at the Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE, 2019, p. 297, doi: 10.1145/3304221.3325583.

[142] U. Schafer, "Training scrum with gamification: Lessons learned after two teaching periods," presented at the IEEE Global Engineering Education Conference, EDUCON, 2017, pp. 754–761, doi: 10.1109/EDUCON.2017.7942932.

[143] S. Kurkovsky, "A simple game to introduce scrum concepts," presented at the Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE, 2020, p. 1321, doi: 10.1145/3328778.3372593.

[144] M. Paasivaara, V. Heikkilä, C. Lassenius, and T. Toivola, "Teaching students scrum using LEGO blocks," presented at the 36th International Conference on Software Engineering, ICSE Companion 2014 - Proceedings, 2014, pp. 382–391, doi: 10.1145/2591062.2591169.

[145] M. Villavicencio, E. Narvaez, E. Izquierdo, and J. Pincay, "Learning scrum by doing real-life projects," presented at the IEEE Global Engineering Education Conference, EDUCON, 2017, pp. 1450–1456, doi: 10.1109/EDUCON.2017.7943039.

[146] A. Przybyłek and M. Zakrzewski, "Adopting collaborative games into agile requirements engineering," presented at the ENASE 2018 - Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software Engineering, 2018, vol. 2018-March, pp. 54–64.

[147] Y. Y. Ng, J. Skrodzki, and M. Wawryk, "Playing the Sprint Retrospective: A Replication Study," *Lecture Notes in Business Information Processing*, vol. 376 LNBIP, pp. 133–141, 2020, doi: 10.1007/978-3-030-37534-8_7.

[148] S. Bick, K. Spohrer, R. Hoda, A. Scheerer, and A. Heinzl, "Coordination Challenges in Large-Scale Software Development: A Case Study of Planning Misalignment in Hybrid Settings," *IEEE Transactions on Software Engineering*, vol. 44, no. 10, pp. 932–950, 2018, doi: 10.1109/TSE.2017.2730870.

[149] T. Dingsøyr, N. B. Moe, T. E. Fægri, and E. A. Seim, "Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation," *Empirical Software Engineering*, vol. 23, no. 1, pp. 490–520, 2018, doi: 10.1007/s10664-017-9524-2.

[150] M. U. Al Hafidz and D. I. Sensuse, "The Effect of Knowledge Management System on Software Development Process with Scrum," in *2019 3rd International Conference on Informatics and Computational Sciences (ICICoS)*, Oct. 2019, pp. 1–6, doi: 10.1109/ICICoS48119.2019.8982506.

[151] A. Altaf, U. Fatima, W. H. Butt, M. W. Anwar, and M. Hamdani, "A systematic literature review on factors impacting agile adaptation in global software development," presented at the ACM International Conference Proceeding Series, 2019, pp. 158–163, doi: 10.1145/3348445.3348463.

[152] M. Shameem, C. Kumar, B. Chandra, and A. A. Khan, "Systematic review of success factors for scaling agile methods in global software development environment: A client-vendor perspective," presented at the Proceedings - 2017 24th Asia-Pacific Software Engineering Conference Workshops, APSECW 2017, 2018, vol. 2018-January, pp. 17–24, doi: 10.1109/APSECW.2017.22.

[153] P. Lous, P. Tell, C. B. Michelsen, Y. Dittrich, and A. Ebdrup, "From scrum to agile: A journey to tackle the challenges of distributed development in an agile team," presented at the ACM International Conference Proceeding Series, 2018, pp. 11–20, doi: 10.1145/3202710.3203149.

[154] J. B. Kroculick, "Architecting doctrinally-consistent C2 infrastructure services," presented at the Proceedings of SPIE - The International Society for Optical Engineering, 2019, vol. 11015, doi: 10.1117/12.2519262.

[155] A. F. Arbain, D. N. A. Jawawi, W. M. N. B. W. Kadir, and I. Ghani, "Case study on non-functional requirement change impact traceability for Agile software development," *International Journal on Advanced Science, Engineering and Information Technology*, no. 1, pp. 34–40, 2020, doi: 10.18517/ijaseit.10.1.10176.

[156] F. E. Castillo-Barrera, M. Amador-Garcia, H. G. Perez-Gonzalez, F. E. Martinez-Perez, and F. J. Torres-Reyes, "Adapting bloom's taxonomy for an agile classification of the complexity of the user stories in SCRUM," presented at the Proceedings - 2018 6th International Conference in Software Engineering Research and Innovation, CONISOFT 2018, 2019, pp. 139–145, doi: 10.1109/CONISOFT.2018.8645899.

[157] N. Santos, J. Pereira, F. Morais, J. Barros, N. Ferreira, and R. J. Machado, "Deriving user stories for distributed Scrum teams from iterative refinement of architectural models," presented at the ACM International Conference Proceeding Series, 2018, vol. Part F147763, doi: 10.1145/3234152.3234165.

[158] S. Hariharan, A. Rengarajan, and R. Prem Kumar, "Scrum based scaling using agile method to test software projects and its future solutions using in artificial neural networks," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 9, pp. 223–230, 2019.

[159] L. Butgereit, "Using machine learning to prioritize automated testing in an agile environment," presented at the 2019 Conference on Information Communications Technology and Society, ICTAS 2019, 2019, doi: 10.1109/ICTAS.2019.8703639.

[160] O. Malgonde and K. Chari, "An ensemble-based model for predicting agile software development effort," *Empirical Software Engineering*, vol. 24, no. 2, pp. 1017–1055, 2019, doi: 10.1007/s10664-018-9647-0.

[161] "About | Elsevier Scopus Blog." https://blog.scopus.com/about (accessed Apr. 19, 2020).