Software process improvement in Scrum teams: A review of the literature

Mejora de procesos de software en equipos Scrum: Una revisión de la literatura

Héctor Cornide Reyes[1]* https://orcid.org/0000-0002-9130-184X
Agustín Astudillo[1] https://orcid.org/0009-0007-7016-6064
Eduardo Durán[1] https://orcid.org/0009-0006-0350-2665
Diego Monsalves[2] https://orcid.org/0000-0001-6780-3754

Recibido 09 de diciembre de 2024, aceptado 22 de mayo de 2025
Received: December 09, 2024 Accepted: May 22, 2025

ABSTRACT

In the Global Software Development (GSD) context, adopting agile methodologies, especially Scrum, has significantly transformed software delivery by emphasizing short iterations, continuous feedback, and adaptability, thus improving productivity and software quality. However, implementing Software Process Improvement (SPI) within Scrum teams still presents significant challenges, such as resistance to change, aligning agile practices with the requirements of standards like ISO/IEC 29110 and CMMI, and the additional administrative burdens these entail. Our primary objective is to identify the SPI practices Scrum teams are considered. We are also interested in identifying the difficulties in implementing SPI practices that have been reported. After analyzing the selected primary studies, we identified the most common SPI practices used by Scrum teams. The critical success factors identified were team and stakeholder commitment, a culture of continuous improvement, effective communication, and the use of tools and metrics to measure progress. Despite the existing challenges, the perception of SPI initiatives among Scrum team members varies. Many appreciate the software quality and structure improvement, but some see a possible incompatibility with agile practices and increased administrative workload. Our research concludes that the benefits of SPI practices justify the effort and that success depends on adaptability, continuous training, empirical improvement, and active team involvement.

Keywords: Software process improvement, Scrum, software projects.

RESUMEN

En el contexto del Desarrollo Global de Software (DGS), la adopción de metodologías ágiles, especialmente Scrum, ha transformado significativamente la entrega de software al enfatizar iteraciones cortas, retroalimentación continua y adaptabilidad, mejorando así la productividad y la calidad del software. Sin embargo, la implementación de la Mejora de Procesos de Software (MPS) dentro de los equipos Scrum aún sigue presentando retos importantes, tales como: la resistencia al cambio, la alineación de las prácticas ágiles con los requisitos de normas como ISO/IEC 29110 y CMMI y las cargas administrativas adicionales que esto implica. Nuestro principal objetivo es identificar las prácticas SPI que están siendo consideradas por los equipos Scrum. También nos interesa identificar las dificultades en la implementación de prácticas SPI que han sido reportadas. Tras analizar los estudios primarios seleccionados, logramos identificar las prácticas de SPI más comunes utilizados por los equipos Scrum. Los factores críticos de

[1] Universidad de Atacama. Departamento de Ingeniería Informática y Ciencias de la Computación. Copiapó, Chile. E-mail: hector.cornide@uda.cl; agustin.astudillo@uda.cl; eduardo.duran@uda.cl
[2] Universidad de Valparaíso. Escuela de Ingeniería Informática. Valparaíso, Chile. E-mail: diego.monsalves@uv.cl
* Autor de correspondencia: hector.cornide@uda.cl

DOI:10.64966/ingeniare.v33.10

éxito identificados fueron el compromiso del equipo y de las partes interesadas, una cultura de mejora continua, una comunicación eficaz y el uso de herramientas y métricas para medir el progreso. A pesar de los desafíos que aún existen, la percepción de las iniciativas SPI entre los miembros del equipo Scrum es variada. Muchos aprecian la mejora de la calidad y la estructura del software, pero algunos ven una posible incompatibilidad con las prácticas ágiles y un aumento de la carga de trabajo administrativo. Nuestra investigación concluye que los beneficios de las prácticas SPI justifican el esfuerzo, y que el éxito depende de la adaptabilidad, la formación continua, la mejora empírica y la participación activa del equipo.

Palabras clave: Mejora de procesos de Software, Scrum, proyectos de software.

INTRODUCTION

In the Global Software Development (GSD) context, adopting agile methodologies has significantly transformed how development teams approach software delivery. Among these methodologies, Scrum stands out for its focus on short iterations, continuous feedback, and adaptability to changes, which has been shown to improve productivity and software quality. Software Process Improvement (SPI) is an essential practice to optimize development processes for better quality, efficiency, and team satisfaction results.

The SPI practices commonly implemented by teams using Scrum include iterative planning, frequent reviews, and retrospectives. These practices allow teams to assess and adjust their work methods continuously. According to Díaz, Garbajosa, and Calvo-Manzano [1], integrating Scrum with the CMMI level 2 model has improved project management, planning, and requirements control. The Scrum methodology facilitates the capture and management of requirements through user stories (US), iterative planning, and product increment reviews, allowing for greater flexibility and adaptability to changes [1].

Several critical success factors have been identified to implement SPI initiatives in Scrum teams effectively. These include team and stakeholder commitment, an organizational culture that favors continuous improvement and learning, and the teams' ability to self-organize and make informed decisions [2]. Scrum's adaptability to different environments and its flexibility to integrate continuous improvement practices are key aspects contributing to the success of SPI initiatives [3].

Scrum team members' perception regarding the effectiveness of SPI initiatives is generally favorable when these practices are properly implemented. Teams that have adopted agile practices report improved collaboration, job satisfaction, and final product quality. Scrum retrospectives and daily meetings provide a platform for constant feedback and identification of areas for improvement, strengthening team cohesion and fostering a culture of continuous improvement [4].

Software Process Improvement (SPI) is a tool to optimize software development and a crucial component to ensure the long-term quality and sustainability of projects. The literature review suggests that implementing SPI in Scrum environments improves operational efficiency and strengthens the culture of collaboration and innovation within teams; this is because SPI practices, such as regular reviews and retrospectives, promote an environment of continuous improvement and collective learning, where each team member can actively contribute to process refinement [5].

On the one hand, implementing SPI in Scrum environments requires a deep understanding of team dynamics and the human factors influencing the adoption of new practices. The literature highlights the importance of effective communication and the participation of all team members to overcome resistance to change and ensure a smooth transition to new methodologies. Adaptability and willingness to experiment and adjust practices as necessary are critical aspects determining the success of SPI initiatives [6].

On the other hand, team member's perception of the effectiveness of SPI initiatives is closely related to how these practices are integrated into their daily workflow. Studies show that teams perceiving SPI practices as an additional burden or an interruption to their work tend to resist their implementation. Therefore, SPI initiatives must be designed and

implemented to align with the team's objectives and integrate seamlessly into their daily routines [4].

Regarding persistent issues over time, despite advances in software engineering, some can be identified, such as lack of communication and coordination, excessive workload, lack or scarcity of qualified human resources, organizational policies and cultures, and resistance to change [7], [8].

This study analyzed how Software Process Improvement (SPI) initiatives are implemented and evaluated in teams using the Scrum methodology. The expected outcomes of this study are to identify the most commonly implemented SPI practices by teams using Scrum, determine the critical success factors contributing to the effective implementation of SPI initiatives in Scrum teams, and analyze the challenges Scrum teams face when implementing SPI initiatives and how these teams overcome those challenges.

RELATED WORK

In the scientific literature, finding studies compiling the Scrum team's most commonly implemented SPI initiatives is challenging. However, in the study by Kouzari, Gerogiannis, Stamelos, and G. Kakarontzas e[9], the authors present a literature review aimed at identifying the critical success factors and barriers of lightweight SPI for SMEs (Small and Medium-sized Enterprises) that apply agile development processes. Three research questions were posed, the most prominent being RQ2: What are the main Critical Success Factors and Barriers involved in SPI? The identified success factors were: i. Top management commitment, ii. Staff participation, iii. Training, iv. Resources, v. Process Action Teams, vi. Staff experience, vii. Guidance, viii. Review and feedback, ix. Implementation methodology, x. Supervision, xi. Communication, xii. Return on Investment (ROI), and xiii. SPI knowledge, among other less relevant factors, is mentioned. Also mentioned are barriers that do not commonly appear in the literature, such as a) Existence of organizational policy, b) Lack of support, c) Negative experiences, d) Necessary paperwork, e) Inadequate or non-existent tools, f) High training costs, g) ROI produced in medium to long-term, h) Deficient project management activities.

Another study that maps a specific SPI model applied in Scrum is by [10]. This study aims to present a mapping between CMMI and Scrum to identify the main gaps between both approaches and observe how organizations adopt complementary practices to make them more compatible. The gaps mentioned include CMMI being applied at an organizational level, whereas Scrum does not mention this; thus, the organizational-level processes must follow CMMI. Another gap is that Scrum does not cover all the specific practices of the project management area of CMMI. Additionally, in Scrum, risks are identified but do not establish response strategies or a mitigation plan for critical risks, so risk management should prioritize CMMI. Moreover, Scrum does not contain practices such as configuration management and product quality assurance.

In contrast, these areas are important for CMMI, so support processes should be based on CMMI. It is concluded that Scrum practices can complement CMMI's project management process areas, especially in small and medium-sized enterprises, improving process agility and helping define a new project management framework based on both methodologies. Additionally, some adaptations in Scrum, mainly related to agile risk management, issue management, and estimation methods, make it much more compatible (or fully compatible) with CMMI's project management process areas.

Another study, which presents a systematic literature review of a specific SPI model in agile development, is by Ferdinansyah and Purwandari [11]. Its objective is to provide software engineering professionals with an initial idea of the possible obstacles to formulating a software process that complies with CMMI and Agile Development. A systematic literature review was conducted, and one of the notable research questions of this study is RQ1: What are the problems and obstacles associated with the joint adaptation of the CMM framework and the Agile methodology? To answer RQ1, the study points out classification through categories for the problems, with "Knowledge and culture" being the most frequently appearing category; this implies that the lack of knowledge or practical experience associated with CMMI or agile development (or both concepts) is the main obstacle to joint implementation. Other categories include "organizational" and "Agile cannot achieve CMMI alone," which present the

challenge of a limitation in agile development, especially at the organizational level. Agile focuses on software delivery projects, while CMMI focuses on improvement at the organizational level. Another category presented is "Control vs. Agility," where difficulties relate to balancing Control, which is one of the goals of CMMI implementation, and the flexibility promoted by Agile; this is consistent with the issues in the category "Documentation, contracts, and planning," where paperwork and planning are considered obstacles to joint application. Documentation, contracts, and plans are considered project control tools. The issues in the categories "Organization/team resistance," "Need for additional resources, effort, and time," and "Project risk" indicate that implementing new rules or standards in an organization will always provoke resistance and require time and effort to establish.

Furthermore, Moe [12] focuses on the main barriers and challenges observed by the authors in empirical studies at the team and organizational levels when improving the software development process in agile teams. This study poses the research question: What are the challenges of software process improvement in agile software development? A case study of SPI was implemented in three companies that conduct agile software development. The study concludes that the challenges of SPI in agile software development are the issues of increasing redundancy to create conditions for the team to self-manage, learn to learn, and perceive agile software development improvement as a significant long-term organizational change project.

The article by Mathrani, Wickramasinghe, and Jayamaha [13] presents a case study that evaluates the documentation requirements for ISO 9001 compliance in Scrum projects. The study's research questions are RQ1: What level of documentation should be adopted in Scrum projects to maintain ISO 9001 compliance? RQ2: How do software professionals perceive the current ISO 9001 documentation requirements for Scrum projects?. How do software professionals perceive the current ISO 9001 documentation requirements for Scrum projects? Regarding RQ1, the study describes that the SPI practices implemented by the teams include product backlogs, international specifications, standards, and regulations, as well as business flows to comply with ISO 9001. The identified success factors include adapting documentation practices to ISO 9001 requirements, such as maintaining detailed records of meetings, requirement specifications, test and validation records, and audit and compliance reports. Effective management of this documentation is crucial for meeting quality standards without excessively interfering with agile methodologies.

Regarding RQ2, the identified challenges include the tension between Scrum's "lightweight" methodologies and the extensive and detailed documentation approaches required by ISO 9001; this can lead to documentation overload and difficulties in maintaining accurate and complete documentation. The study collects software professionals' perceptions of ISO 9001 documentation in Scrum projects, noting that some documentation tasks are considered excessive and burdensome, which can interfere with the team's creativity and agility. These perceptions highlight the need to balance ISO 9001 documentation requirements with Scrum's flexibility and efficiency. The study also identifies future research topics, such as effectively integrating agile management tools with formal process frameworks like ISO 9001 and seeking best practices for achieving an optimal balance between agile flexibility and structured control. Although the text does not explicitly mention all these future research directions, they can be inferred from the discussions about challenges and recommendations provided in the study. This balance refers to finding a middle ground where the Scrum team's agility and responsiveness can be maintained while meeting the rigorous documentation and quality control requirements imposed by ISO 9001.

METHODOLOGY

The main goal of this work is to analyze the scientific evidence that informs about the challenges associated with using traditional software process improvement models in agile software development. The literature review process was conducted to achieve this objective based on the guidelines proposed by [14], [15], and it is consistent with other recently published rapid literature reviews [16], [17]. The process was conducted through the following steps: 1) statement of the research questions, 2) search process, 3) selection of studies, and 4) analysis of results. All these steps are described below.

Research questions

The research questions that drive this literature review are:

1. RQ1: What are the Software Process Improvement (SPI) practices most implemented by teams using Scrum?
2. RQ2: What are the critical success factors for effective SPI implementation in Scrum teams?
3. RQ3: What challenges do Scrum teams face when implementing SPI initiatives, and how do they overcome them?
4. RQ4: What are Scrum team members' perceptions of the effectiveness of SPI initiatives?

Search process

The citation databases used were SCOPUS and Web of Science (WoS), while the scientific publication databases used were ACM Digital Library and IEEE Xplore. This selection is mainly due to the reputation of these databases in the discipline and the fact that we have full access to the published material. Table 1 details the method used to construct the search string. For this purpose, we use the PICOC method, whose acronyms represent the criteria that drive the process [14], [18], [19]. The PICOC method is structured to formulate research questions and construct efficient search chains. PICOC is an acronym for five key elements: ***P*** de Population; ***I*** de Intervention; ***C*** de Comparison; ***O*** de Outcomes; and ***C*** de Context. For the stated scope of this research, the Comparison and Context components do not apply (NA). For the stated scope of this research, the Comparison and Context components do not apply (NA); this is because this research aims to explore the general overview in an area where the literature is limited and because we have determined that these elements are not critical to answering the research questions. The clarity in these elements facilitates systematic and comprehensive searches of databases and other information sources.

Based on the PICOC method, the following search string was developed: *(organization OR company OR Institution OR industry) AND (scrum OR agile) AND ("process improvement") AND (practices OR Guidelines OR "critical factors" OR implementation OR challenge OR challenges OR evaluation)*. This search string was validated with a set of papers we identified and used as a control group. The research was initially conducted in March and last updated in mid-May 2024.

The inclusion/exclusion criteria were as follows:

- Published between 2018 and 2023.
- Contains information about SPI and Scrum.
- Only written in English.

Selection of primary studies

The data extraction form was developed with the following fields: paper Title, Year, DOI; Published in (Journal, Conference); Study Type (Quantitative, Qualitative, Mixed); Research Method (Literature Review, Case Study, Interviews, Questionnaires, Observation); Objective; Main Findings; Main Conclusions; Future Work; Tributes to RQ1 (Yes/No); Evidence RQ1; Tributes to RQ2 (Yes/No); Evidence RQ2; Tributes to RQ3 (Yes/No); Evidence RQ3; Tributes to RQ4 (Yes/No); Evidence RQ4; research field (industrial, business, educational, university); Number of people; Number of Women; Number of Men.

The methodology for searching and selecting papers was structured into three successive stages:

- First Stage (1F): We began by reviewing all titles and keywords of the papers obtained

Table 1. Construction of the search string using the PICOC method.

Population	Intervention	Comparision	Outcomes	Context
organization/company/ institution/Industry	agile/scrum process improvement	NA	practices/guidelines/ critical factors/implementation/ challenge /evaluation	NA
(organization OR company OR Institution OR industry)	(scrum OR agile) AND ("process improvement")	NA	(practices OR Guidelines OR "critical factors" OR implementation OR challenge OR challenges OR evaluation)	NA

from each database. Subsequently, duplicates were removed to ensure the uniqueness of the documents.

- Second Stage (2F): Next, we detailed the abstracts of the papers that passed the first filter. This step allowed us to make a preliminary assessment of each paper's relevance and pertinence.
- Third Stage: Finally, the papers selected after the second filter were downloaded, read in their entirety, and recorded in a form designed in Microsoft Excel, following the established protocol. This stage involved an exhaustive review to determine their final inclusion in the study.

After executing the search string in each consulted database, we obtained the following initial results: 46 papers from Web of Science (WoS), 22 from SCOPUS, 39 from IEEE Xplore, and 11 from ACM Digital Library. Subsequently, we applied the previously described filters (1F and 2F), which resulted in a more refined selection: 26 papers from WoS, 10 from SCOPUS, 18 from IEEE Xplore, and four from ACM Digital Library. Finally, we selected 27 primary studies for detailed analysis and discussion (Table 2).

Analysis of results

Figure 1 provides a clear and detailed chronological distribution of the selected primary studies over the years. This graph is particularly revealing, as it visually demonstrates the growing interest within the scientific community in digital transformation and Industry 4.0 topics. A notable increase in the number of studies conducted in recent years is highlighted, reflecting the growing recognition of the importance of these topics in the current context.

Table 3 offers an integrated and organized view of how the selected primary studies align with the previously defined research questions. This table shows the correspondence between the research questions and the primary studies to facilitate a deeper

Table 2. Selected primary studies (PS) and DOI.

Reference	Title	DOI
[20]	A case study of using the hybrid model of scrum and Six Sigma in software development	10.11591/ijece.v11i6.pp5342-5350
[21]	A gamification solution for improving Scrum adoption	10.1007/s10664-020-09816-9
[22]	A Theory of Scrum Team Effectiveness	10.1049/iet-sen.2018.5096
[23]	Agile Process Improvement in Retrospectives	10.1109/ICSE-Companion.2019.00063
[24]	Agile software process improvement: a collaborative game toolbox	10.1049/iet-sen.2018.5096
[25]	A Means-Ends Design of SCRUM+: an agile-disciplined balanced SCRUM enhanced with the ISO/IEC 29110 Standard	10.1007/978-3-319-69341-5_2
[26]	Agile-CMMI V2.0 alignment: Bringing to light the agile artifacts pointed out by CMMI	10.1016/j.csi.2021.103610
[27]	Blending agile methodologies to support automotive SPICE compliance	10.1002/smr.2391
[28]	Business process management approach for improving agile software process and agile maturity	10.1002/smr.2331
[29]	Comparative effects of knowledge-based antecedents in different realms of CMMI-based software process improvement success	10.1016/j.csi.2021.103599
[30]	Combining CMMI-Specific Practices with Scrum Model to Address Shortcomings in Process Maturity	10.1007/978-3-030-14118-9_88
[31]	Evaluation and Recommendation for Scrum Implementation Improvement with Hybrid Scrum Maturity Model: A Case Study of A New Telco Product	10.1109/ICIMCIS51567.2020.9354311
[32]	Feedback in Scrum: Data-Informed Retrospectives	10.1109/ICSE-Companion.2019.00081
[33]	FLEX-RCA: a lean-based method for root cause analysis in software process improvement	10.1007/s11219-018-9408-8
[34]	Fuzzy AHP-based prioritization and taxonomy of software process improvement success factors in global software development	10.1016/j.asoc.2019.105648

Reference	Title	DOI
[35]	How has SPI changed in times of agile development? Results from a multi-method study	10.1002/smr.2182
[36]	Implementing ISO/IEC 29110 to reinforce four very small entities of Mexico under an agile approach	10.1049/iet-sen.2019.0040
[37]	Lean and agile software process improvement in traditional and agile environments	10.1002/smr.1986
[38]	Measuring and Improving Agile Processes in a Small-Size Software Development Company	10.1109/ACCESS.2020.2990117
[39]	Organizational training in agile settings under the lens of CMMI V2.0	10.1002/smr.2502
[40]	Perceived Benefits and Challenges of Implementing CMMI on Agile Project Management: A Systematic Literature Review	10.14569/ijacsa.2024.0150122
[41]	Process Improvement Framework for DevOps Adoption in Software Development	10.1109/SCSE59836.2023.10214992
[42]	Reconciliation of scrum and the project management process of the ISO/IEC 29110 standard-Entry profile-an experimental evaluation through usability measures	10.1007/s11219-021-09552-3
[43]	Scrum Challenges: An Agile Process Reengineering in Software Engineering	10.35940/ijitee.D1567.029420
[44]	SPI is Dead, isn't it? Clear the Stage for Continuous Learning!	10.1109/ICSSP.2019.00012
[45]	Supporting the deployment of ISO-based project management processes with agile metrics	10.1016/j.csi.2019.103405
[46]	The Visualization of ISO/IEC29110 on SCRUM under EPF Composer	10.3390/info12050190

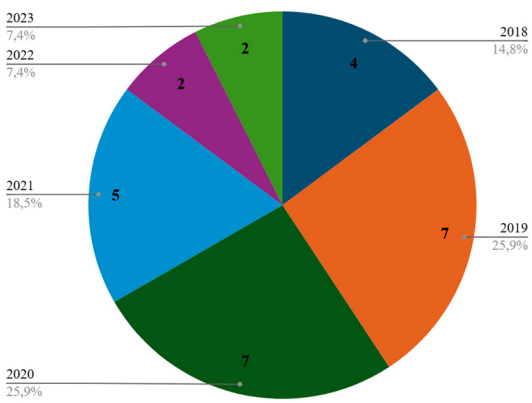

Figure 1. Distribution of primary studies by year.

understanding of the focus areas and specific aspects each study addresses concerning the posed questions.

Furthermore, Table 4 establishes a relationship between the research questions and the research methods used in the selected primary studies.

DISCUSSION OF RESULTS

A process is implemented to analyze the information and facilitate a detailed output discussion. This process involved analyzing and categorizing the selected primary studies based on their alignment with the precise research questions that led to our

Table 3. Matching research questions and primary studies.

Research Questions	References to primary studies
RQ1: What are the Software Process Improvement (SPI) practices most implemented by teams using Scrum?	[20], [21], [22], [23], [25], [26], [27], [28], [29], [30], [32], [33], [34], [36], [37], [38], [41], [44], [45], [46]
RQ2: What are the critical success factors for effective SPI implementation in Scrum teams?	[20], [22], [23], [25], [26], [27], [28], [29], [30], [32], [35], [36], [37], [39], [40], [43], [44], [45], [46]
RQ3: What challenges do Scrum teams face when implementing SPI initiatives and how do they overcome them?	[20], [22], [23], [26], [27], [28], [29], [31], [32], [35], [36], [37], [40], [42], [43], [44], [46]
RQ4: What are Scrum team members' perceptions of the effectiveness of SPI initiatives?	[23], [26], [27], [28], [40], [41]

Table 4. Matching research questions with research method used in primary studies.

Research Questions	Research method used		
	Literature review	Case study	Experimentation
RQ1: What are the Software Process Improvement (SPI) practices most implemented by teams using Scrum?	[26], [30], [34], [37], [41], [44], [45]	[28], [32], [36], [38], [46]	[27], [29]
RQ2: What are the critical success factors for effective SPI implementation in Scrum teams?	[26], [30], [35], [39], [40], [43], [45]	[20], [22], [23], [25], [28], [32], [36]	[27], [29], [31]
RQ3: What challenges do Scrum teams face when implementing SPI initiatives, and how do they overcome them?	[26], [35], [37], [40], [43]	[20], [23], [28], [32], [36], [46]	[27], [29], [31], [42]
RQ4: What are Scrum team members' perceptions of the effectiveness of SPI initiatives?	[26], [40], [41]	[23], [28], [46]	[27]

study. In doing so, we could understand how each study contributes individually to the overall body of knowledge and collectively assess the progress and trends in the field concerning our research questions.

RQ1: What are the Software Process Improvement (SPI) practices most implemented by teams using Scrum?

The SPI practices most commonly implemented by teams using Scrum typically focus on integrating agile practices with recognized quality frameworks. The following are the most relevant findings. In [46], the study analyzes the implementation of the ISO/IEC 29110 software development standard in Scrum environments for Very Small Entities (VSEs). These entities face challenges such as high user expectations, frequent changes in requirements, and the need for rapid deployment. The use of agile methodologies and software development standards is perceived as a promising solution to improve the quality of the development process. However, the authors note significant incompatibilities between the standards and the agile approach, as the standards require the creation of many quality artifacts, which does not align with the agile philosophy that prioritizes working software over documentation. The study analyzes ISO/IEC 29110 standard implementation cases developed specifically for VSEs in Scrum environments. It provides an Eclipse Process Framework (EPF) to effectively implement this standard in Scrum software development. The research highlights the need for an external consultant to guide the implementation of the standard in VSEs due to the lack of clear guidelines. Additionally, the study proposes a process model in EPF Composer that includes capability patterns and delivery processes, defining roles, tasks, work products, and guidelines necessary to achieve standard implementation. The analysis results indicate that using EPF Composer facilitates the visualization and adaptation of the ISO/IEC 29110 standard in Scrum environments, providing a practical guide for VSEs seeking to improve their software development processes.

In [32], the authors investigate how agile teams can continuously use the data generated during their development activities to improve their processes through data-informed retrospectives. The study is based on the hypothesis that development data, such as commits and tickets, contains valuable information about team collaboration and performance. This data can be used to identify and track process improvements. Through preliminary interviews with Scrum Masters and agile coaches, it was found that many companies already employ some form of data analysis to guide their process improvement efforts. However, the implementation and tracking of these improvements vary significantly among teams. The article highlights the importance of measuring the progress of improvements to gain a clear view of their effectiveness, suggesting that transparency and data analysis can strengthen Scrum retrospectives. A study of semi-structured interviews in multiple companies is proposed to understand better current process improvement practices and how development data can support these practices. The results of this research can help design tools and methodologies that optimize continuous improvement in agile teams, promoting a more empirical and data-driven approach to process management in the software industry. The SPI practices implemented by teams include analyzing development data, identifying deviations

in the executed process, and discussing possible improvements based on concrete data.

A very interesting work is presented in [44]. This article analyzes the evolution of SPI and its integration into projects based on continuous learning, contrasting with classical SPI approaches. It highlights that companies are increasingly adopting SPI activities in the project to optimize specific processes continuously. It is mentioned that agile and lean methods are shifting towards project-integrated SPI, moving away from extensive and prescriptive maturity models. The idea is proposed that continuous learning, facilitated by agile and lean methods, enriches classical SPI frameworks. It is argued that classical SPI remains relevant but is increasingly complemented by continuous learning activities. SPI implementation is suggested to evolve towards a continuous learning environment where SPI activities are integrated into projects to address specific problems agilely. For future research, areas such as revisiting SPI to focus on continuous learning, balancing formal SPI models with continuous learning, integrating specific project learnings into broader SPI initiatives, developing effective metrics to measure the success of continuous learning in SPI, and exploring the role of standards in implementing continuous learning strategies in SPI are proposed. These areas explore how continuous learning can improve SPI practices in an agile and constantly changing software development environment. The implemented practices are customer involvement, knowledge transfer between projects, realistic planning, artifact reuse, tool support, company certification, product quality improvement, and project speed improvement.

The article presented in [20] evaluates the implementation of a hybrid software development model combining Scrum and Six Sigma in a case study. The results indicate that this combination can improve team performance and customer satisfaction. However, an increase in the number of reworks, defects, and project duration was also observed. These authors justified these results as necessary to achieve higher quality and customer satisfaction, highlighting the importance of balancing agility and rigor in development processes. The SPI practices implemented include the combination of Scrum with Six Sigma, using processes such as DMAIC in each Sprint, the use of process maps to create Sprint backlogs, the determination of defect causes in review and retrospective meetings, and the prioritization of the product backlog according to customer-expected improvements.

The SPI practices implemented in [23] include performance monitoring tools, measurement systems, automated testing environments, and the conduct of Sprint retrospective meetings to assess and improve the Scrum process continuously. Meanwhile, [28] considers SPI practices to improve Scrum ceremonies, such as daily meetings, short iterations, Sprint planning, retrospectives, Sprint reviews, and testing techniques.

Incorporating of these practices helps Scrum teams align their processes with recognized quality standards, improve software efficiency and quality, and foster a culture of continuous improvement. Integrating these approaches can result in greater customer satisfaction, better risk management, and an improved ability to adapt to changes and challenges.

RQ2: What are the critical success factors for effective SPI implementation in Scrum teams?
From the experience gathered in other research with Scrum teams, we have identified that there is not always a positive view towards incorporating new elements into the existing Scrum framework. However, teams recognize that work must be oriented towards continuous improvement to increase the likelihood of projects success, be process. Scrum teams that incorporate standards such as CMMI (Capability Maturity Model Integration) or ISO/IEC 29110 seek to improve software quality, standardize processes, promote continuous improvement, increase customer confidence, enhance market competitiveness, improve risk management, and align development processes with organizational strategy. These standards provide frameworks of best practices that help ensure predictable and high-quality results, fostering a structured and coherent approach to software development, which is crucial for efficiency and customer satisfaction.

In [36], for example, it is stated that the success of SPI implementation in Very Small Entities (VSEs) through the ISO/IEC 29110 standard is based on a five-step method. First, identify the main problems that VSEs face in carrying out their projects through

meetings to gather detailed information about their activities. Second, identify and formalize the VSEs' software development processes, focusing on project management and its development processes; and third, map the current processes of VSEs with the fundamental processes of ISO/IEC 29110 to identify gaps. Fourth, select and adapt the practices provided by the ISO/IEC 29110 standard to the specific context of each VSE, analyzing how each practice can impact its current processes. Finally, review a project where the reinforced processes were implemented, providing feedback and resolving non-conformities. The execution of this five-step method, with regular meetings and teams integrated by external experts and members of each VSE, is crucial for the success of SPI implementation, ensuring continuous improvement adapted to the specific needs of VSEs.

In [23], the authors present a toolbox of collaborative games to improve agile software development processes, specifically in teams using Scrum. The games are designed to foster communication, cohesion, and team coordination during retrospective meetings, facilitating continuous process improvement. The toolbox includes 12 games requiring no prior preparation and is classified according to the group's development and meeting phase. Initial results show that using these games improves team motivation and performance.

The article presented by [22] proposes a theory on the effectiveness of Scrum teams based on seven years of research, including case studies and statistical analysis. Five key factors that determine the effectiveness of these teams were identified: responsiveness, stakeholder concern, continuous improvement, team autonomy, and management support. These factors were validated through a structural equation model based on data from approximately 5,000 developers and 2,000 Scrum teams. The research offers recommendations for organizations to support their Scrum teams and provides a generalizable theory on the effectiveness of these teams.

In [28], the authors identify success factors such as customer and stakeholder orientation, collaboration and planning, improved requirements engineering, and customer satisfaction. Similarly, [32] indicates success factors such as the use of development data to inform and track improvement steps, and the implementation of specific metrics and measures to track process progress and compliance.

RQ3: What challenges do Scrum teams face when implementing SPI initiatives and how do they overcome them?

Scrum teams face several challenges when implementing SPI initiatives, such as integrating CMMI or ISO/IEC 29110 standards. These challenges include resistance to change, aligning agile practices with standard requirements, additional administrative burdens, and potential loss of flexibility. Resistance to change may arise because team members are accustomed to their current methods and may see new practices as an unnecessary burden. The key to overcoming this is to involve all team members in the implementation process, provide adequate training, and distinctly communicate the expected benefits of the SPI initiatives. Several significant incompatibilities exist between the standards and the agile software development approach. For example, the need identified in the standards to create many quality artifacts does not align with agile philosophies. Since Agile focuses on working software over documentation, implementing Agile with standards can be challenging [46].

In [42], for example, the authors address integrating Scrum's agile practices with the project management standards of ISO/IEC 29110, designed for very small entities (VSEs). Large and medium-sized organizations use software process standards and models to achieve the Iron Triangle (time, cost, and quality). However, VSEs, due to their technical and economic limitations, tend to use agile practices. The study presents the experimental evaluation of Scrum + EPG (a combination of Scrum and the project management process of ISO/IEC 29110) versus Scrum EPG (an unmodified version of Scrum) using six usability metrics. Thirty-two academics and international professionals from Latin America, North America, and Asia-Pacific participated, using a within-subject design and Wilcoxon signed-rank tests. The main challenges mentioned include:

- Compatibility between Agility and Rigor: One of the main challenges is reconciling agile practices, which are inherently flexible and adaptive, with software standards and processes that are more rigorous and structured, such as

ISO/IEC 29110; this may require adjustments and complements to Scrum practices to meet the stricter requirements of the standards.
- Difficulties in Project Planning and Control: Implementing Scrum in combination with ISO/IEC 29110 standards can present difficulties in planning and project control; this includes risk management, project estimation, and progress tracking, which need to be improved to meet the standards.
- Configuration Management and Documentation: Another challenge is ensuring proper configuration management and software documentation, which are critical areas covered by ISO/IEC 29110 standards but may be less rigorous in Scrum's agile practices.
- Acceptance and Adoption of Improved Practices: Adopting improved SPI practices can face resistance within agile teams, which may perceive these practices as an additional burden or a threat to their flexible and rapid way of working.

In [37], the authors present the following challenges from the SPI perspective:

- Team Maturity: Team maturity is crucial for the effective implementation of SPI. Teams must progress gradually through maturity levels without attempting to skip several levels simultaneously.
- Definition of Team Product Responsibility: Defining the team's product responsibility is fundamental. Teams should focus on product outcomes and the procedures, methods, and tools to achieve them.
- Change Management: Change management is critical due to the resistance teams may have to new procedures. Agile organizations face challenges similar to traditional ones in this aspect, and teams must see explicit benefits in new procedures and accept them.
- Compliance with Certain Standards: Meeting security and other established standards is a significant challenge for Scrum teams, especially in industries where this is non-negotiable.

In general, the keys to effectively addressing the challenges of implementing SPI initiatives in Scrum can be overcome by involving all members, carefully aligning practices, automating administrative processes, and iteratively and evaluatively implementing the necessary changes.

RQ4: What are Scrum team members' perceptions of the effectiveness of SPI initiatives?

Scrum team members' perceptions of the effectiveness of Software Process Improvement (SPI) initiatives can be mixed. On the positive side, many value these initiatives, such as implementing CMMI or ISO/IEC 29110 standards, improving software quality, providing structure and clarity, and increasing the team's confidence and credibility with clients and other stakeholders. However, they can also perceive resistance to change, an increase in administrative burden, and a potential incompatibility with the agile nature of Scrum, which can cause frustration and reduce team morale. The key to improving this perception is to involve the team in its implementation, using tools that facilitate the integration of standards into the agile workflow and providing adequate training and clear communication about the expected benefits of SPI initiatives.

In this regard, we highlight [24], which presents the design of SCRUM+, an enhanced methodology that integrates SCRUM with the best practices of the ISO/IEC 29110 standard to balance agility with discipline. Using a means-ends analysis, SCRUM+ is proposed to help small software development entities improve the quality of both the process and the product without losing the agile essence. The results suggest that SCRUM+ can provide a more robust and disciplined framework that facilitates project management while maintaining the flexibility and efficiency characteristic of SCRUM. The team's perception of SPI is positive, as the improvements have contributed to greater cooperation and communication among team members and have shown progress in Scrum maturity.

In [26], it is indicated that the general perception is that alignment with CMMI should not mean "doing CMMI" but "doing Agile better," suggesting that some adjustments are necessary to align both approaches. Similarly, [27] indicates that the perception is positive, as agile methodologies such as Scrum and Kanban help meet SPICE requirements, improve project management, and facilitate adaptation to changes in software development. Finally, in [28], it is concluded that current agile practices are still evolving and maturing and benefit from training and education to improve.

CONCLUSIONS

Implementing Software Process Improvement (SPI) practices in Scrum teams presents significant benefits and challenges. The analyzed studies indicate that the most common SPI practices include integrating standards such as ISO/IEC 29110 and CMMI, using data-informed retrospectives, and adopting hybrid approaches like the combination of Scrum and Six Sigma. These practices help improve software quality, standardize processes, and foster a culture of continuous improvement, resulting in greater customer satisfaction, better risk management, and an enhanced ability to adapt to changes and challenges. Critical success factors for the implementation of SPI include identifying and formalizing processes, adapting specific practices to the team context, and using tools and metrics to measure the progress and effectiveness of improvements. Active team participation and management support are also crucial to overcome resistance to change and ensure the acceptance of new practices. Scrum teams face significant challenges when implementing SPI initiatives, such as resistance to change, additional administrative burden, and aligning agile practices with the standards' requirements. Overcoming these challenges requires clear communication of the benefits, adequate training, and careful integration of SPI practices into the agile workflow. The members' perception of a team regarding the effectiveness of SPI initiatives is mixed. While many value the improvements in software quality and the structure provided by the standards, others perceive a potential incompatibility with the agile nature of Scrum and an increase in administrative burden. This perception can be fundamentally improved by involving the team in the implementation and providing tools and training to integrate standards into the agile environment.

Although implementing SPI practices in Scrum teams can present challenges, the potential benefits in terms of quality, efficiency, and customer satisfaction justify the effort. The key to success lies in adaptability, continuous training, and a focus on empirical and data-driven improvement.

REFERENCES

[1] J. Diaz, J. Garbajosa, and J.A. Calvo-Manzano, "Mapping CMMI level 2 to Scrum practices: An experience report," in *Software Process Improvement*. EuroSPI 2009.Communications in Computer and Information Science, R.V. O'Connor, N. Baddoo, J. Cuadrago Gallego, R. Rejas Muslera, K. Smolander, and R. Messnarz, Eds. vol. 42, Heidelberg, Germany: Springer, 2009, pp. 93-104, doi: 10.1007/978-3-642-04133-4_8.

[2] 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," *Procedia Computer Science*, vol. 64, pp. 188-195, 2015, doi: 10.1016/j.procs.2015.08.480.

[3] M.L. Duc, J. Nedoma, P. Bilik, R. Martinek, L. Hlavaty, and N. T.P. Thao, "Application of fuzzy TOPSIS and harmonic mitigation measurement on lean six sigma: A case study in smart factory," *IEEE Access*, vol. 11, pp. 81577-81599, 2023, doi: 10.1109/ACCESS.2023.3299326.

[4] G. Nalepa, R.M. Fontana, S. Reinehr, and A. Malucelli, "Using agile approaches to drive software process improvement initiatives," in *Systems, Software and Services Process Improvement*. Communications in Computer and Information Science, A. Walker, R.V. O'Connor, and R. Messnarz, Eds. vol. 1060. Cham, Switzerland: Springer, 2019, pp. 495-506, doi: 10.1007/978-3-030-28005-5_38.

[5] F.J. Pino, O. Pedreira, F. García, M.R. Luaces, and M. Piattini, "Using Scrum to guide the execution of software process improvement in small organizations," *Journal of Systems and Software*, vol. 83, no. 10, pp. 1662-1677, Oct. 2010, doi:10.1016/j.jss.2010.03.077.

[6] J. Vogel and R. Telesko, "Derivation of an Agile Method Construction Set to Optimize the Software Development Process," *Journal of Cases on Information Technology*, vol. 22, no. 3, pp. 19-34, Jul. 2020, doi: 10.4018/JCIT.2020070102.

[7] E.A. Elfaiz and T. Raharjo, "The Obstacles of Software Process Improvement in Software House: A Systematic Literature Review and Empirical Study," *Journal of Physics: Conference Series*, vol. 1811, no. 1, Art. no. 012102, Mar. 2021, doi:10.1088/1742-6596/1811/1/012102.

[8] E. Monteiro Soares, S.R. Bezerra Oliveira, I. Souza Elgrably, and R. Henrique Barbosa

Monteiro, "Application of a gamified approach to learning in the treatment of problems in software process improvement: Analysis and discussion of results," *International Journal of Emerging Technologies in Learning (iJET)*, vol. 17, no. 11, pp. 242-296, Jun. 2022, doi:10.3991/ijet.v17i11.30671.

[9] E. Kouzari, V.C. Gerogiannis, I. Stamelos, and G. Kakarontzas, "Critical success factors and barriers for lightweight software process improvement in agile development: A literature review," in *10th International Joint Conference on Software Technologies (ICSOFT)*, Colmar, France, July 2015, pp. 1-9.

[10] Z. Lina and S. Dan, "Research on combining Scrum with CMMI in small and medium organizations," in *2012 International Conference on Computer Science and Electronics Engineering*, Hangzhou, China, Mar. 2012, pp. 554-557, doi:10.1109/ICCSEE.2012.477.

[11] A. Ferdinansyah and B. Purwandari, "Challenges in combining agile development and CMMI: A systematic literature review," in *2021 10th International Conference on Software and Computer Applications*, Kuala Lumpur, Malaysia: ACM, Feb. 2021, pp. 63-69, doi:10.1145/3457784.3457803.

[12] N.B. Moe, "Key challenges of improving agile teamwork," in *Agile Processes in Software Engineering and Extreme Programming*. Lecture Notes in Business Information Processing, H. Baumeister, and B. Weber, Eds., vol. 149, Heidelberg, Germany: Springer, 2013, pp. 76-90, doi: 10.1007/978-3-642-38314-4_6.

[13] A. Mathrani, S. Wickramasinghe, and N.P. Jayamaha, "An evaluation of documentation requirements for ISO 9001 compliance in scrum projects," *The TQM Journal*, vol. 34, no. 5, pp. 901-921, Nov. 2022, doi:10.1108/TQM-08-2020-0177.

[14] K. Petersen, S. Vakkalanka, and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update," *Information and Software Technology*, vol. 64, pp. 1-18, Aug. 2015, doi: 10.1016/j.infsof.2015.03.007.

[15] H. Snyder, "Literature review as a research methodology: An overview and guidelines," *Journal of Business Research*, vol. 104, pp. 333-339, Nov. 2019, doi:10.1016/j.jbusres.2019.07.039.

[16] H. Cornide-Reyes, C. Duran, S. Baltierra, F. Silva-Aravena, and J. Morales, "Improving UX in digital transformation projects through lean principles," in *Social Computing and Social Media*. Lecture Notes in Computer Science, A. Coman and S. Vasilache, Eds. vol. 14703, Cham, Switzerland: Springer, 2024, pp. 161-178, doi: 10.1007/978-3-031-61281-7_11.

[17] J. Morales, H. Cornide-Reyes, F. Silva-Aravena, J. Sepúlveda, and G. Muñoz, "Tourist eXperience and Use of Virtual Reality, Augmented Reality and Metaverse: A Literature Review," in *Social Computing and Social Media*, A. Coman and S. Vasilache, Eds., *Lecture Notes in Computer Science*, vol. 14703. Cham, Switzerland: Springer, 2024, pp. 205-221, doi:10.1007/978-3-031-61281-7_14.

[18] A. Carrera-Rivera, W. Ochoa, F. Larrinaga, and G. Lasa, "How-to conduct a systematic literature review: A quick guide for computer science research," *MethodsX*, vol. 9, Art. no. 101895, 2022, doi:10.1016/j.mex.2022.101895.

[19] C.M. da Costa Santos, C.A. da Matos Pimenta, and M.R. Cuce Nobre, "The PICO strategy for the research question construction and evidence search," *Revista Latino-Americana de Enfermagem*, vol. 15, no. 3, pp. 508-511, Jun. 2007, doi:10.1590/S0104-11692007000300023.

[20] M.N. Sarpiri and T.J. Gandomani, "A case study of using the hybrid model of scrum and six sigma in software development," *International Journal of Electrical and Computer Engineering*, vol. 11, no. 6, pp. 5342-5350, 2021, doi:10.11591/ijece.v11i6.pp5342-5350.

[21] R. Marques, G. Costa, M. da Silva, D. Gonçalves, and P. Gonçalves, "A gamification solution for improving Scrum adoption," *Empirical Software Engineering*, vol. 25, no. 4, pp. 2583-2629, Jul. 2020, doi:10.1007/s10664-020-09816-9.

[22] C. Verwijs and D. Russo, "A Theory of Scrum Team Effectiveness," *ACM Transactions on Software Engineering and Methodology*, vol. 32, no. 3, Art. no. 74, Apr. 2023, doi:10.1145/3571849.

[23] C. Matthies, "Agile process improvement in retrospectives," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, Montreal, Canada, May 2019, pp. 150-152, doi:10.1109/ICSE-Companion.2019.00063.

[24] A. Mesquida, J. Jovanovic, M. Jovanovic, and A. Mas, "Agile software process improvement: a collaborative game toolbox," *IET Software*, vol. 13, no. 2, pp. 106-111, Apr. 2019, doi:10.1049/iet-sen.2018.5096.

[25] S. Galván-Cruz, M. Mora, and R. O'Connor, "A Means-Ends Design of SCRUM+: an agile-disciplined balanced SCRUM enhanced with the ISO/IEC 29110 Standard," in *Trends and Applications in Software Engineering*, Advances in Intelligent Systems and Computing, J. Mejia, M. Muñoz, Á. Rocha, Y. Quiñonez, and J. Calvo-Manzano, Eds. vol. 688. Cham, Switzerland: Springer, 2018, pp. 13-23, doi:10.1007/978-3-319-69341-5_2.

[26] V. Henriquez, J. Calvo-Manzano, A. Moreno, and T. San Feliu, "Agile-CMMI V2.0 alignment: Bringing to light the agile artifacts pointed out by CMMI," *Computer Standards & Interfaces*, vol. 82, Art. no. 103610, Aug. 2022, doi:10.1016/j.csi.2021.103610.

[27] M. Noureldin, A. Ghalwash, L. Abd-Ellatif, and M. ElGazzar, "Blending agile methodologies to support automotive SPICE compliance," *Journal of Software: Evolution and Process*, vol. 34, no. 1, Jan. 2022, doi:10.1002/smr.2391.

[28] B. Kose, "Business process management approach for improving agile software process and agile maturity," *Journal of Software: Evolution and Process*, vol. 33, no. 4, Apr. 2021, doi:10.1002/smr.2331.

[29] C. Chen and J. Lee, "Comparative effects of knowledge-based antecedents in different realms of CMMI-based software process improvement success," *Computer Standards & Interfaces*, vol. 81, Art. no. 103599, Apr. 2022, doi:10.1016/j.csi.2021.103599.

[30] S.K. Amer, N. Badr, and A. Hamad, "Combining CMMI Specific Practices with Scrum Model to Address Shortcomings in Process Maturity," in The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2019). Advances in Intelligent Systems and Computing, A. E. Hassanien, A. T. Azar, T. Gaber, R. Bhatnagar, and M. F. Tolba, Eds. vol. 921. Cham, Switzerland: Springer, 2020, pp. 898-907, doi: 10.1007/978-3-030-14118-9_88.

[31] N.F. Arifin, B. Purwandari, and F. Setiadi, "Evaluation and recommendation for Scrum implementation improvement with hybrid Scrum maturity model: A case study of a new Telco product," in *2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*, Jakarta, Indonesia, Nov. 2020, pp. 178-183, doi:10.1109/ICIMCIS51567.2020.9354311.

[32] C. Matthies, "Feedback in Scrum: Data-informed retrospectives," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, Montreal, Canada, May 2019, pp. 198-201, doi:10.1109/ICSE-Companion.2019.00081.

[33] J. Pernstål, R. Feldt, T. Gorschek, and D. Florén, "FLEX-RCA: a lean-based method for root cause analysis in software process improvement," *Software Quality Journal*, vol. 27, no. 1, pp. 389-428, Mar. 2019, doi:10.1007/s11219-018-9408-8.

[34] A. Khan, M. Shameem, R. Kumar, S. Hussain, and X. Yan, "Fuzzy AHP based prioritization and taxonomy of software process improvement success factors in global software development," *Applied Soft Computing*, vol. 83, Art. no. 105648 Oct. 2019, doi:10.1016/j.asoc.2019.105648.

[35] S. Kupper, D. Pfahl, K. Jürisoo, P. Diebold, J. Münch, and M. Kuhrmann, "How has SPI changed in times of agile development? Results from a multi-method study," *Journal of Software: Evolution and Process*, vol. 31, no. 11, Nov. 2019, doi:10.1002/smr.2182.

[36] M. Muñoz, J. Mejia, and C. Laporte, "Implementing ISO/IEC 29110 to reinforce four very small entities of Mexico under an agile approach," *IET Software*, vol. 14, no. 2, pp. 75-81, Apr. 2020, doi:10.1049/iet-sen.2019.0040.

[37] A. Poth, S. Sasabe, A. Mas, and A. Mesquida, "Lean and agile software process improvement in traditional and agile environments," *Journal of Software: Evolution and Process*, vol. 31, no. 1, Jan. 2019, doi:10.1002/smr.1986.

[38] M. Choras *et al.*, "Measuring and Improving Agile Processes in a Small-Size Software Development Company," *IEEE Access*, vol. 8, pp. 78452-78466, 2020, doi:10.1109/ACCESS.2020.2990117.

[39] V. Henriquez, A.M. Moreno, and S. Gutiérrez, "Organizational training in agile settings under the lens of CMMI V2.0," *Journal of Software: Evolution and Process*, vol. 35, no. 9, 2023, doi:10.1002/smr.2502.

[40] A. Astridita, T. Raharjo, and A.N. Fitriani, "Perceived Benefits and Challenges of Implementing CMMI on Agile Project Management: A Systematic Literature Review," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 1, pp. 232-239, 2024, doi:10.14569/IJACSA.2024.0150122.

[41] J.A.V.M.K. Jayakody and W.M.J.I. Wijayanayake, "Process improvement framework for DevOps adoption in software development," in *2023 International Research Conference on Smart Computing and Systems Engineering (SCSE)*, Kelaniya, Sri Lanka, Jun. 2023, pp. 1-7, doi:10.1109/SCSE59836.2023.10214992.

[42] S. Galvan-Cruz, M. Mora, C. Laporte, and H. Duran-Limon, "Reconciliation of scrum and the project management process of the ISO/IEC 29110 standard-Entry profile-an experimental evaluation through usability measures," *Software Quality Journal*, vol. 29, no. 2, pp. 239-273, Jun. 2021, doi:10.1007/s11219-021-09552-3.

[43] K. Bhavsar, V. Shah, and S. Gopalan, "Scrum Challenges: An Agile Process Reengineering in Software Engineering," *International Journal of Innovative Technology and Exploring Engineerin*g, vol. 9, no. 4, pp. 1233-1241, Feb. 2020, doi:10.35940/ijitee.D1567.029420.

[44] M. Kuhrmann and J. Münch, "SPI is Dead, isn't it? Clear the Stage for Continuous Learning!," in *2019 IEEE/ACM International Conference on Software and System Processes (ICSSP)*, Montreal, QC, Canada, May 2019, pp. 9-13, doi:10.1109/ICSSP.2019.00012.

[45] A. Mas, A. Mesquida, and M. Pacheco, "Supporting the deployment of ISO-based project management processes with agile metrics," *Computer Standards & Interfaces*, vol. 70, Art. no. 103405, Jun. 2020, doi:10.1016/j.csi.2019.103405.

[46] K. Suteeca and S. Ramingwong, "The visualization of ISO/IEC29110 on SCRUM under EPF Composer," *Information*, vol. 12, no. 5, Art. no. 190, May 2021, doi:10.3390/info12050190.