Global software development agile planning model: challenges and current trends

Hajar Lamsellak, Mohammed Ghaouth Belkasmi

Smart Information, Communication and Technologies Laboratory (SmartICT Lab), National School of Applied Sciences of Oujda, Mohammed First University, Oujda, Morocco

Article Info

ABSTRACT

Agile planning offers a number of benefits that make the customers active members of the team throughout the project. In global software development (GSD), geographic separations demand special attention to harness these benefits. Our paper conducted a systematic mapping study (SMS) to analyze GSD-specific agile planning challenges followed by a systematic literature review (SLR) for efficient solutions. These studies led to a model for agile planning in global software development supporting GSD practitioners during this process.

Corresponding Author:

Hajar Lamsellak
Smart Information, Communication and Technologies Laboratory (SmartICT Lab)
National School of Applied Sciences of Oujda, Mohammed First University
Oujda, Morocco
Email: lamsellak.hajar@ump.ac.ma

1. INTRODUCTION

In the past decade, Agility and Global Software Development (GSD) emerged as distinct approaches, garnering attention despite fundamental differences [1]–[3]. GSD, involving participants from diverse regions [4]–[6], is adopted by companies for purposes including market expansion, diverse skills, and cost savings [7], [8]. However, GSD practitioners face challenges across the development lifecycle [9]–[86], leading to coordination, collaboration, and communication issues [29], [30] negatively affecting project planning.

Traditional project planning, with its defined goals, scopes, and actions [31], often leads to budget overruns and project failures [32]. In contrast, agile planning is simple, its clear roles and short iterations enhance goal refinement and efficiency [33]. Agile methodologies boost team productivity, product delivery, and customer satisfaction [34]–[39]. They have been effective in addressing Global Software Development (GSD) challenges [40]–[52]. However, GSD introduces novel issues, widespread agile adoption [53], [54].

Existing studies focuses on agile in remote software development contexts [55]–[76]: Scrum [77]–[80], Kanban [81]–[84], extreme programming [85]–[88] and related issues [89]–[94]. Our earlier contributions to the pre-planning [95], release planning [96], and iteration planning [97]. This paper introduces the agile GSD planning model. We first identified challenges encountered in agile GSD through a systematic mapping study by reviewing 22 papers from 2012 to 2022. Second, we select proposed solutions through a systematic literature review by reviewing 34 selected studies from 2018 to 2022. Sections 2-4 outline the research method, findings, and future directions, respectively.

2. METHOD

2.1. Systematic mapping study

We used a systematic mapping study following the methods of [98]. The mapping process encompasses five key steps: i) defining mapping questions, ii) search approach, iii) selecting relevant studies, iv) extracting data, and iv) synthesizing data. Subsequent sections detail these steps in the process.

2.1.1. Defining mapping questions

The objective of the conducted systematic mapping study is to categorize and examine challenges encountered when using agile planning in distributed software development. Aligned with this, Table 1 presents three mapping questions (MQs). Each mapping question is associated with the corresponding motivation.

Table 1. Mapping questions (MQs)

ID	Mapping question (MQ)	Primary motivation
MQ1	What are the issues faced during pre-planning?	Explore studied issues in global software development pre-planning.
MQ2	What are the issues faced during release planning?	Uncover key global software development release planning issues.
MQ3	What are the issues faced during iteration planning?	Choose key global software development iteration planning issues.

2.1.2. Search approach

The search approach, targeting Table 1 mapping questions, involved five online databases: Google Scholar, ScienceDirect, Scopus, SpringerLink, and Web of Science. Table 2 illustrates the search query employed in this study. Which was formulated by identifying synonyms and alternative spellings for the primary terms outlined in Table 1.

Table 2. Systematic mapping study: search query

Search query used in the systematic mapping study
"Barriers" OR "Problems"OR "Challenges" OR "Issues" AND "agile global software development" OR "agile global software engineering" OR "agile distributed software development" OR "agile distributed virtual Team" OR "agile alobal teams".

2.1.3. Selecting relevant studies

The selection aimed to identify relevant papers for the systematic mapping study goal. We evaluated each paper using the inclusion and exclusion criteria from Table 3. Resulting in either inclusion or exclusion. Figure 1 illustrates the paper selection process: 806 potential papers, 5 added through author knowledge, 20 remaining after deduplication, and 2 from reference studies, totaling 22 selected.

Table 3. Systematic mapping study: inclusion and exclusion criterias

| Inclusion criteria	Exclusion criteria
(1) Studies addressing issues in global software project pre-planning.	(1) Studies issues in non-agile global software development.
(2) Researches on challenges in global software release planning.	(2) Studies addressing issues unrelated to agile planning activities.
(3) Studies challenges in global software project iteration planning.	(3) Studies on issues in traditional software development planning.
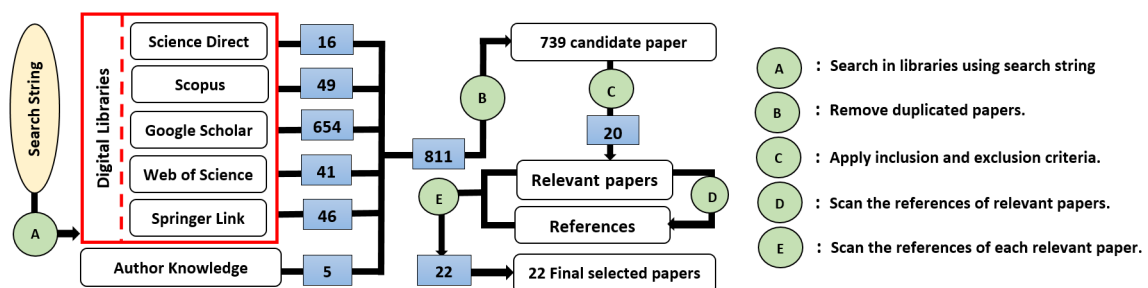
Figure 1. Effects of selecting different switching under dynamic condition

2.1.4. Extracting data strategy and synthesis method

The extracting data process collected information related to the MQs outlined in the study. Table 4 shows the data extraction process employed to gather information from the chosen studies. The narrative synthesis was used to synthesize and summarize the data related to MQs.

Table 4. Systematic mapping study: extracting data form

Extracting data form of the systematic mapping study results
Study informations : study identifier, year of publication, author(s) and paper title
Mapping question 1 : Encountered challenges in pre-planning of global software development projects.
Mapping question 2 : Encountered challenges in release planning of global software development projects.
Mapping question 3 : Encountered challenges in iteration planning of global software development projects.

2.2. Systematic literature review

The second study was a systematic literature review following [99]. The review comprises five steps: i) defining review questions, ii) research approach, iii) selecting relevant studies, iv) extracting data, and v) synthesizing data. All these steps will be explained in the subsequent sections.

2.2.1. Defining review questions

The systematic literature review aims to synthesize relevant studies on the definition and implementation of agile planning in global software development. Table 5 provides a list of the three reviews questions we identified. Each review question is associated with their main motivation in order to acheive this goal.

Table 5. Review questions (RQs)

ID	Review question (RQ)	Primary motivation
RQ1	How organizations pre-plan global software development?	To identify and describe GSD pre-planning activities.
RQ2	How is release planning conducted in global software development?	To highlight established practices for GSD release planning.
RQ3	How is iteration planning held in global software development?	To select practiced activities for GSD iteration planning.

2.2.2. Search approach

The search approach aimed to find studies addressing the RQs in Table 5. First, search terms were derived from the RQs and checked for synonyms to establish the search string for the systematic literature review presented in Table 6. These terms were used to search in the same five digital libraries.

Table 6. Systematic literature review: search query

Search query used in the systematic literature review
"Agile planning" OR "agile software development planning" OR "agile planning phases" OR ("pre-planning" AND " Release planning" AND "iteration planning") AND "global software development" OR "distributed software development".

2.2.3. Selecting relevant studies

We aimed to find relevant studies addressing the review questions. Each paper was assessed using the inclusion and exclusion standards in Table 7. The decision based on title, abstract, keywords, or full content. Figure 2 shows the paper selection process: 749 candidate papers. The 5 added through authors knowledge, 32 remaining after deduplication, and 2 from references studies scan. Finally, 34 papers were selected.

Table 7. Systematic literature review: inclusion and exclusion criterias

| Inclusion criteria	Exclusion criteria
(1) Studies describing software development agile planning levels.	(1) Studies addressing GSD traditionnel planning.
(2) Studies addressing the agile global software development term.	(2) Studies presenting reviews of global software development.
(3) Agile global software development agile planning studies.	(3) Studies focusing on non agile planning practices.
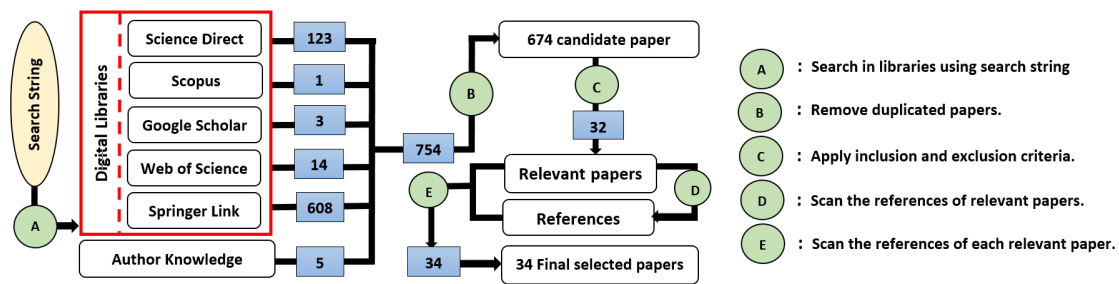
Figure 2. Systematic literature review: search, selection and quality assessment process

2.2.4. Extracting and synthesizing data

Table 8 outlines the data extraction process employed in this study. We involved two primary steps: reading selected papers and collecting information relevant to the research questions. Synthesizing data categorized evidence for RQs and identified agile practices in global software development teams.

Table 8. Systematic literature review: extracting data form

Extracting data form of the systematic literature review results
Study informations : Study identifier, year of publication, author(s) and paper title.
Review question 1 : How the pre-planning process is carried out in the global software Development ?
Review question 2 : How is the release planning process conducted in Distributed Software Development ?
Review question 3 : How is the execution of iteration planning of global software development (GSD) ?

3. RESULTS AND DISCUSSION
3.1. Systematic mapping study

This section reveals the outcomes of our systematic mapping study, responding to the three mapping questions in Table 1. Table 9 in *appendix* outlines encountered issues in the agile planning process, including pre-planning, release planning, and iteration planning stages. A total of 33 issues were identified, with 10 occurring during the pre-planning stage (30%), 12 during release planning (37%), and 11 affecting iteration planning (33%).

3.2. Systematic literature study

Table 10 in *appendix* displays the systematic literature review findings on agile planning levels. The agile best practices that were deemed appropriate for application during each level were listed. Furthermore, there is a summary of the key solutions that each agile approach has contributed to.

3.3. Global software development agile planning model

The results of the conducted systematic sapping study and the systematic siterature seview were used to implement the agile planning model presented in Figure 3. The model is composed by the three main phases of agile planning process. A group of effective agile practices are linked to each phase.
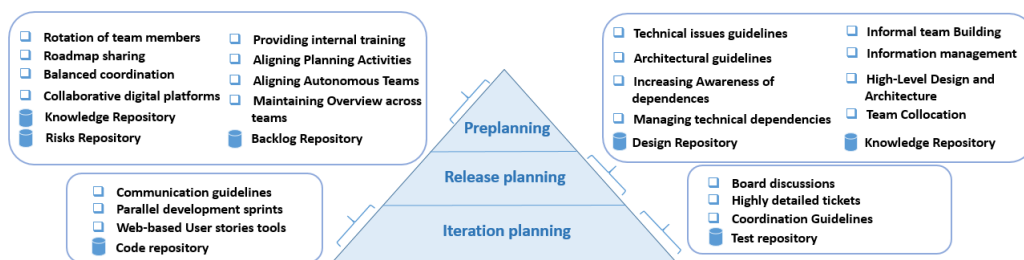

Figure 3. Agile global software development planning model

4. CONCLUSION

In this paper, a systematic mapping study (SMS) and a systematic literature review (SLR) were conducted to analyze challenges in agile planning in GSD. These findings were used to develop an agile planning model for GSD. Future work includes creating a detailed knowledge database for implementing agile planning phases effectively.

APPENDIX

Table 9. Agile planning challenges in global software development projects

Source	Challenge description	Agile planning level
[100]		
	• Budgeting: Agile methods have limitations such as lack of up front planning, budget constraints, lack of documents, predictability, meetings, compliance, and training.	Pre-planning
	• Requirement prioritization: Is difficult due to multiple stakeholders and customers involved, making it difficult to decide which features to implement and deliver first.	Release planning
	• The team management: It is a major difficulty encountered that includes internal conflicts, difficulty integrating new members, offshoring, remplacement, and resistance to change.	Release planning
	• Organization: Agile Global Software Development practitioners face organizational issues such as commitment, fear of new approaches, learning, senior management support, and control.	Release planning
	• Poor technical debt management: It can be overcome by placing technical debt repayment at the top of the product backlog; utilizing impartial testing to pay off technical debts.	Release planning
	• Process: Agile Global Software Development faces process-related challenges such as lack of best practices, direction, mapping, human factors, resistance to change, and lack of support.	Release planning
	• Customer: Communication with consumers can lead to irrational and unplanned change requests, making it difficult to follow Agile Software Development principle.	Release planning
	• Planning: Lack of understanding of complexity, lack of focus on objectives, and insufficient development involvement lead to over-scoping requirements.	Release planning
	• Absence of unified "Definition of Done": Creates challenges in distributed agile teams as they lack a shared criteria for defining the completion of stories, iterations, releases, and projects.	Release planning
	• Task dependency in Agile Global Software Development: where work relies on prior tasks, can lead to canceled sprints, project delays, and reduced final product delivery efficiency.	Iteration planning
[101]		
	• Fragmented documentation: Documentation in agile projects is fragmented, making it difficult to access and decentralized and Artifacts of documentation are further fragmented by distribution.	Pre-planning
	• Unclear outcome of combined agile method usage: Uncoordinated usage of hybrid agile techniques can undermine project success and discourage future adoption.	Iteration planning
	• Agile Global Software Development lacks clear guidelines: Artifacts like burndown charts and product backlogs are employed to address distribution issues.	Iteration planning
	• Lack of acceptance of agile principles: Resisting Agile principles despite understanding; diverse stakeholders often do not support common Agile values.	Iteration planning
[102]		
	• Teams lack dependency awareness: Dependency understanding is vital; Scrum-based intra-team coordination effectively handles daily issues and dependencies, highlighting their significance.	Release planning
	• Misalignment of planning activities: Iterative planning between teams aids in resolving conflicts and enables task definition and prioritization at various levels.	Release planning
	• Lack of evaluation process: to ensure they enhance dependency awareness and coordinate project aspects effectively (specification, prioritization, estimation, and allocation).	Iteration planning
[103], [104]	• Unprepared communication tools: Communication tools are essential for effective communication between construction teams and clients, such as web-based or video conferencing facilities.	Pre-planning Pre-planning
	• Costly initial Adoption: Including operating system updates, guidance, and training expenses.	
	• Lack of Management commitments: Management commitment is essential for successful application of agile methodologies in distributed software development contexts.	Pre-planning
	• Lack of knowledge sharing: It hinders Agile adoption in distributed software development.	Iteration planning
	• Skepticism of Agile software development: Due to its contrast with waterfall methodologies, making adaptation challenging. Managers encounter additional hurdles with new technology.	Iteration planning

Table 9. Agile planning challenges in global software development projects (*continue*)

Source	Challenge description	Agile planning level
[105]		
	• Requirements Understanding: Diverse stakeholder communities pose challenges in interpreting and integrating information. Varied language formality require community-specific structuring.	Release planning
	• Balanced Coordination: Is crucial, with project leads harmonizing tasks for success.	Pre-planning
	• Roadmap sharing: The Scrum Master and Business Analyst should communicate the company's objectives and product direction to the entire team to ensure developers understand them.	Pre-planning
	• User stories communication: Companies can enhance collaboration, reduce communication issues, and improve understanding of user stories and product visions using web-based tools.	Iteration planning
[106]		
	• Miscommunication of requirements: Developers may suffer from a lack of specific requirements information which can lead them to create their own requirements to meet client needs.	Pre-planning
	• Lack of trust: Obstructs distributed teams, and hinders teamwork and community-building.	Iteration planning
	• Technical issues: Poor internet access and infrastructure limit distributed communication.	Release planning
	• Informal and frequent communication among distant coworkers: Skype and Slack are popular for proximity, instant responses, and informal, continuous dialogue in addition to work.	Iteration planning
[107]		
	• Complex technical dependencies: Lead to bottlenecks, code bases, errors, and repeating patterns.	Iteration planning
	• Aligning Autonomous Teams: Challenging to align autonomous teams in a large-scale program due to process dependencies, limited freedom, and technical inconsistency.	Pre-planning
	• Cross-team Overview: Weekly program demo and Confluence documents used to tackle challenges in maintaining cross-Team visibility and aiding identification.	Pre-planning

Table 10. Agile practices recommended for adoption during the agile planning phase

References	Best agile practice and apported solutions
[108]-[112]	
	• Knowledge Repository: This holds project vision, policies, and terminology for team-wide comprehension.
	• Vision and Scope: Project planning should encompass creating a well-defined and widely understood vision, team collocation for process, tool, and technique briefings.
	• Training: Effective training is crucial in transitioning to Agile; inadequate training may cause problems.
	• Completeness of Agile Scope: Scope definition is significant in diverse company settings. The key components of scope definition were defined and quantified using a scope development tool.
	• High-Level Design and Architecture: Creating a carefully planned workload distribution, collocation, and architecture rules is essential for reducing Global Software Development challenges.
[113], [114]	
	• Task Allocation: Distributed tasks require consideration of dependencies, skills, geography, costs, and communication.
	• Risks Repository: Identifies and mitigates anticipated risks in Global Software Development through risk assessment.
	• Code repository: The code repository stores the code for finished user stories.
	• Test Repository: Holds test scenarios connected to user stories.
	• Release Repository: Stores fully tested user stories with documentation ready for deployment.
	• Design Repository: This repository can store the product's structure, enabling the architecture team to store succinct and prioritized design choices along with their rationales.
[115]	
	• Time required for knowledge management practice implementation: Is divided into two components: time each team member should reserve and time needed for management initiatives to have an impact.
	• Knowledge management sharing: Knowledge exchange and management strategies should be common to all teams.
	• Team Rotation: The team members should rotate through all of the roles that make up a Scrum team.
	• Team routines: Scrum team defines weekly weekly tasks to be carried out on a regular basis.
	• Collaborative platforms: Distance meetings and real-time monitoring enable effective communication, knowledge sharing, reduced time spent, decreased costs, and increased satisfaction and teamwork among employees.
	• Internal training: Product-level training programs provide company-wide training resources for team members.
	• Book studies: Promoting organizational learning through community-based book discussions among team members.
	• Architectural guidelines: Ensure consistency and foster agile culture across corporate levels.
	• Board discussions serve to clarify ticket implementation processes by addressing any doubts or questions that may arise.

Table 10. Agile practices recommended for adoption during the agile planning phase (*continue*)

References	Best agile practice and apported solutions
[115]	- Retrospectives: Monthly local retrospectives, involving Scrum Masters, address issues in large agile teams. - Direct contact in an open workspace: Informal exchanges between individuals, regardless of their Scrum team affiliation. - Collaborative digital platforms: Real-time monitoring and remote meetings are essential for effective communication, knowledge sharing, reduced costs, and increased employee satisfaction. - Online tools for content collaboration: Jira and Confluence centralize task management, dependencies, workflow verification, and performance reporting for dynamic, transparent, and effective development. - Tickets with a lot of info: The presence of larger and more complete tickets helped the developed software's assurance of quality and awareness of client needs.
[116]–[123]	- Initial Program Increment Planning: Knowledge gaps, chaotic event, scheduling conflicts, surprises, limited technical understanding, logistic challenges, technical interdependencies, and management reviews. - Agile Release Team: Manages dependencies and train adjustments efficiently.

REFERENCES

[1] M. Afshari and T. J. Gandomani, "Quality of agile adoption in global software development: an assessment model," *The Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 21, no. 1, pp. 367, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp367-376.

[2] B. G. Tavares, M. Keil, C. E. S. Da-Silva, and A. D. De-Souza, "A risk management tool for agile software development," *Journal of Computer Information Systems*, vol. 61, no. 6, pp. 561–570, Nov. 2021, doi: 10.1080/08874417.2020.1839813.

[3] H. C. Estler, M. Nordio, C. A. Furia, B. Meyer, and J. Schneider, "Agile vs. structured distributed software development: a case study," *Empirical Software Engineering*, vol. 19, no. 5, pp. 1197–1224, 2014, doi: 10.1007/s10664-013-9271-y.

[4] S. Y. Chadli, A. Idri, J. L. Fernandez-Aleman, J. N. Ros, and A. Toval, "Identifying risks of software project management in global software development: an integrative framework," in *2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA)*, 2016, pp. 1–7. doi: 10.1109/AICCSA.2016.7945664.

[5] S. V. Shrivastava and U. Rathod, "Categorization of risk factors for distributed agile projects," *Information and Software Technology*, vol. 58, pp. 373–387, 2015, doi: 10.1016/j.infsof.2014.07.007.

[6] R. Vallon, S. Strobl, M. Bernhart, R. Prikladnicki, and T. Grechenig, "ADAPT: a framework for agile distributed software development," *IEEE Software*, vol. 33, no. 6, pp. 106–111, 2016, doi: 10.1109/MS.2016.130.

[7] S. Beecham, T. Clear, R. Lal, and J. Noll, "Do scaling agile frameworks address global software development risks? An empirical study," *Journal of Systems and Software*, vol. 171, pp. 110-823, 2021, doi: 10.1016/j.jss.2020.110823.

[8] J. D. Herbsleb and A. Mockus, "An empirical study of speed and communication in globally distributed software development," *IEEE Transactions on software engineering*, vol. 29, no. 6, pp. 481–494, 2003, doi: 10.1109/TSE.2003.1205177.

[9] R. Jain and U. Suman, "Effectiveness of agile practices in global software development," *International Journal of Grid and Distributed Computing (IJGDC)*, vol. 9, no. 10, pp. 231–248, Oct. 2016, doi: 10.14257/ijgdc.2016.9.10.21.

[10] S. Jalali and C. Wohlin, "Global software engineering and agile practices: a systematic review," *Journal of Software: Evolution and Process*, vol. 24, no. 6, pp. 643–659, 2012, doi: 10.1002/smr.561.

[11] T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: a systematic review," *Information and Software Technology*, vol. 50, no. 9–10, pp. 833–859, 2008, doi: 10.1016/j.infsof.2008.01.006.

[12] D. Šmite, C. Wohlin, T. Gorschek, and R. Feldt, "Empirical evidence in global software engineering: a systematic review," *Empirical Software Engineering*, vol. 15, no. 1, pp. 91–118, 2010, doi: 10.1007/s10664-009-9123-y.

[13] S. Tenhunen, T. Männistö, M. Luukkainen, and P. Ihantola, "A systematic literature review of capstone courses in software engineering," *Information and Software Technology*, vol. 159, p. 107191, 2023, doi: 10.1016/j.infsof.2023.107191.

[14] M. Satratzemi, X. Stelios, and D. Tsompanoudi, "Distributed pair programming in higher education: a systematic literature review," *Journal of Educational Computing Research*, vol. 61, no. 3, pp. 546–577, 2023, doi: 10.1177/07356331221122884.

[15] D. Itzik and G. Roy, "Does agile methodology fit all characteristics of software projects? Review and analysis," *Empirical Software Engineering*, vol. 28, no. 4, p. 105, 2023, doi: 10.1007/s10664-023-10334-7.

[16] Z. Hoy and M. Xu, "Agile software requirements engineering challenges-solutions-a conceptual framework from systematic literature review," *Information*, vol. 14, no. 6, p. 322, Jun. 2023, doi: 10.3390/info14060322.

[17] A. S. Campanelli and F. S. Parreiras, "Agile methods tailoring-a systematic literature review," *Journal of Systems and Software*, vol. 110, pp. 85–100, 2015, doi: 10.1016/j.jss.2015.08.035.

[18] Y. I. Alzoubi, A. Q. Gill, and A. Al-Ani, "Empirical studies of geographically distributed agile development communication challenges: a systematic review," *Information and Management*, vol. 53, no. 1, pp. 22–37, 2016, doi: 10.1016/j.im.2015.08.003.

[19] M. Zahedi et al., "A systematic review of knowledge sharing challenges and practices in global software development," *International Journal of Information Management*, vol. 36, no. 6, pp. 995–1019, 2016, doi: 10.1016/j.ijinfomgt.2016.06.007.

[20] F. O. Bjørnson et al., "Knowledge management in software engineering: a systematic review of studied concepts, findings and research methods used," *Information and Software Technology*, vol. 50, no. 11, pp. 1055–1068, 2008, doi: 10.1016/j.infsof.2008.03.006.

[21] T. Tahir, G. Rasool, and C. Gencel, "A systematic literature review on software measurement programs," *Information and Software Technology*, vol. 73, pp. 101–121, 2016, doi: 10.1016/j.infsof.2016.01.014.

[22] R. Hoda, N. Salleh, J. Grundy, and H. M. Tee, "Systematic literature reviews in agile software development: a tertiary study," *Information and Software Technology*, vol. 85, pp. 60–70, 2017, doi: 10.1016/j.infsof.2017.01.007.

[23] T. Besker, A. Martini, and J. Bosch, "Managing architectural technical debt: a unified model and systematic literature review," *Journal of Systems and Software*, vol. 135, pp. 1–16, 2018, doi: 10.1016/j.jss.2017.09.025.

[24] L. Chen and M. A. Babar, "A systematic review of evaluation of variability management approaches in software product lines," *Information and Software Technology*, vol. 53, no. 4, pp. 344–362, 2011, doi: 10.1016/j.infsof.2010.12.006.

[25] S. Nidhra *et al.*, "Knowledge transfer challenges and mitigation strategies in global software development-a systematic literature review and industrial validation," *International Journal of Information Management (IJIM)*, vol. 33, no. 2, pp. 333–355, 2013, doi: 10.1016/j.ijinfomgt.2012.11.004.

[26] W. Afzal, R. Torkar, and R. Feldt, "A systematic review of search-based testing for non-functional system properties," *Information and Software Technology*, vol. 51, no. 6, pp. 957–976, 2009, doi: 10.1016/j.infsof.2008.12.005.

[27] A. V. Pizzoleto, F. C. Ferrari, J. Offutt, L. Fernandes, and M. Ribeiro, "A systematic literature review of techniques and metrics to reduce the cost of mutation testing," *Journal of Systems and Software*, vol. 157, p. 110388, 2019, doi: 10.1016/j.jss.2019.07.100.

[28] P. Bjørn, A. M. Søderberg, and S. Krishna, "Translocality in global software development: the dark side of global agile," *Human-Computer Interaction*, vol. 34, no. 2, pp. 174–203, Mar. 2019, doi: 10.1080/07370024.2017.1398092.

[29] G. Ammad *et al.*, "An empirical study to investigate the impact of communication Issues in GSD in Pakistan's IT Industry," *IEEE Access*, vol. 7, pp. 171648–171672, 2019, doi: 10.1109/ACCESS.2019.2953008.

[30] Y. I. Alzoubi, A. Q. Gill, and A. Al-Ani, "Distributed agile development communication: an agile architecture driven framework," *Journal of Software (JSW)*, vol. 10, no. 6, pp. 681–694, 2015, doi: 10.17706/jsw.10.6.681-694.

[31] M. G. Belkasmi *et al.*, "Global IT project management: an agile planning assistance," in *Advances in Smart Technologies Applications and Case Studies*, 2020, pp. 575–582. doi:10.1007/978-3-030-53187-4_63.

[32] S. Zhong, C. Liping, and C. Tian-en, "Agile planning and development methods," *In 2011 3rd International Conference on Computer Research and Development*, 2011, pp. 488-491, doi: 10.1109/ICCRD.2011.5764064.

[33] H. F. Cervone, "Improving strategic planning by adapting agile methods to the planning process," *Journal of Library Administration*, vol. 54, no. 2, pp. 155–168, Feb. 2014, doi: 10.1080/01930826.2014.903371.

[34] T. M. Gahroee, T. J. Gandomani, and M. S. Aghaei, "The main pillars of agile consolidation in newly agile teams in agile software development," *The Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 26, no. 2, pp. 1216, May 2022, doi: 10.11591/ijeecs.v26.i2.pp1216-1226.

[35] S. Alsaqqa, S. Sawalha, and H. Abdel-Nabi, "Agile software development: methodologies and trends," *International Journal of Interactive Mobile Technologies*, vol. 14, no. 11, p. 246, Jul. 2020, doi: 10.3991/ijim.v14i11.13269.

[36] G. S. Matharu, A. Mishra, H. Singh, and P. Upadhyay, "Empirical study of agile software development methodologies: a comparative analysis," *ACM SIGSOFT Software Engineering Notes*, vol. 40, no. 1, pp. 1–6, Feb. 2015, doi: 10.1145/2693208.2693233.

[37] P. Rodríguez, M. Mäntylä, M. Oivo, L. E. Lwakatare, P. Seppänen, and P. Kuvaja, "Advances in using agile and lean processes for software development," in *Advances in Computers*, Elsevier, 2019, pp. 135–224. doi: 10.1016/bs.adcom.2018.03.014.

[38] S. Gupta and D. Gouttam, "Towards changing the paradigm of software development in software industries: an emergence of agile software development," *International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM)*, Chennai, India: IEEE, 2017, pp. 18–21. doi: 10.1109/ICSTM.2017.8089120.

[39] Gandomani, "Obstacles in moving to agile software development methods; at a glance," *Journal of Computer Science*, vol. 9, no. 5, pp. 620–625, May 2013, doi: 10.3844/jcssp.2013.620.625.

[40] M. Paasivaara, S. Durasiewicz, and C. Lassenius, "Using scrum in distributed agile development: a multiple case study," *In 2009 Fourth IEEE International Conference on Global Software Engineering*, 2009, pp. 195-204, doi: 10.1109/ICGSE.2009.27.

[41] E. Hossain, M. A. Babar, and H.-Y. Paik, "Using scrum in global software development: a systematic literature review," *In 2009 Fourth IEEE International Conference on Global Software Engineering*, 2009, pp. 175-184, doi: 10.1109/ICGSE.2009.25.

[42] K. Dikert, M. Paasivaara, and C. Lassenius, "Challenges and success factors for large-scale agile transformations: a systematic literature review," *Journal of Systems and Software*, vol. 119, pp. 87–108, 2016, doi: 10.1016/j.jss.2016.06.013.

[43] M. Kaisti *et al.*, "Agile methods for embedded systems development-a literature review and a mapping study," *Journal on Embedded Systems*, vol. 2013, no. 1, p. 15, 2013, doi: 10.1186/1687-3963-2013-15.

[44] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, "Agile software development methods: review and analysis," arXiv, 2017, doi: 10.48550/ARXIV.1709.08439.

[45] L. Lagerberg *et al.*, "The impact of agile principles and practices on large-scale software development projects: a multiple-case study of two projects at ericsson," *Empirical Software Engineering and Measurement (ESEM)*, 2013, pp. 348-356, doi: 10.1109/ESEM.2013.53.

[46] O. Salo and P. Abrahamsson, "Agile methods in European embedded software development organisations: a survey on the actual use and usefulness of extreme programming and scrum," *IET Software*, vol. 2, no. 1, p. 58, 2008, doi: 10.1049/iet-sen:20070038.

[47] P. D. O. Santos and M. M. De Carvalho, "Exploring the challenges and benefits for scaling agile project management to large projects: a review," *Requirements Engineering*, vol. 27, no. 1, pp. 117–134, 2022, doi: 10.1007/s00766-021-00363-3.

[48] M. Alqudah and R. Razali, "A review of scaling agile methods in large software development," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 6, no. 6, p. 828, Dec. 2016, doi: 10.18517/ijaseit.6.6.1374.

[49] M. A. Blooshi, S. Jafer, and K. Patel, "Review of formal agile methods as cost-effective airworthiness certification processes," *Journal of Aerospace Information Systems*, vol. 15, no. 8, pp. 471–484, 2018, doi: 10.2514/1.I010601.

[50] J. Díaz, J. Pérez, P. P. Alarcón, and J. Garbajosa, "Agile product line engineering-a systematic literature review," *Software: Practice and Experience*, vol. 41, no. 8, pp. 921–941, 2011, doi: 10.1002/spe.1087.

[51] H. Saeeda, H. Khalid, M. Ahmed, A. Sameer, and F. Arif, "Systematic literature review of agile scalability for large scale projects," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 6, no. 9, 2015, doi: 10.14569/IJACSA.2015.060908.

[52] T. Dingsøyr, T. E. Fægri, and J. Itkonen, "What is large in large-scale? A taxonomy of scale for agile software development," in *Product-Focused Software Process Improvement*, 2014, pp. 273–276. doi: 10.1007/978-3-319-13835-0_20.

[53] U. Eklund *et al.*, "Industrial challenges of scaling agile in mass-produced embedded systems," in *Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation*, 2014, pp. 30–42. doi: 10.1007/978-3-319-14358-3_4.

[54] C. Berger *et al.*, "Expectations and challenges from scaling agile in mechatronics-driven companies-a comparative case study," *in Agile Processes in Software Engineering and Extreme Programming*, 2015, pp. 15–26. doi: 10.1007/978-3-319-18612-2_2.

[55] N. Rashid and S. U. Khan, "Agile practices for global software development vendors in the development of green and sustainable software," *Journal of Software: Evolution and Process*, vol. 30, no. 10, p. e1964, 2018, doi: 10.1002/smr.1964.

[56] A. A. Khan, M. Shameem, M. Nadeem, and M. A. Akbar, "Agile trends in Chinese global software development industry: fuzzy AHP based conceptual mapping," *Applied Soft Computing*, vol. 102, p. 107090, 2021, doi: 10.1016/j.asoc.2021.107090.

[57] C. J. P. Calvache *et al.*, "Scrum+: a scaled Scrum for the agile global software development project management with multiple models," *Revista Facultad de Ingeniería*, no. 93, pp. 105–116, Aug. 2019, doi: 10.17533//udea.redin.20190519.

[58] T. Dingsøyr *et al.*, "Towards principles of large-scale agile development: a summary of the workshop at XP2014 and a revised research agenda," *in Agile Methods. Large-Scale Development, RefTest*, 2014, pp. 1–8. doi: 10.1007/978-3-319-14358-3_1.

[59] M. Paasivaara *et al.*, "Towards rapid releases in large-scale XaaS development at ericsson: a case study," *in 2014 IEEE 9th International Conference on Global Software Engineering*, 2014, pp. 16–25. doi: 10.1109/ICGSE.2014.22.

[60] A. A. Khan *et al.*, "Systematic literature review and empirical investigation of barriers to process improvement in global software development: Client–vendor perspective," *Information and Software Technology*, vol. 87, pp. 180–205, 2017, doi: 10.1016/j.infsof.2017.03.006.

[61] S. El-Sharkawy *et al.*, "Metrics for analyzing variability and its implementation in software product lines: a systematic literature review," *Information and Software Technology*, vol. 106, pp. 1–30, 2019, doi: 10.1016/j.infsof.2018.08.015.

[62] M. Marques, J. Simmonds, P. O. Rossel, and M. C. Bastarrica, "Software product line evolution: a systematic literature review," *Information and Software Technology*, vol. 105, pp. 190–208, 2019, doi: 10.1016/j.infsof.2018.08.014.

[63] R. S. Machado, R. B. Almeida, A. M. Pernas, and A. C. Yamin, "State of the art in hybrid strategies for context reasoning: a systematic literature review," *Information and Software Technology*, vol. 111, pp. 122–130, 2019, doi: 10.1016/j.infsof.2019.01.010.

[64] A. S. Guinea, G. Nain, and Y. Le Traon, "A systematic review on the engineering of software for ubiquitous systems," *Journal of Systems and Software*, vol. 118, pp. 251–276, 2016, doi: 10.1016/j.jss.2016.05.024.

[65] R. T. Geraldi, S. Reinehr, and A. Malucelli, "Software product line applied to the internet of things: a systematic literature review," *Information and Software Technology*, vol. 124, p. 106293, 2020, doi: 10.1016/j.infsof.2020.106293.

[66] A. Dikici, O. Turetken, and O. Demirors, "Factors influencing the understandability of process models: a systematic literature review," *Information and Software Technology*, vol. 93, pp. 112–129, 2018, doi: 10.1016/j.infsof.2017.09.001.

[67] L. T. Heeager and P. A. Nielsen, "A conceptual model of agile software development in a safety-critical context: a systematic literature review," *Information and Software Technology*, vol. 103, pp. 22–39, 2018, doi: 10.1016/j.infsof.2018.06.004.

[68] S. Hosseinzadeh *et al.*, "Diversification and obfuscation techniques for software security: a systematic literature review," *Information and Software Technology*, vol. 104, pp. 72–93, 2018, doi: 10.1016/j.infsof.2018.07.007.

[69] B. Kitchenham and P. Brereton, "A systematic review of systematic review process research in software engineering," *Information and Software Technology*, vol. 55, no. 12, pp. 2049–2075, 2013, doi: 10.1016/j.infsof.2013.07.010.

[70] J. A. Pereira, M. Acher, H. Martin, J. M. Jézéquel, G. Botterweck, and A. Ventresque, "Learning software configuration spaces: a systematic literature review," *Journal of Systems and Software*, vol. 182, p. 111044, 2021, doi: 10.1016/j.jss.2021.111044.

[71] R. Malhotra, "A systematic review of machine learning techniques for software fault prediction," *Applied Soft Computing*, vol. 27, pp. 504–518, 2015, doi: 10.1016/j.asoc.2014.11.023.

[72] D. Radjenović, M. Heričko, R. Torkar, and A. Živković, "Software fault prediction metrics: a systematic literature review," *Information and Software Technology*, vol. 55, no. 8, pp. 1397–1418, 2013, doi: 10.1016/j.infsof.2013.02.009.

[73] L. Ochoa, O. González-Rojas, A. P. Juliana, H. Castro, and G. Saake, "A systematic literature review on the semi-automatic configuration of extended product lines," *Journal of Systems and Software*, vol. 144, pp. 511–532, 2018, doi: 10.1016/j.jss.2018.07.054.

[74] V. Garousi, A. Coşkunçay, A. Betin-Can, and O. Demirörs, "A survey of software engineering practices in Turkey," *Journal of Systems and Software*, vol. 108, pp. 148–177, 2015, doi: 10.1016/j.jss.2015.06.036.

[75] D. Smite, N. B. Moe, G. Levinta, and M. Floryan, "Spotify guilds: how to succeed with knowledge sharing in large-scale agile organizations," *IEEE Software*, vol. 36, no. 2, pp. 51–57, 2019, doi: 10.1109/MS.2018.2886178.

[76] S. Shafiq *et al.*, "Towards scrum based agile framework for global software development teams," *Mehran University Research Journal of Engineering and Technology*, vol. 38, no. 4, pp. 979-998, 2019, doi: 10.3316/informit.729896454477135.

[77] M. Paasivaara, "Adopting SAFe to scale agile in a globally distributed organization," *in 2017 IEEE 12th International Conference on Global Software Engineering (ICGSE)*, 2017, pp. 36–40. doi: 10.1109/ICGSE.2017.15.

[78] 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.

[79] A. Khalid, S. A. Butt, T. Jamal, and S. Gochhait, "Agile scrum issues at large-scale distributed projects: scrum project development at large," *International Journal of Software Innovation*, vol. 8, no. 2, pp. 85–94, Apr. 2020, doi: 10.4018/IJSI.2020040106.

[80] M. Tanner and M. E. Dauane, "The use of kanban to alleviate collaboration and communication challenges of global software development," *Issues in Informing Science and Information Technology (IISIT)*, vol. 14, pp. 177–197, 2017, doi: 10.28945/3716.

[81] N. Tripathi *et al.*, "Scaling kanban for software development in a multisite organization: challenges and potential solutions," *in Agile Processes in Software Engineering and Extreme Programming*, 2015, pp. 178–190. doi: 10.1007/978-3-319-18612-2_15.

[82] M. O. Ahmad, D. Dennehy, K. Conboy, and M. Oivo, "Kanban in software engineering: a systematic mapping study," *Journal of Systems and Software*, vol. 137, pp. 96–113, 2018, doi: 10.1016/j.jss.2017.11.045.

[83] H. Alaidaros, M. Omar, and R. Romli, "The state of the art of agile kanban method: challenges and opportunities," *Independent Journal of Management & Production (IJMP)*, vol. 12, no. 8, pp. 2535–2550, Dec. 2021, doi: 10.14807/ijmp.v12i8.1482.

[84] L. Layman et al., "Essential communication practices for extreme programming in a global software development team," *Information and Software Technology*, vol. 48, no. 9, pp. 781–794, 2006, doi: 10.1016/j.infsof.2006.01.004.

[85] J. Erickson, K. Lyytinen, and K. Siau, "Agile modeling, agile software development, and extreme programming: the state of research," *Journal of Database Management*, vol. 16, no. 4, pp. 88–100, Oct. 2005, doi: 10.4018/jdm.2005100105.

[86] M. Marinho, R. Camara, and S. Sampaio, "Toward unveiling how safe framework supports agile in global software development," *IEEE Access*, vol. 9, pp. 109671–109692, 2021, doi: 10.1109/ACCESS.2021.3101963.

[87] A. Akhtar, B. Bakhtawar, and S. Akhtar, "Extreme programming vs scrum: a comparison of agile models," *International Journal of Technology, Innovation and Management (IJTIM)*, vol. 2, no. 2, Oct. 2022, doi: 10.54489/ijtim.v2i2.77.

[88] K. Schmid, "Challenges and solutions in global requirements engineering-a literature survey," *in Software Quality. Model-Based Approaches for Advanced Software and Systems Engineering*, 2014, pp. 85–99. doi: 10.1007/978-3-319-03602-1_6.

[89] R. Anwar *et al.*, "Systematic literature review of knowledge sharing barriers and facilitators in global software development organizations using concept maps," *IEEE Access*, vol. 7, pp. 24231–24247, 2019, doi: 10.1109/ACCESS.2019.2895690.

[90] M. A. Akbar *et al.*, "Requirements change management challenges of global software development: an empirical investigation," *IEEE Access*, vol. 8, pp. 203070–203085, 2020, doi: 10.1109/ACCESS.2020.3035829.

[91] R. K. Gupta *et al.*, "Challenges in adopting continuous delivery and devops in a globally distributed product team: a case study of a healthcare organization," *in International Conference on Global Software Engineering (ICGSE)*, 2019, pp. 30–34. doi: 10.1109/ICGSE.2019.00020.

[92] A. A. Khan and J. Keung, "Systematic review of success factors and barriers for software process improvement in global software development," *IET software*, vol. 10, no. 5, pp. 125–135, 2016, doi: 10.1049/iet-sen.2015.0038.

[93] A. Nguyen-Duc *et al.*, "The impact of global dispersion on coordination, team performance and software quality-a systematic literature review," *Information and Software Technology*, vol. 57, pp. 277–294, 2015, doi: 10.1016/j.infsof.2014.06.002.

[94] H. Lamsellak *et al.*, "Pre-planning process model in agile global software development," *in Innovations in Smart Cities Applications Volume 5*, 2022, pp. 393–400. doi: 10.1007/978-3-030-94191-8_31.

[95] H. Lamsellak *et al.*, "Release planning process model in agile global software development," *in International Conference on Advanced Intelligent Systems for Sustainable Development*, 2023, pp. 756–762. doi: 10.1007/978-3-031-26384-2_66.

[96] H. Lamsellak *et al.*, "Agile practices in iteration planning process of global software development," *in International Conference on Advanced Intelligent Systems for Sustainable Development*, 2023, pp. 302–309. doi: 10.1007/978-3-031-35251-5_29.

[97] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering," *presented at the 12th International Conference on Evaluation and Assessment in Software Engineering (EASE)*, 2008. doi: 10.14236/ewic/EASE2008.8.

[98] D. Budgen and P. Brereton, "Performing systematic literature reviews in software engineering," *in Proceedings of the 28th international conference on Software engineering*, Shanghai China: ACM, May 2006, pp. 1051–1052. doi: 10.1145/1134285.1134500.

[99] W. R. Fitriani *et al.*, "Challenges in agile software development: a systematic literature review," *in 2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2016, pp. 155–164. doi: 10.1109/ICACSIS.2016.7872736.

[100] I. Richter *et al.*, "Problems in agile global software engineering projects especially within traditionally organised corporations: [an exploratory semi-structured interview study]," *C3S2E '16*, 2016, pp. 33–43. doi: 10.1145/2948992.2949019.

[101] S. Bick *et al.*, "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, Oct. 2018, doi: 10.1109/TSE.2017.2730870.

[102] H. Noor *et al.*, "Identifying communication issues contributing to the formation of chaotic situation: an AGSD view," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 2, 2021, doi: 10.14569/IJACSA.2021.0120268.

[103] R. Sinha *et al.*, "SWOT: strength, weaknesses, opportunities, and threats for scaling agile methods in global software development," *Innovations in Software Engineering Conference*, 2020, pp. 1–10. doi: 10.1145/3385032.3385037.

[104] A. Rasheed *et al.*, "Requirement engineering challenges in agile software development," *Mathematical Problems in Engineering*, vol. 2021, pp. 1–18, May 2021, doi: 10.1155/2021/6696695.

[105] N. K. Kamaruddin *et al.*, "Chaos issues on communication in agile global software development," *In 2012 IEEE Business, Engineering and Industrial Applications Colloquium (BEIAC)*, 2012, pp. 394-398, doi: 10.1109/BEIAC.2012.6226091.

[106] M. Berntzen *et al.*, "Coordination strategies: managing inter-team coordination challenges in large-scale agile," *in Agile Processes in Software Engineering and Extreme Programming*, 2021, pp. 140–156. doi: 10.1007/978-3-030-78098-2_9.

[107] T. Dingsoyr and D. Smite, "Managing knowledge in global software development projects," *IT Professional*, vol. 16, no. 1, pp. 22–29, 2014, doi: 10.1109/MITP.2013.19.

[108] T. J. Gandomani *et al.*, "The impact of inadequate and dysfunctional training on agile transformation process: a grounded theory study," *Information and Software Technology*, vol. 57, pp. 295–309, 2015, doi: 10.1016/j.infsof.2014.05.011.

[109] S. Amjad *et al.*, "Calculating completeness of agile scope in scaled agile development," *IEEE Access*, vol. 6, pp. 5822–5847, 2018, doi: 10.1109/ACCESS.2017.2765351.

[110] M. Usman, R. Britto, L. O. Damm, and J. Börstler, "Effort estimation in large-scale software development: an industrial case study," *Information and Software Technology*, vol. 99, pp. 21–40, 2018, doi: 10.1016/j.infsof.2018.02.009.

[111] O. Sievi-Korte, S. Beecham, and I. Richardson, "Challenges and recommended practices for software architecting in global software development," *Information and Software Technology*, vol. 106, pp. 234–253, 2019, doi: 10.1016/j.infsof.2018.10.008.

[112] R. Jain and U. Suman, "A project management framework for global software development," *SIGSOFT Software Engineering Notes*, vol. 43, no. 1, pp. 1–10, Mar. 2018, doi: 10.1145/3178315.3178329.

[113] A. Manjavacas, A. Vizcaíno, F. Ruiz, and M. Piattini, "Global software development governance: Challenges and solutions," *Journal of Software: Evolution and Process*, vol. 32, no. 10, p. e2266, 2020, doi: 10.1002/smr.2266.

[114] F. Almeida *et al.*, "Challenges and facilitators practices for knowledge management in large-scale scrum teams," *Journal of Information Technology Case and Application Research*, vol. 21, no. 2, pp. 90–102, Apr. 2019, doi: 10.1080/15228053.2019.1637087.

[115] A. Putta *et al.*, "Benefits and challenges of adopting the scaled agile framework (SAFe): preliminary results from a multivocal literature review," *in Product-Focused Software Process Improvement*, 2018, pp. 334–351. doi: 10.1007/978-3-030-03673-7_24.

[116] A. Ampatzoglou, A. Ampatzoglou, A. Chatzigeorgiou, and P. Avgeriou, "The financial aspect of managing technical debt: a systematic literature review," *Information and Software Technology*, vol. 64, pp. 52–73, 2015, doi: 10.1016/j.infsof.2015.04.001.

[117] V. Garousi and M. V. Mäntylä, "When and what to automate in software testing? A multi-vocal literature review," *Information and Software Technology*, vol. 76, pp. 92–117, 2016, doi: 10.1016/j.infsof.2016.04.015.

[118] V. Alves, N. Niu, C. Alves, and G. Valença, "Requirements engineering for software product lines: A systematic literature review," *Information and Software Technology*, vol. 52, no. 8, pp. 806–820, 2010, doi: 10.1016/j.infsof.2010.03.014.

[119] M. Khurum and T. Gorschek, "A systematic review of domain analysis solutions for product lines," *Journal of Systems and Software*, vol. 82, no. 12, pp. 1982–2003, 2009, doi: 10.1016/j.jss.2009.06.048.

[120] C. Catal and B. Diri, "A systematic review of software fault prediction studies," *Expert Systems with Applications*, vol. 36, no. 4, pp. 7346–7354, 2009, doi: 10.1016/j.eswa.2008.10.027.

[121] L. B. Lisboa, V. C. Garcia, D. Lucrédio, E. S. De Almeida, S. R. L. De Meira, and R. P. M. De Fortes, "A systematic review of domain analysis tools," *Information and Software Technology*, vol. 52, no. 1, pp. 1–13, 2010, doi: 10.1016/j.infsof.2009.05.001.

[122] L. Marchezan, E. Rodrigues, W. K. G. Assunção, M. Bernardino, F. P. Basso, and J. Carbonell, "Software product line scoping: a systematic literature review," *Journal of Systems and Software*, vol. 186, p. 111189, 2022, doi: 10.1016/j.jss.2021.111189.

[123] A. E. Chacón-Luna, A. M. Gutiérrez, J. A. Galindo, and D. Benavides, "Empirical software product line engineering: A systematic literature review," *Information and Software Technology*, vol. 128, p. 106389, 2020, doi: 10.1016/j.infsof.2020.106389.

BIOGRAPHIES OF AUTHORS

Hajar Lamsellak Is currently a PhD student and member of the Laboratory Smart Information, Communication & Technologies (SmartICT Lab) at the National School of Applied Sciences at Mohammed First University, Oujda, Morocco (2019-). She has more than four years of experience working as a Quality Assurance (QA) engineer for SQLI Morocco company. She earned her degree in software engineering from it in 2019 and became a State engineer. Agile software development, agile planning, agile methodologies and practices, global software development, project management, and global software management are some of her interests. She can be contacted at email: lamsellak.hajar@ump.ac.ma.

Mohammed Ghaouth Belkasmi Is an associate professor at the National School of Applied Sciences of Oujda, with more than a decade of experience in supporting students towards the training of engineers in software development. Specializing in IT project management, NLP and network security, he founded the Smart Management Solutions (SMS) research team. And at the head of this team, he supervises numerous theses in related research fields. This team is also part of the Smart Information & Communication Technologies (SmartICT) Laboratory, which MG. Belkasmi is the co-founder. He can be contacted at email: m.belkasmi@ump.ac.ma.