Array 28 (2025) 100595

Contents lists available at ScienceDirect

Array

journal homepage: www.sciencedirect.com/journal/array

Agile change approach for collaborative software development contexts: A systematic literature review

José Luis González-Blázquez [*,1], Alicia García-Holgado [1], Francisco José García-Peñalvo [1]

Computer Science Department, University of Salamanca, Salamanca, Spain

ARTICLE INFO

Keywords:
Free software
Open-source
Non-proprietary software
Agile culture
Agile methodologies
Agile methods
Agile frameworks
Agile scaling
Agile software development

ABSTRACT

This systematic literature review examines how agile solutions can drive organizational change in collaborative open-source software (OSS) contexts. Motivated by persistent challenges in governance, alignment, contribution lifecycles, workflow, leadership, and measurement, the review asks which prescriptive and non-prescriptive agile approaches are being applied when organizations collaborate with OSS communities, and how these approaches mitigate those issues. The study first conducts an umbrella review (2000–2024) to confirm the gap and scope, then performs a main systematic review across digital libraries using inclusion, exclusion, and quality criteria. The synthesis maps findings to a conceptual framework of nine problem areas and two change paths. Results show a dominance of prescriptive methods, especially Scrum, LeSS, SAFe, and Kanban, for workflow transparency, dependency management, and coordination, while governance and leadership models remain underexplored. Building on this evidence, the paper proposes: (1) a prescriptive change approach for low-maturity organizations that integrates holacratic governance with Scrum/LeSS, Communities of Practice, Design Thinking for innovation, Management 3.0 leadership, and KPI-oriented cultures; and (2) a non-prescriptive approach for mature organizations based on unFIX's fractal organizational design, forums and collaboration patterns, delegation levels, and outcome-focused metrics to extend co-evolution with communities. The dual pathway enables organizations to select and sequence interventions that align with their paradigm and maturity, thereby bridging organizational and community boundaries to foster sustained agility. The review highlights open research needs on governance mechanisms, leadership in symbiotic ecosystems, and empirical evaluations of combined scaling approaches beyond SAFe, as well as longitudinal studies on alignment, dependency management, and measurement cultures in high-variability OSS environments.

1. Introduction

The pace of technological change and the need for innovation in an increasingly globalized society have accelerated dramatically in the current century, impacting all sectors and organizations without distinction. This new disruptive paradigm has challenged traditional management mechanisms, forcing organizations to reconfigure their business models and ways of working to generate new forms of value. In response, there has been a notable increase in the adoption of open-source projects and agile management methodologies within organizational portfolios. This trend reflects the increasingly prominent role these approaches play in business strategy, driven by their distinctive values, principles, and characteristics, as well as their ability to support digital transformation, enable data-driven analytics, and improve software application deployment [1].

This scenario has opened up new markets and fostered the emergence of new Open Source/Free Software (FOSS) initiatives [2], based on the collective intelligence of open-source software (OSS) communities. Within these communities, the delocalized participation of a multitude of contributors enables sustainability [3] and fosters innovation through their reviews, ideas, or contributions. These communities are spaces of open innovation and knowledge generation where an immense amount of collective intelligence is produced, which, when made available to organizational projects, allows us to understand not only their success but also the origin of the complexity of the numerous difficulties that arise in their interactions when they collaborate in the development of open-source projects [4].

Harnessing the distinctive collective potential of these communities

* Corresponding author.
E-mail addresses: idu018772@usal.es (J.L. González-Blázquez), aliciagh@usal.es (A. García-Holgado), fgarcia@usal.es (F.J. García-Peñalvo).
[1] https://ror.org/02f40zc51.

https://doi.org/10.1016/j.array.2025.100595
Received 30 March 2025; Received in revised form 14 October 2025; Accepted 19 November 2025
Available online 21 November 2025

and aligning it with flexible value solutions represents a challenge; more and more organizations are choosing to incorporate it into their business models. This is reflected in the incorporation of numerous projects into organizational portfolios, which involve internal open-source development and agile management practices. However, these initiatives are not always accompanied by the appropriate decisions necessary for their successful structural and operational implementation.

The emergence of these new business niches has led to changes in the way organizations develop and manage their software projects [5]. In particular, there has been a growing incorporation of new projects created in open-source environments, supported by collaborative contexts, and a progressive move away from traditional forms of management, in favor of evolutionary agile management practices [6]. As a result, the market has witnessed a proliferation of agile solutions, including Scrum, Kanban, Scrumban, LeSS, SAFe, Nexus, and Scrum@Scale. These frameworks provide prescriptive implementation roadmaps for organizations seeking to leverage the benefits of adopting organizational agility. However, selecting and adapting any of these approaches within a specific organizational context is far from a straightforward decision.

Adopting and scaling agility across both operational and organizational dimensions, regardless of the framework chosen, is a long and arduous journey fraught with difficulties and complex decisions that require a comprehensive adaptation of both culture and organizational structure to align with this new paradigm. Many companies become discouraged when they fail to achieve the expected progress and results, often leading them to abandon their pursuit of organizational agility. Valderrama [7] points out that the success of a successful transformation must be based on four pillars: the customer, collaboration, employees, and agility itself. In addition, decision-making is focused on customers and their needs, allowing the organization to operate agilely through the collaboration of empowered people, thereby providing value to customers.

With this idea in mind, research was initiated focused on the collaborative context of organizations whose business and innovation models revolve around open-source development in partnership with OSS communities. The objective was to guide these organizations in their decision-making processes when facing the challenges and complex dynamics that arise from collaboration.

This was addressed by developing a quantitative change framework based on various agile solutions that, following the model prescribed by Kotter [8], allows organizations and communities to effectively address common problems within this collaborative context (Table 1), to position them within a Teal paradigm (Fig. 1), as described by Laloux [9], from which they can achieve organizational agility.

The path to achieving organizational agility, both at the operational and managerial levels, begins with a preliminary assessment of the initial level of agile maturity [40], using practices such as Scrum Level or SMM. This process continues with the approach to change based on agile solutions. To develop it, a conceptual framework is proposed (Fig. 2), supported by two alternative change proposals that combine various prescriptive and non-prescriptive agile solutions. These proposals aim to guide decision-making processes in addressing the problems and challenges mentioned above (Table 1) that contribute to achieving organizational agility.

The choice between the two change pathways will depend on the organizational paradigm (Fig. 1) and the maturity level assessed through Scrum Level or SMM evaluations.

a) Prescriptive agile change proposal

The set of change objectives proposed (SA_P1–SA_P9) within this prescriptive framework (Table 2) constitutes an integrated model for advancing organizational agility in collaborative contexts, particularly within open-source software ecosystems. This framework is designed for organizations with a low level of agile maturity, whose

Table 1
Scope and characterization of common challenges in collaborative contexts.

ID	Problem area	Description of the problem
P1	Governance	The absence of transparent [10] and accountable [11] and excessively centralized governance models hinders the fluidity of interactions between the organization and the community. This is due to the lack of an organizational design [12] suited to this collaborative context, which represents one of the major challenges [13] for the implementation of agility.
P2	Business Model	Rigid and inefficient business models [14] for these collaborative contexts result in a loss of value in organizational portfolios, as they are unable to harness the huge intellectual and innovative capital of OSS communities. This limitation can threaten the sustainability and profitability of these organizations.
P3	Operational Model	The presence of inefficient work models and patterns [15], misaligned with the community, discourages the participation of new contributors and leads to a decrease in their contributions. This negatively impacts the quality of the software developed, poor management of contributions, poorly sized work teams, non-optimized workflows and delays in the time to market of the developed product.
P4	Innovation	The loss of creative potential and collective intelligence [16] generated by OSS communities highlights the urgent need to find solutions to capitalize on them for the benefit of organizations
P5	Contribution Lifecycle	Large-scale product development requires the involvement of numerous organizational teams and a large number of community collaborators. This scenario leads during the development process to inefficient contribution cycles [17], the appearance of numerous blockages and dependencies between teams and collaborators or the appearance of inefficiencies in the management of community contributions.
P6	Workflow	Collaboration between organizations and OSS communities leads to numerous workflow-related issues, including misaligned release cycles, frequent bottlenecks, and excessive increased time to market [18]. These challenges can result in a decrease in brand value, which weakens the organization's market positioning against the competition.
P7	Alignment and Synchronization	In these collaborative contexts, a recurring problem is the poor synchronization between teams and collaborators and the latter's alignment with organizational objectives. This translates into teams that generate numerous unresolved dependencies [19] and a lack of alignment of collaborators with the objectives and results expected by the organization, which causes the appearance of negative metrics in projects, non-compliance with quality levels or delays in the progress of projects.
P8	Leadership	The presence of meritocratic structures with leadership roles such as the Benevolent Dictator for Life (BDFL) [20] in communities, or highly hierarchical organizational designs characterized by numerous managerial roles, significantly reduces structural flexibility, hinders interactions between both parties, and prevents a rapid response to the changing needs of these collaborative contexts.
P9	Measurement Culture	The lack of a culture of measuring results and performance [21] in the communities has a negative impact on the organization, impacting not only the measurement of its results and the fulfilment of objectives, but also damages the quality agreed in the SLAs/SLAs signed with clients, causing compliance issues and severely damaging the sustainability and reputation of these projects.

Table 2
Prescriptive agile change proposal.

(SA_P1)	**Governance Model Based on a Holacratic Organizational Design with Agile Methods Scrum/LeSS** The Teal paradigm has as its outstanding exponent an organizational design based on holacracy or the fractal model. Holacracy [22] seeks greater dynamism through an adaptable circular structure, distributed authority where people have roles rather than titles, fostering innovation, and empowering workers through equity and collaboration. This governance model is based on a holacratic organizational design [22], which replaces traditional hierarchical structures with a system of distributed and decentralized authority that encourages self-organized roles in the form of teams and collaborators. From an operational point of view, the model combines the agile Scrum methodologies and its scaled method, LeSS (Large-Scale Scrum), integrating them within a holacratic structure that will facilitate the work autonomy of self-organized teams and encourages reciprocal prosocial behavior with the community.
(SA_P2)	**Hybrid Collaboration Model with the OSS Community Based on Community Management and Support Models, adopting a Symbiotic Approach Inspired by Lean Manufacturing** This model proposes the adoption of a hybrid collaborative model with the OSS community towards co-evolutionary development, based on the community management model and the community support model, where a symbiotic approach is adopted that takes advantage of the characteristics of both. This social innovation business model generates mutual value that can be optimized by embracing agility with scalable practices across the organization and community through Lean Manufacturing, allowing organizations to align and respect community norms through shared roles and synchronized workflows, providing collaborators with an appropriate technological infrastructure for their participation and contributions.
(SA_P3)	**Crowdsourcing-Based Work Model as a Decentralized Labor Movement and Collective Production Framework in Convergence with InnerSource Development Patterns to Enable Symbiotic Collaboration with OSS Communities** This working model takes advantage of crowdsourcing [23] as a decentralized work and for collective production model, allowing for large-scale distributed participation in software development. To improve the internal structure of collaboration with the community, the model integrates patterns from the InnerSource internal development approach along with agile management practices. InnerSource [24–26] encourages decentralization and reciprocal cooperation between organizational teams and collaborators. By aligning the agile management of Feature teams (Scrum/LeSS) operating with an internal Innersource community and with an external symbiotic OSS community, it is possible to develop a work model that optimizes the needs and organizational structure as a channel for the innovative potential of the community.
(SA_P4)	**Innovation Capitalization through Design Thinking within the Scrum/LeSS Cycle** This model proposes a structured approach to capitalize on open innovation in communities by integrating the five phases of Design Thinking [27] (empathize, define, ideate, prototype, and test) within the iterative cycle of Scrum and its scaled framework, LeSS (Scrum at Scale). This integration of Design Thinking with Scrum/LeSS [28] will empower innovation within the product development cycle, positively impacting the delivery of high-quality software. In addition, it will allow all the ideas and contributions of the community that are systematically incorporated into the development cycle to be capitalized in the different phases of Design Thinking, providing a shared understanding of the problems that will enrich the innovative process of the organization.
(SA_P5)	**Agile Scaling in Organizational Dimensions and Symbiotic Communities through a Holacratic Design with the LeSS Framework** Achieving organizational agility is a progressive process that depends on the maturity level of the organization and the community and becomes more complex and challenging in environments at scale. All of the structural changes proposed above in this agile transformation (SA_P1-to-SA_P3) approach led to the creation of a suitable organizational design and structure optimized to embrace agility at scale. This model promotes the application of the Less scale frame in a holacratic design, based on the criteria of choice described by Almeida and Espinheira [29], to create an agile structure at scale that enables the integration of external community contributions into internal value streams. It's not about delivering value faster, but rather about creating an agile structure with the ability to effectively adapt to change and deliver value on an ongoing basis.
(SA_P6)	**Workflow and Queue Management with LeSS and Kanban: Application of Queueing Theory and the STATIK Method** In large-scale agile environments, coordinating multiple teams around a shared product presents significant challenges in workflow management that avoids systemic inefficiencies and queue management. As the number of initiatives and tasks within a project increases, numerous queues arise. LeSS proposes as the best option a holacratic or fractal organizational design, based on Feature Teams framed in a Teal paradigm, which avoids the appearance of queues [30], It only recommends applying queue management (Queue Theory) when it cannot be completely eliminated, for which it prescribes as the best option the change of organizational design and efficient management of the workflow. Integrating Kanban practices within the LeSS framework provides an agile tool for visualizing, limiting, and optimizing workflow and queues. A frequent problem in projects where several agile teams are involved is to achieve a balance between the demand of the work that each team is self-assigned (regularly or irregularly) and the speed at which this team delivers (Throughput), so as not to affect its capacity and the quality of the code developed. Kanban proposed a dragging system to control that work entry through WIP (Work In Progress) limits and that no new element is incorporated until another one has been finished, and the team has the capacity to develop it. This will produce slack within teams, so that team members can have time to collaborate on the internal Innersource project. In this collaborative context, the STATIK (System Thinking Approach to Introducing Kanban) approach [31] offers a systematic method identifying 8 iterative steps to design Kanban systems based on systems thinking. When applied in a LeSS environment, STATIK enables organizations to design Kanban systems tailored to collaborative contexts involving multiple teams working from a single product backlog. This integrated approach not only improves predictability, but also promotes systemic learning, allowing organizations to dynamically and sustainably adapt their workflows to team participation and contributions from external collaborators.
(SA_P7)	**Alignment of Teams and Community Collaborators with LeSS through Communities of Practice (CoP) & Dependency Management with LeSS and Kanban** LeSS emphasizes the importance of coordinating and aligning multiple teams working on a single product. Communities of practice (CoPs) [32], in addition to their role in knowledge sharing and collaborative learning, are essential to align and ensure consistency between internal teams and external community contributors. These communities are a tool for integration and decentralized coordination [33] linked to the social theory of learning [34] that strengthen cohesion between teams and communities, ensuring that contributions and technical standards evolve consistently. Dependency management with LeSS and Kanban: In large-scale environments, teams inevitably face technical dependencies (shared components), organizational dependencies (shared specialist roles), and community dependencies (integration of internal teams with external collaborators). This model proposes synchronizing teams and collaborators to manage dependencies using LeSS [35] which provides internal and external structural alignment, and Kanban, which provides operational transparency to manage dependencies, queues, and workflow constraints.
(SA_P8)	**Leadership Management in LeSS with Agile Leadership and Management 3.0** In large-scale agile environments, leadership must foster self-management, alignment, and continuous improvement across multiple teams and contributors working on a single product. LeSS provides the structural framework, while Agile Leadership [36] and Management 3.0 [37] offer principles and practices to empower leaders at all levels of the organization and community. The introduction of agile roles within the community (Business Owner, Community Facilitator, Contributor Crews, etc.) improves transparency of contributions and two-way coordination with the organization.
(SA_P9)	**Fostering a Culture of Objectives, Outcomes, and Agile Metrics** In collaborative contexts characterized by high variability, where the overall goal is to achieve organizational agility, it is essential to promote a culture focused on aligned goals, measurable results, and actionable metrics [38]that drive continuous improvement. This model aims to improve development software processes throughout their lifecycle, based on key performance indicators (KPIs). The incorporation of agile practices and KPIs (burn-down and burn-up graphs, team velocity, etc.) so that they can be implemented and accepted by teams and collaborators allows the performance of the Scrum/LeSS [39] process to be evaluated and will allow organizations to effectively translate their strategic objectives into tangible and measurable results.

Fig. 1. Organizational paradigms described by Laloux.

	ID Problem area	Solution Scope: Prescriptive agile change proposal	Solution Scope: Non-prescriptive agile change proposal
Organizational change objectives	P1	(SA_P1): Governance Model Based on a Holacratic Organizational Design with Scrum/LeSS Agile Methods	(SA_NP1): Governance Model Based on a Fractal Organizational Design with the unFIX Agile Method
	P2	(SA_P2) & (SA_NP2): Hybrid Collaborative Model with the OSS Community Based on Community Management and Support Models, adopting a Symbiotic Approach Inspired by Lean Manufacturing	
	P3	(SA_P3): Work Model Based on Crowdsourcing as a Decentralized Labor Movement and Collective Production Framework in Convergence with InnerSource Development Patterns to Enable Symbiotic Collaboration with OSS Communities.	(SA_NP3): Collaborative Work Model Based on the Interaction of unFIX Crews with an InnerSource Community and a Symbiotic OSS Ecosystem
Operational change objectives	P4	(SA_P4): Innovation Capitalization through Design Thinking within the Scrum/LeSS Cycle	(SA_NP4): Innovation Capitalization through the unFIX Innovation Vortex Pattern
	P5	(SA_P5): Agile Scaling in Organizational Dimensions and Symbiotic Communities through a Holacratic Design with the LeSS Framework	(SA_NP5): Agile Scaling across Organizational Dimensions and Symbiotic Communities through a Fractal Design with unFIX
	P6	(SA_P6): Workflow and Queue Management with LeSS and Kanban: Applying Queueing Theory and the STATIK Method	(SA_NP6): Workflow, Queue, and Delay Management with unFIX Patterns and Kanban
	P7	(SA_P7): Alignment of Teams and Community Collaborators with LeSS through Communities of Practice (CoP) & Dependency Management with LeSS and Kanban	(SA_NP7): Aligning Crews and Community Collaborators with unFIX Forums and unFIX Initiatives and Managing Dependencies with unFIX Dependency-Breaker Patterns
	P8	(SA_P8): Leadership Management in LeSS with Agile Leadership and Management 3.0	(SA_NP8): Strategic Leadership through Delegation Levels: unFIX and Management 3.0 Insights.
	P9	(SA_P9): Fostering a Culture of Agile Objectives, Results, and Metrics.	(SA_NP9): A Culture of Agile Objectives, Outcomes, and Metrics in unFIX.

Fig. 2. Key factors for achieving Organizational Agility through prescriptive and non-prescriptive agile solutions in this collaborative context.

paradigms—according to Laloux's organizational typology—correspond to Red, Amber, or Orange stages.

Its purpose is to go beyond traditional prescriptive approaches, addressing areas as diverse as they are complementary within the organization, such as governance, business and work models, collaboration at team and community scale, workflow optimization, community alignment with organizational goals, innovation, agile leadership, and a culture of measurement. Together, these areas provide a holistic perspective of organization and community, as an integrated and interconnected whole, where agile solutions can be applied in organizational environments of collaborative software development, with a symbiotic community culture.

This integrated approach highlights both the opportunities and challenges of embedding agile practices in open and collaborative ecosystems, while also indicating avenues for future empirical validation and refinement.

b) Non-prescriptive agile change proposal

The set of change objectives proposed (SA_NP1–SA_NP9) presented within this non-prescriptive framework (Table 3) constitutes, similar to the previous integrated model, a structured opportunity to advance organizational agility in collaborative contexts. In contrast to the prescriptive framework, this model is designed for organizations with a higher level of agile maturity, whose paradigm, according to Laloux's organizational typology, corresponds to Green organizations. The framework offers a holistic and scalable approach to applying agile practices within symbiotic organizational–community ecosystems, thereby fostering sustained adaptability and continuous co-evolution.

The identification of the two agile change proposals (prescriptive and non-prescriptive), together with the use of practices, methods, and techniques that enable the challenges and difficulties of this collaborative context to be addressed from an agile perspective (Fig. 3), will represent a valuable and relevant contribution to the software engineering community. This approach offers a novel perspective for further research on how agility can serve as a means to overcome problems and challenges in collaborative and highly variable environments, which are characteristic of open-source software development.

The objective of this study is to identify and examine existing systematic reviews in the literature on the application of agile solutions within organizations operating in collaborative open-source development contexts, and to determine whether proposals exist that seek to leverage agility as a means to address the difficulties, challenges, and issues faced by OSS organizations and communities when collaborating in open-source projects.

To achieve this objective, the present study reviews the literature published between 2000 and 2024, following the guidelines of Kitchenham and Charters [56] and drawing on the work of authors such as García-Holgado et al. [57] and García-Peñalvo [58]. Her study outlines the different phases that a systematic review should include, emphasizing the introduction of methodological frameworks that serve as references, as well as detailing the planning, execution, and reporting stages. This structured approach supports the identification of existing proposals that either confirm or refute the development of the proposed change approach based on agile solutions.

Currently, the contexts of agility and free and open-source software (FOS) [59], which frame this research, are experiencing remarkable growth. This expansion is primarily driven by the increasing volume of research on agile methods and methodologies, the emergence of new non-prescriptive approaches such as unFIX [60], and the rise of numerous OSS projects. Notable examples include socially oriented projects such as Freedone [61], Open source security initiatives like Google OSS-Fuzz [62], and the OSS Rebuild initiative [63], which highlight the importance of continuous code release and its tangible impact on users [64], as well as the introduction of new functionalities developed in Open source environments [65].

Table 3
Non-prescriptive agile change proposal.

(SA_NP1)	**Governance Model Based on a Fractal Organizational Design Using the unFIX Agile Framework** This governance model is based on a fractal organizational design [41], where the structure is composed of autonomous fractal teams at the operational level and self-similar at scale, which replicate governance patterns at multiple levels of the organization. To implement this design within a collaborative context, the model leverages the fractal approach of the unFIX agile framework [42], along with its library of structural patterns and their self-organizing multifunctional crews. The combined integration of fractal design and unFIX shows a flexible model different from the holacratic disruptive model, from which a mature organization can adapt and integrate the community with which it collaborates into its organizational design.
(SA_NP2)	**Hybrid Collaborative Model with the OSS Community Based on Community Management and Support Models, adopting a Symbiotic Approach Inspired by Lean Manufacturing** The same solution (SA_P2) is proposed for both proposals.
(SA_NP3)	**Collaborative Work Model Based on the Interaction Between unFIX Crews, an InnerSource Community, and a Symbiotic OSS Ecosystem** This collaborative work model is supported by the library of teamwork patterns of the unFIX framework and is based on the interaction of the different self-organized and autonomous crews, operating the value streams through the organizational structure with an internal Innersource community and a symbiotic external OSS community. It encourages co-evolution and cooperation between internal crews and the external symbiotic community, allowing constant interaction between the different actors involved in the development of the project.
(SA_NP4)	**Innovation Capitalization through the unFIX Innovation Vortex Pattern** The capitalization of open innovation and collective intelligence of communities requires, in order to transform it into results, that organizations enable internal structures and processes that serve as reciprocal co-creation channels. In this model within the unFIX pattern library there exists for this purpose, the innovation vortex pattern [43] composed of 7 flows (Contextualize, Empathize, Synthesize, Hypothesize, Outsource, Validate and Systematize) that represents a fusion of the Design Thinking and Lean Startup approaches and constitutes a crucial element within fractal organizational design. versatile and adaptable promoted by the unFIX framework.
(SA_NP5)	**Agile Scaling Across Organizational Dimensions and Symbiotic Communities through a Fractal Design with unFIX** Although there is still no empirical evidence of any organizational scaling with unFix [44] due to the youth of this agile method, this model proposes to replicate at scale the adoption of the Unfix model from a fractal design in its structural and community layers, relying on its library of patterns (including self-similar scaling patterns) [45] that guarantee the cohesion and alignment of the entire network of internal and external crews. through a self-similar and co-evolutionary design adapted to this collaborative context.
(SA_NP6)	**Workflow, Queue, and Delay Management with unFIX and Kanban Patterns** Workflow efficiency is one of the most critical factors in effectively delivering value within agile organizations. Delays and queues often emerge as hidden sources of waste, negatively impacting predictability and time to market. While Kanban provides well-established practices for visualizing and managing workflow inefficiencies, integrating unFIX patterns allows organizations to structure crews and collaborators in ways that minimize systemic queues and delays and maximize the value stream By integrating these patterns fractally (Value stream patterns [46], Role Attributes [47] or Queues and backlogs [48]) into crews and the community, unFIX ensures that the organizational structure itself reinforces the operational structure. This alignment optimizes workflow efficiency and mitigates systemic bottlenecks, enabling a sustainable balance between performance and continuous value delivery.
(SA_NP7)	**Aligning Teams and Community Collaborators through unFIX Forums and Initiatives, and Managing Dependencies with unFIX Dependency-Breaking Patterns** Scaling collaboration beyond crews requires coordination mechanisms that preserve their autonomy while allowing for effective integration with the wider community. The unFIX method employs two complementary mechanisms to address alignment and synchronization: forums and initiatives (to align stakeholders) and a set of dependency breakers patterns [49] (to minimize, absorb, or eliminate dependencies that would otherwise create delays, software quality fragility, and extended time to market). The combination of unFIX's forums [50](continuous spaces for coordination and collective learning) and unFIX's Collaboration Moments patterns [51] establish a practical alignment mechanism between crews and community contributors. This integration allows collaboration to seamlessly scale across organizational boundaries and OSS ecosystems, ensuring consistency in joint value creation.
(SA_NP8)	**Strategic Leadership through Delegation Levels: unFIX and Management 3.0 Perspectives** The unFIX method with its delegation patterns [52], inspired by Management 3.0, provides a comprehensive framework for strategic leadership and delegation of responsibilities. By leveraging the unFIX pattern library, leaders can design fractal organizational structures that support decentralized decision-making. Applying the seven levels of delegation allows leaders to ensure that authority is appropriately distributed, fostering an environment in which crews and collaborators are empowered to make decisions aligned with the organization's strategic objectives. This integrated approach enables organizations and communities to extend distributed decision-making across the structure, facilitates dynamic governance and adaptive coordination with the community.
(SA_NP9)	**A Culture of Objectives, Outcomes, and Agile Metrics in unFIX** This model uses the Bets and Objectives [53], Goal-Setting Aspects [54] and Measures and Metrics [55] patterns of the unFIX method, to design a culture of agile goals, results and metrics. The latter introduces five key indicators (input, impact, outcome, output, and process metrics) that collectively provide a comprehensive view of organizational performance, as well as serving as a valuable tool for measuring progress toward strategic goals. unFIX promotes that strategic objectives are systematically translated into measurable results. This commitment to continuous measurement and outcome-based improvement reinforces the alignment of organizational objectives with the results of community input.

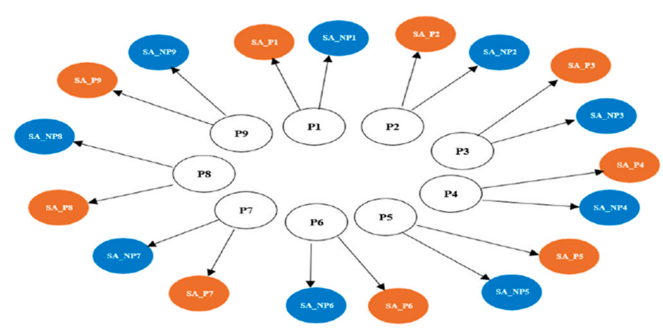

Fig. 3. Radial hierarchical diagram Pi => SA_Pi/SA_NPi

Furthermore, agile management is playing an increasingly relevant role across all areas of software development. This can be observed, for example, in the integration of agility with usability metrics for mobile applications developed in community-driven environments [66] or in the contribution of agile culture to organizational productivity and sustainable growth, as illustrated by the case of KFC [67], where agility was fundamental in supporting critical decision-making to adapt to market demands.

The structure of this study is as follows. Section 2 presents the research methodology, knowledge sources, search protocol, and evaluation criteria. Section 3 reports the results and analyzes those that most accurately address our mapping and research questions, highlighting their significance and relevance to the study. Section 4 revisits the research problem and relates the obtained results to demonstrate how they contribute to answering the research question. Section 5 discusses threats to validity and potential biases that may affect the results. Finally, Section 6 presents the conclusions, outlining the rationale for the study, emphasizing the main findings, and suggesting possible directions for future research.

2. Methodology of the review

This section describes the research methodology employed in this study to conduct the systematic literature review. Through a rigorous and detailed analysis, the aim is to ensure a thorough and replicable review.

2.1. Umbrella review

Given the problem outlined in the introduction, before conducting the main review, a preliminary literature review, an umbrella review [68], was carried out to verify that no reviews exist within the defined period of 2000–2024 that address the objective of this study. The purpose of the umbrella review is to identify previous bibliographic reviews on the subject to confirm whether the proposed review in Section 2.2 is justified. This preliminary review examines studies published between 2000 and 2024 that focus on reviews of agile adoption in open-source organizations and OSS communities.

The search protocol used in this first review is as follows: *"Systematic literature review" OR "SLR" OR "literature review" OR "systematic literature mapping") AND ("Open approach scaled agile" OR "Open source organizations" OR "Free Software organizations" OR "Free Software Communities"*

A total of 67 studies (Table 4) were identified, primarily journal articles, published between 2009 and 2024, from the sources Web of Science (WoS), Scopus, ACM, and IEEE Xplore. A selection was then carried out to include only those studies considered relevant to Research Question 1 (IPQ1): *Are there systematic reviews or literature mappings within the defined period that address agile change approaches in Free or Open source Software organizations and OSS communities?*

A total of 67 studies were initially identified, primarily journal articles, published between 2009 and 2024, from the sources WoS, Scopus, ACM, and IEEE Xplore. From the initial set, 15 studies were considered relevant (Primary Selection) and selected for Research Question 1 (IPQ1). After removing duplicate studies and focusing on our specific context, 11 studies (Table 5) were finally selected for analysis (Secondary Selection). Their contributions and limitations were examined in relation to the study's objective, and key terms and keywords were analyzed to identify relevant findings within these studies.

After analysis, no reviews were found addressing agile scaling approaches in OSS organizations and communities. However, there is scientific literature focused on agility [69–75,79], which addresses aspects related to agile scaling using prescriptive methods, the identification of motivators present in scaling, or the discussion of problems and challenges that arise when organizations scale agility. Additionally, studies have been found that examine the adoption and impact of open-source software in organizations [76–78]. Based on this umbrella review, the authors conclude that, although there is growing interest and several studies on the applicability of agility in organizational contexts through agile methods, a lack of understanding remains regarding the adoption of these methods in open-source organizations and OSS communities.

The results of this umbrella review contribute to a better understanding of the applicability of agility within organizations, as well as the prevalence of prescriptive methods and the absence of non-prescriptive approaches. Existing limitations include the scarcity of studies addressing scaling through a combination of different agile methods in specific, highly variable contexts such as open-source development, as well as limited research on the applicability of agility in collaborative environments based on the collective intelligence of OSS communities.

The literature is heavily focused on the application of prescriptive methods for achieving organizational agility, overlooking the potential

Table 4
Umbrella review results.

Source	Total Results Found	Primary Selection	Secondary Selection	Selected Studies for Final Analysis
SCOPUS	10	4	3	[69–71]
ACM	1	1	1	[72]
WOS	38	8	6	[73–78]
IEEE Xplore	18	2	1	[79]

Table 5
Secondary selection of studies from the umbrella review.

	Year	Contributions	Limitations
[69]	2019	A review focused on identifying, at scale, how agile scaling frameworks such as SAFe, LeSS, and Scrum of Scrums address inter-team dependencies.	There is a scarcity of studies in the literature that provide a detailed understanding of how large-scale agile frameworks operate within organizations.
[70]	2018	Review focused on SAFe and LeSS scaling practices, challenges encountered, and critical success factors for agile scaling in organizations.	Limited scope of the review, aimed at a business audience, whose collaboration is often constrained by strict deadlines.
[71]	2013	Review focused on evaluating the relationship and integration between Agile Software Development (ASD) methods and open-source Software Development (OSSD) methods.	Although case studies exist in the literature using ASD and OSSD jointly, there is insufficient scientific evidence on their full integration.
[72]	2018	Review focused on identifying factors in the adoption of the prescriptive SAFe scaling framework within organizations.	Inclusion of grey literature not peer-reviewed, due to the current scarcity of research on scaling frameworks.
[73]	2015	Review focused on studies analyzing the problems and limitations of scaling agile methods in large-scale projects.	Scarcity of research on agile scaling that addresses challenges in large-scale projects, with limitations related to documentation, iteration time constraints, and coordination/dependencies between distributed teams.
[74]	2016	Review focused on studies on the adoption of agile methods and large-scale Lean software development, emphasizing challenges and success factors in industry.	Limitations in the classification and importance ranking of scaling challenges and success factors, and the percentage of studies situated in industrial contexts.
[75]	2019	Review focused on identifying organizational motivators to be addressed by management for the implementation of agility.	Theoretical work, limited to the academic research context, where most participating researchers lack practical experience in large-scale agile software development.
[76]	2010	Review focused on analyzing the context of open-source software development and its adoption in different ways within organizations.	Limitations associated with the scarcity, relevance, and quality of empirical publications on open-source software in organizations, as well as the lack of sufficient support from related literature.
[77]	2020	Review focused on identifying which economic and technical factors are critical for organizations to adopt open source and free software.	Limitations in the literature on evidence-based software studies. The context of organizations adopting OSS is not adequately described.
[78]	2017	Review focused on identifying the current state of understanding of OSSECO ecosystems by organizations and free Software communities.	Limited to the assessment of the current state of research on OSSECO ecosystems and their specific characteristics.
[79]	2022	Review focused on comparing the main prescriptive agile methods and two customized scaling approaches at Nokia and Ericsson.	Limited current empirical evidence in the reviewed studies regarding comparisons between different large-scale agile methods.

of change approaches that combine different prescriptive solutions or integrate these methods with non-prescriptive alternatives. Nevertheless, these existing reviews provide the foundation upon which our study builds, offering a general perspective on the applicability of prescriptive agile methods in collaborative contexts, while also highlighting potential directions for future research.

2.2. Main review

The process of the main review consists of defining the research and mapping questions, establishing search terms, setting inclusion and exclusion criteria, and conducting quality assessment.

2.2.1. Research questions

The PICOC method was employed to identify the primary concepts and ideas of the review, laying the groundwork for formulating the inclusion and exclusion criteria, and to design the search strategy (Table 6).

The main research question (RQ1) (Table 7) and eight mapping questions [80] (MQ1–MQ7) (Table 8) were formulated to guide the identification of search terms. The purpose of RQ1 is to determine which agile solutions (frameworks, approaches, methods, or methodologies), both prescriptive and non-prescriptive, are being applied in organizations operating within collaborative open-source development contexts, and how these solutions address, through the adoption of agility, the challenges, difficulties, and issues faced when collaborating with OSS communities.

To define the scope of the RQ1 research question more precisely, five Sub-Questions (SPs) (Table 7) were established to categorize the selected studies into different research categories (CAT1-CAT5) (Table 21).

These categories contain all the aspects addressed in the research: (CAT1:SP1) open-source organizations and free communities, (CAT2: SP2) the agile context and the open-source context, (CAT3:SP3) organizational agile scaling methods, (CAT4:SP4) challenges of the agile change process, and (CAT5:SP5) the degree of agile maturity.

In addition, eight mapping questions were defined (Table 8) with the aim of presenting, explaining and contextualizing the dataset that represents the corpus of the systematic review using mapping techniques [81]. The mapping questions (MQ1-MQ8) are.

This structured approach ensures that the review systematically captures both prescriptive and non-prescriptive agile solutions, their application, and their effectiveness in OSS collaborative environments. In this study, the systematic review was conducted following the PRISMA protocol (Fig. 4), aiming to optimize the transparency and quality of the research process through a detailed and reproducible description of all stages of the review.

2.2.2. Search strategy

Once the research and mapping questions were defined, a search strategy was developed to identify all studies within the scope of this investigation. This involved determining the sources, keywords, search terms, and the period covered by the review. Finally, the repository where the complete dataset of both the umbrella and main review will be stored is indicated.

Table 6
The PICOC method.

Population (P)	OSS organizations and communities.
Intervention (I)	Specific agile proposals or solutions implemented through change-oriented approaches.
Comparison (C)	Implicit. The intervention (agile solutions) is compared to the current state or to traditional approaches that fail to adequately address the challenges in OSS contexts.
Outcomes (O)	Identification of practices and approaches that enable organizations and OSS communities to address the complex challenges and dynamics of collaboration in open-source projects, with the expected result of improved coordination, collaboration, and effectiveness.
Context (C)	Collaborative environments inherent to open-source software projects.

Table 7
Description of the research question and sub-questions.

	Research question	Sub-Questions (SPs) derived from RQ1
RQ1	What proposals exist that propose specific agile solutions to the complex difficulties and dynamics that OSS organizations and communities face when collaborating together on open-source projects?	**SP1.** How are open-source organizations and communities structured, and how do they interact in collaborative development projects? **SP2.**How do agile practices integrate with the context of open-source development in collaborative OSS environments? **SP3.** What agile scaling methods are applied in OSS contexts and how do they adapt to the culture and structure of the organization? **SP4.** What are the main challenges and difficulties encountered during the adoption of agile practices in OSS organizations and communities? **SP5.** What is the level of maturity in terms of agile adoption of OSS organizations and communities and how is this maturity assessed or measured?

Table 8
Description of the mapping questions.

	Mapping questions
MQ1	What studies have been published each year?
MQ2	What types of studies and proposals have been published?
MQ3	What is the context of the proposed solutions?
MQ4	In which sources are the analyzed studies located?
MQ5	Which domains do these studies focus on?
MQ6	What types of organizations are involved?
MQ7	Which agile methodological solutions are addressed in the studies?
MQ8	Which research categories are covered by the selected studies?

2.2.2.1. Sources. A fundamental part of any systematic review is the selection of appropriate sources for the literature search [82]. In an environment where information is abundant and often of varying quality, we consciously choose rigorous, expertly validated sources. Our selection was based on key criteria: accessibility for research, the ability to use Boolean operators and logical expressions, the availability of advanced search functions, and the ability to perform both specific and exhaustive searches.

The final list of sources included highly reputable databases such as SCOPUS and WoS, along with digital libraries specializing in software engineering, such as ACM Digital Library and IEEE Xplore. These platforms are crucial for comprehensive searches, as they enable the use of complex and accurate search strings in titles, abstracts, and metadata, ensuring thorough coverage of the academic literature.

Additionally, Google Scholar was included not as a primary source for replicating the search, but as a strategic complement. Unlike structured databases, Google Scholar employs a distinct search algorithm that indexes a broader range of documents. For this reason, a more specific search string, limited to article titles, was used to maximize accuracy and avoid loading irrelevant results. This complementary approach enabled us to identify high-impact studies that the more technical searches in conventional digital libraries might have missed. In this way, our combined strategy ensured both completeness and accuracy in identifying relevant literature, as recommended by experts in the field.

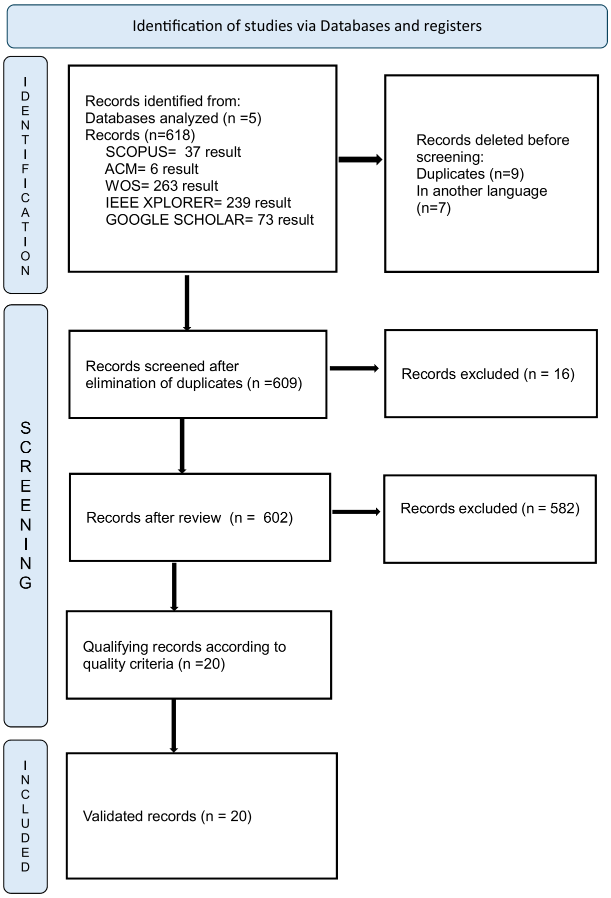

Fig. 4. PRISMA flow.

2.2.2.2. Keywords and search terms. The definition of keywords and search terms was limited to the contexts of agility and open source to encompass the maximum possible body of knowledge. For the formulation of the search terms, the following English keywords were proposed: free software, open source, non-proprietary software, agile culture, agile methodologies, agile methods, agile frameworks, agile scaling, and agile software development.

After selecting the keywords related to the objective of this review, the Boolean operators "AND", "OR", and "NOT" were used as logical connectors to combine these terms, as all databases and most search engines support these operators to interpret search queries. Based on the defined terms, the following general search string was formulated, which will be adapted for each of the selected databases for the review:

("Agile methods" OR "Agile method" OR "Agile frameworks" OR "Agile framework" OR "Agile methodologies" OR "Agile methodology" OR "Scaled agile framework" OR "Scaled agile methods" OR "Agile software development" OR "Agile culture") AND ("Open source" OR "Open source" OR "Free software" OR "Non-proprietary software" OR "Open source organizations" OR "Free software organizations" OR "Free software communities").

The search string was adapted to the specific requirements of each database. The search was conducted similarly across all sources, with particularities that can be consulted in the public dataset of this review.

2.2.3. Study selection and quality assessment criteria

The selection of studies was conducted using the sources specified in Section 2.2.2.1. Seven inclusion criteria (IC) and seven exclusion criteria (EC) were defined for the study selection process.

2.2.3.1. Inclusion and exclusion criteria. The inclusion and exclusion criteria were established after defining the research questions. All studies related to the keyword searches across the selected sources were considered. The following inclusion and exclusion criteria (Table 9) were applied to delimit the scope of the systematic review and choose all studies that addressed the proposed research and mapping questions. It is essential to note that only studies published from 2000 onwards were included, as knowledge regarding agile contexts before that period is limited and outdated. Regarding language, only studies published in English were selected, as it has become the universal scientific language.

The study filtering stage, based on the interpretation of the inclusion and exclusion criteria, was conducted by two authors. Each author independently screened titles, keywords, and abstracts, and disagreements were discussed until a consensus was reached. As a final step to avoid bias, the other author reviewed and evaluated the screening results.

A study was discarded if it met at least two exclusion criteria. Conversely, to be accepted and included in the final analysis, a study had to satisfy the following five inclusion criteria fully: IC3 (Publication Period), IC4 (Source), IC5 (Language), IC6 (Replicability), and IC7 (Transparency), and at least one of IC1 (Applicability) or IC2 (Context).

2.2.3.2. Quality assessment criteria. Once the inclusion and exclusion criteria were applied, the methodological rigor of the studies from the primary selection was assessed using a structured evaluation framework. This framework enabled us to systematically appraise the quality of each study based on the criteria outlined in Table 10.

Table 9
The inclusion and exclusion criteria.

	Criteria	Inclusion		Exclusion
IC1	Applicability	Studies focused on the applicability of agility at both operational and organizational levels.	**EC1**	Studies focused on agility that do not address the adaptation process and applicability in organizations, including their challenges, difficulties, or issues.
IC2	Context	Studies centered on agile contexts and open-source software development contexts.	**EC2**	Studies centered on contexts unrelated to the research objective
IC3	Period	Studies published within the period 2000 onwards.	**EC3**	Studies published before the year 2000.
IC4	Source	Studies published in books, doctoral theses, dissertations, master's theses, seminars, conferences, presentations, scientific articles, journals, and technical reports. Grey literature is also included (technical documents from official websites of agile frameworks).	**EC4**	Articles from organizational websites, non-scientific magazines, or unofficial agile framework sites
IC5	Language	Studies published in English.	**EC5**	Studies not published in English.
IC6	Replicability	Studies providing detailed evidence and covering all stages of their development, allowing full reproducibility.	**EC6**	Studies that do not provide datasets or evidence allowing full reproducibility.
IC7	Transparency	Studies presenting clear, sufficient, and reliable data.	**EC7**	Studies that do not provide clear, sufficient, or detailed data

Table 10
The Quality criteria.

	Criteria	Description
C1	Importance	Relevance and impact: the degree to which the study addresses the research question (RQ1) and its contribution to understanding the specific context of the investigation.
C2	Rigor	Methodological rigor: the extent to which the study demonstrates scientific rigor in its design and execution, including the systematic application of the scientific method.
C3	Quality	Scholarly recognition: the publication venue (e.g., high-impact journals or peer-reviewed conferences) and the study's academic influence, as indicated by its citation rate.
C4	Consistency	Consistency and clarity of methodology: the transparency, quality, and internal coherence of the methodology applied, including the adequacy of data collection procedures and the robustness of the evidence presented.

Each criterion was scored on a six-point Likert scale ranging from 1 (very low) to 6 (very high). A minimum average score of 4 across all criteria was required for a study to be included in the secondary selection for detailed analysis.

To ensure alignment with the research objective, the studies included in the secondary selection were further assessed using a verification checklist (Table 11), where each item was scored as High (1), Medium (0.5), or Low (0). Studies could obtain a maximum score of 10 points (V1-V10), which was used to calculate similarity percentages in comparison to the present research. A threshold of 40 % was established, above which the continuation of this study would have been reconsidered.

2.2.4. General review process

The overall process for searching and filtering studies in this review was structured as follows.

1. Search Execution: The search equation was applied across the selected databases (Scopus, ACM, WoS, IEEE Xplore, and Google Scholar).
2. Filtering Process:
   - In the first stage, the authors reviewed the documents to detect and remove potential duplicates. The search and selection process was documented using the PRISMA flow (Fig. 4), following the guidelines proposed by Page et al. [83].

   - Subsequently, the inclusion and exclusion criteria (Table 9) were applied to the titles, abstracts, and keywords. This stage was independently conducted by two authors, who then compared their results and resolved discrepancies through discussion until consensus was reached. To further minimize bias, a third author reviewed the studies included in this phase.
   - Studies in the primary selection were then evaluated according to the quality criteria (Table 10), with a focus on the scientific rigor of each study. Those who passed this stage formed the secondary selection, which was further subjected to a 10-item verification checklist (Table 11). This analysis was performed by two authors and cross-validated by a third, ensuring a reproducible validation process.

3. Selection Results:
   - A total of 618 studies were identified in the initial search.
   - After removing duplicates and applying the inclusion and exclusion criteria, 42 studies were included in the primary selection (Table 12).
   - Following the application of quality criteria and the verification checklist, 20 relevant studies (3.5 % of the total) were retained for detailed analysis. Consequently, 96.5 % of the studies were excluded because they did not meet the established methodological standards.
4. Final Analysis: The 20 selected studies were analyzed in depth, evaluating their contributions and limitations, as well as their relevance to the main research question (RQ1). This process included reviewing terms, keywords, and relevant findings to identify existing proposals and potential gaps in the literature.

The entire process is documented and reproducible via the public datasets (Section 5.4). This expanded approach ensures transparency, reduces bias, and provides a rigorous foundation for answering the main research question (RQ1).

Table 11
Verification checklist.

Verification checklist	
1	Does the study explicitly describe the objectives of this research?
2	Is the study published within the time frame defined by the investigation?
3	Is the proposal published within the population described?
4	Does the study identify most of the relevant keywords?
5	Does the scope of the proposal align with the objectives of the research?
6	Is the proposal situated in a real organizational and business context?
7	Does the proposed solution address the complex difficulties and dynamics encountered in an agile change process of an adapted open solution?
8	Does the solution propose a scaling approach applicable to open-source software development organizations?
9	Does the proposed solution integrate the participation of free software communities?
10	Is the proposed solution intended to be tested and evaluated in open-source software development organizations?

3. Results extraction

In this section, the key findings from the systematic literature review are presented. Data were extracted systematically from each secondary selection study, including details on the agile frameworks or approaches applied, organizational contexts, types of OSS communities involved, implementation challenges, scaling methods, and reported outcomes. The extraction process aimed to identify patterns, gaps, and best practices, enabling a comprehensive understanding of how agile methodologies are applied to address the complex difficulties and challenges faced by OSS organizations and communities.

This structured extraction forms the foundation for the subsequent synthesis and discussion of results, which link empirical evidence to the research objectives and highlight direction for future investigation.

3.1. Data extraction and management

The selected studies are presented in Table 13. For each study, relevant information was systematically extracted and organized to ensure a consistent and reproducible analysis.

This systematic approach to data extraction and management ensures that the review is both transparent and reproducible, providing a solid basis for answering the research and mapping questions.

Table 12
Main review.

Source	Total Results Found	Primary Selection	Secondary Selection
SCOPUS	37	8	4
ACM	6	3	3
WOS	263	7	3
IEEE Xplore	239	13	2
Google Scholar	73	11	8
	618	42	20

Table 13
Description of eligible studies.

	Author	Title
[84]	Oktay Turetken,Igor Stojanov,Jos J. M. Trienekens	Assessing the adoption level of scaled agile development: a maturity model for scaled agile framework
[85]	Rehman A.U., Nawaz A., Ali M.T., Abbas M	A comparative study of agile methods, testing challenges, solutions tool support
[86]	Galán O.A.A., Valdéz J.L.C., Medina H.F., Gustavo A. Vanegas Contreras, Sumuano J.L.S.	Proposal of a sustainable agile model for software development
[87]	Harzl A.	Can FOSS projects benefit from integrating Kanban: a case study
[88]	Okoli C., Carillo K	The best of adaptive and predictive methodologies: Open source software development, a balance between agility and discipline
[89]	Kevin Gary, Andinet Enquobahrie, Luis Ibanez, Patrick Cheng, Ziv Yaniv, Kevin Cleary, Shylaja Kokoori, Benjamin Muffih, and John Heidenreich	Agile methods for open source safety-critical software
[90]	Periklis Tsirakidis, Felix Kobler, and Helmut Krcmar	Identification of success and failure factors of two agile software development teams in an open source organization
[91]	Beatrice Düring	Sprint driven development: agile methodologies in a distributed open-source project
[92]	Conboy, Kieran; Carroll, Noel	Implementing large-scale agile frameworks: challenges and recommendations
[93]	Dingsoyr, Torgeir; Falessi, Davide; Power, Ken	Agile development at scale: the next frontier
[94]	Brataas, Gunnar; Hanssen, Geir Kjetil; Herbst, Nikolas; van Hoorn, Andre	Agile scalability engineering: the scrumscale method
[95]	M. Jovanović, A. Mesquida, A. Mas and R. Colomo-Palacios	Agile transition and adoption frameworks, issues and factors: a systematic mapping
[96]	S. Butler et al.	On company contributions to community open source software projects
[97]	Jesús M. Gonzalez-Barahona; Daniel Izquierdo-Cortázar; Stefano Maffulli; Gregorio Robles	Understanding how companies interact with free software communities
[98]	D Salikhov, G Succi, A Tormasov	An empirical analysis of success factors in the adaption of the scaled agile framework–first outcomes from an empirical study
[99]	W Collins	Understanding scaled agile framework coordination methods for reducing failure rates
[100]	JF Carilli	The perceived effectiveness of the scaled agile framework® in software development organizations
[101]	Sandur Madhu Murthy, Srinivas	Qualitative study of scaled agile framework: an organizational approach
[102]	Igor Stojanov	Scaling agile using scaled agile framework
[103]	Send H, Hölzner H, Hassan M.	Community interaction in open business models: how IOT companies can handle community-generated innovation

3.2. Data analysis

3.2.1. MQ1: Which studies have been published each year?

This section presents the evolution of scientific production (Table 14, Fig. 5) within the investigated context, highlighting the predominant research trends over the years. The analysis shows that studies are distributed relatively evenly between 2006 and 2023, indicating both the growing interest and the consistent efforts of researchers in advancing studies in the areas of agile methodologies and open-source software development. The only exception is 2020, which exhibits the highest output (35 %) of studies focused on the agile context and the challenges of the change process.

Studies published in 2024, 2022, 2018, 2016, 2014, 2010, and between 2000 and 2008 were not selected due to their limited relevance and low alignment with the objectives and scope of this research.

Table 14
Description of eligible studies from 2000 onwards.

Year	Study	Percentage
2023	[103]	5 %
2021	[96,100]	10 %
2020	[85,86,94,95,98,99,101]	35 %
2019	[92,93]	10 %
2017	[84,87]	10 %
2015	[102]	5 %
2013	[97]	5 %
2012	[88]	5 %
2011	[89]	5 %
2009	[90]	5 %
2006	[91]	5 %

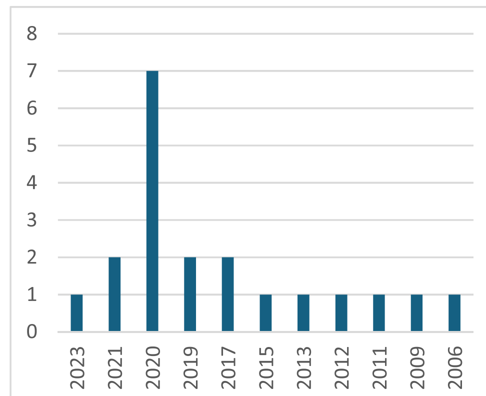

Fig. 5. Eligible studies published in the period 2000 onwards.

3.2.2. MQ2: What types of studies and proposals have been published?

This section distinguishes the formats in which the studies have been published (Table 15, Fig. 6), including books, doctoral theses, master's theses, undergraduate theses, scientific articles, or technical reports. The majority of publications are scientific articles, accounting for 75 % of the total, as these allow authors to present their research results following a rigorous scientific methodology. The remaining 25 % are represented by doctoral theses, master's theses, and books, reflecting a smaller proportion of scholarly output in these formats.

These findings underscore the significance of scientific articles as the most effective format for generating research knowledge, thereby enabling researchers to make more informed decisions when designing their studies.

3.2.3. MQ3: What is the context in which the proposals are situated?

In this section, the studies are classified according to the contexts addressed in the research. Specifically, 15 % of the proposals are situated within the free/open-source software context, 55 % within the agile

Table 15
Distribution of studies according to where they are published.

Type of study	Study	Percentage
Scientific Paper	[84–92,94–98,103]	75 %
Doctoral Thesis	[99–101]	15 %
Master Thesis	[102]	5 %
Book	[93]	5 %

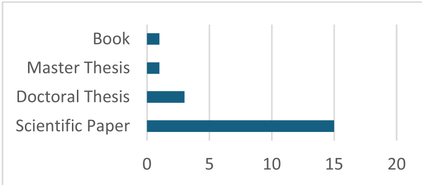

Fig. 6. Distribution of studies according to where they are published.

context, and 30 % cover both agile methodologies and open-source software development simultaneously.

The analyzed results (Table 16, Fig. 7) indicate that the majority of the selected studies (85 %) are framed within the agile context. This scientific output has increasingly attracted the interest of researchers in recent years, driven by the emergence of new agile methods, the various approaches and challenges associated with their adoption, and the growing interest of organizations in implementing agile practices [104].

Building on the success of agile methodologies, there has also been a growing interest in exploring the evident relationship and complementarity between the agile context and open-source software development (OSSD). Gandomani et al. [71] precisely define the fundamental role of shared concepts and principles between the two, highlighting the lack of empirical case studies that support the perspective that both contexts can successfully complement and collaborate in open-source software projects. Finally, despite the increasing relevance of open-source within organizations, only three studies (15 %) were found that addressed the interactions between open-source organizations and the free communities with which they collaborate.

3.2.4. MQ4: In which sources are the analyzed studies located?

The data for this section (Table 17, Fig. 8) were extracted after reviewing and mapping the different sources selected for this research.

To ensure rigor and verified evidence, this study relied on searches in specialized and well-established sources [105] (60 %), such as Scopus, ACM, WoS, and IEEE Xplore, which provide advanced search capabilities and quality filters that enhance relevance through peer-reviewed work. Additionally, indexed sources, such as Google Scholar (40 %), were enabled, allowing for comprehensive searches of academic literature across a wide range of disciplines and sources, thereby providing a broad overview of the research context. Torres-Salinas et al. [106], in line with the approach adopted in this study, recommend using Google Scholar alongside other specialized sources and advise against relying solely on it as a source of information. Although Google Scholar helps retrieve scientific contributions and identify citations received, it offers ambitious citation coverage beneficial to researchers. However, it has limitations in terms of information processing and result presentation, as it does not index all available academic sources and lacks quality filters.

3.2.5. MQ5: In which domains are these studies focused?

In this section, the extraction of results (Table 18) distinguishes between studies published with an academic research purpose and those oriented toward an organizational context. The findings reveal a highly balanced distribution across both domains, with 45 % corresponding to scholarly research and 55 % to organizational studies (Fig. 9).

These results reflect the growing interest in publishing studies aimed either at applicability in organizational contexts or at dissemination for academic purposes. This increase in the production of studies stems from the rising interest of both organizations and the scientific community in investigating the benefits of agile management and open-source software development. Merchán-Narváez et al. [107] highlight how, in the software development industry, the traditional approach established over many years has gradually been replaced by agile methodologies as a way of addressing the current challenges of software project management. A recent study conducted by the business group NTT Data [108] confirms the growth of agile methodologies in the organizational context of LATAM, driven by their associated benefits in reducing time-to-market for products and lowering corporate costs. Along the same lines, Franco Bedoya [109] notes that open-source software ecosystems (OSSECO), which encompass communities, contributors, or foundations, have attracted increasing interest from organizations, which are increasingly recognizing the differentiating potential of OSS as a source of competitive value.

Table 16
Distribution of studies according to their context.

Context	Study	Percentage
Agility	[68,69,84,85,92–95,98–100]	55 %
Agility&Free/Open source	[86–91]	30 %
Free/Open source	[96,97,103]	15 %

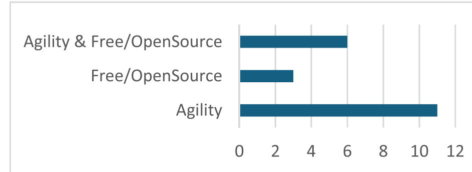

Fig. 7. Distribution of studies according to their context.

Table 17
Description studies by source.

Source	Study	Percentage
SCOPUS	[85–88]	20 %
ACM	[89–91]	15 %
WOS	[92–94]	15 %
IEEE Xplore	[95,96]	10 %
Google Scholar	[84,97–103]	40 %

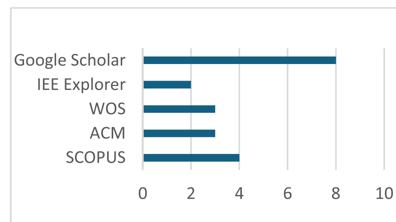

Fig. 8. List of studies by source.

Table 18
Description studies by domain.

Domains	Study	Percentage
Organizational	[90,92,94,96–103]	55 %
Academic	[84–89,91,93,95]	45 %

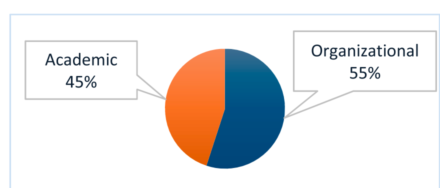

Fig. 9. Percentage of studies by domain.

3.2.6. MQ6: what types of organizations are involved?

Based on the results from MQ5, where 55 % of the studies were situated in organizational contexts, this section examines the distribution of proposals by organizational type (Table 19, Fig. 10). The results show that neither open-source software development nor agile adoption is confined to specific sectors; instead, they can be adopted across diverse organizational settings regardless of industry.

Most of the studies analyzed by Refs. [84–87,91,93] do not focus on a specific type of organization or sector. Okoli and Carillo [88] and Gary et al. [89] frame their studies within open-source software development organizations, whereas Brataas et al. [94] and Jovanović et al. [95] concentrate on open-source communities. Studies addressing the adaptation of agile methods span a range of heterogeneous organizational types, including telecommunications [100] and the public sector [92]. Additional research [96–99,101,102] focuses on more general business sectors.

3.2.7. MQ7: which agile methodological solutions are addressed in the studies?

The popularity of agile methodologies has grown exponentially in recent decades, both at the organizational and research levels. This trend is reflected in the results obtained and supported by surveys from entities such as Digital [110] and Scrum Network [111], which identify SAFe and Scrum as the most in-demand methods among organizations. Schwartz [112], CEO of Scrum Network, emphasizes the increasing relevance of Scrum in companies and the global rise of scaling frameworks such as Scrum@Scale. The results of this section (Table 20, Fig. 11) are consistent with the data reported in these surveys, showing that Scrum and SAFe are the most frequently addressed approaches in studies focused on agility.

In line with the findings of this study, Knaster and Leffingwell [113] identify SAFe as the most widely adopted robust framework among organizations seeking business agility, as it enables them to outperform competitors by leveraging Agile, Lean, and DevOps practices. This view is supported by Alqudah and Razali [114], who cite Scrum (particularly in its scaled form, such as LeSS) and SAFe as the most commonly adopted scaling frameworks in organizations. The popularity of Scrum is further emphasized by Morales Carrillo et al. [115], who identify it as the most extensively studied agile method in the literature, due to its adaptability and flexibility in software development across diverse work environments.

Scrum and SAFe emerge as the dominant frameworks in both research and practice, with Scrum being highlighted for its adaptability and SAFe for its role in enabling enterprise-level agility through Lean principles.

3.2.8. MQ8: which research categories are covered by the selected studies?

In this section, the selected studies were classified according to the knowledge domain addressed by research question RQ1, into five significant research categories (Table 21). These categories cover the most relevant aspects of knowledge within each primary domain and address the derivative questions from RQ1 (Table 7).

The distribution of studies (Table 22) shows that 85 % focus on CAT2 (Agile and open-source context), CAT4 (Challenges in agile change), CAT3 (Agile scaling methods), and CAT5 (Agile maturity), while only 15 % address CAT1 (Open-source organizations and OSS communities), highlighting the growing interest in agility across multiple contexts.

Table 19
Description studies by type of organization.

Sector	Study	Percentage
Companies in general	[96–99,101,102]	30 %
Sector specific	[90,92,100,103]	20 %
Open sources Companies and Communities OSS	[88,89,94,95]	20 %
No specific type of organization	[84–87,91,93]	30 %

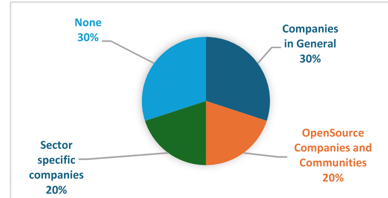

Fig. 10. Distribution of studies by type of organization.

Table 20
Description of agile methodological solutions discussed.

Methodology	Study	Percentage
SAFe	[96–99,101]	18 %
Scrum	[87,89–92]	18 %
None	[84,94,95,103]	14 %
XP	[85,86,89,91]	14 %
Combined agile methodologies	[93,98,100,102]	14 %
Kanban	[85,91]	7 %
ASD	[85,88]	7 %
DSDM	[86]	4 %
Scrum@Scale	[92]	4 %

Table 21
Description of research categories.

ID	Research categories	Description of the knowledge domain
CAT1	Open-source organizations and free communities.	Studies addressing the interactions and relationships between open-source organizations and communities to analyze the full lifecycle of contributions, answering to question SP1.
CAT2	Agile context and open-source context.	Studies exploring the relationship between the agile context and the open-source context, answering to question SP2.
CAT3	Organizational agile scaling methods.	Studies addressing various organizational agile scaling methods, answering to question SP3.
CAT4	Challenges of the agile change process.	Studies examining the challenges organizations face when initiating the agile change process, answering to question SP4.
CAT5	Degree of agile maturity.	Studies examining the level of agile maturity achieved by organizations, answering to question SP5.

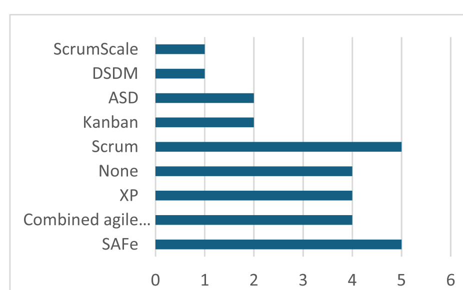

Fig. 11. Agile methodological solutions discussed.

3.2.9. RQ1: what proposals exist that offer specific agile solutions to the complex challenges and dynamics faced by OSS organizations and communities when collaborating on open-source projects?

The results of the sub-questions (SPs) derived from Research Question 1 (RQ1) are analyzed below, classified within each of the research categories (CAT1–CAT5).

Table 22
Studies by research question category.

Research question	Sub-Questions derived from RQ1	Research categories	Study	Percentage
RQ1	SP1	CAT1	[96,97,103]	15 %
	SP2	CAT2	[86–91]	30 %
	SP3	CAT3	[93,94,100, 102]	20 %
	SP4	CAT4	[85,92,95, 98,99,101]	30 %
	SP5	CAT5	[84]	5 %

3.2.9.1. SP1: how are open-source organizations and communities structured, and how do they interact in collaborative development projects?. In the first category, CAT1: Open-source organizations and communities, the contributions and limitations of the three studies examining the interactions and dynamics between open-source organizations and the communities they collaborate with are analyzed (Table 23). These studies address a relevant topic for the research question, as they contribute to understanding the complex relationship between companies and communities involved in joint open-source development projects, as well as the internal functioning of the community ecosystem. Within this category, the analysis focused on how the selected proposals address identified challenges, existing governance models, business models applied in these contexts, operational workflows, and the relationships between teams and contributors throughout the contribution lifecycle between the OSS organization and the community.

A comparative analysis was conducted (Table 24) to examine how the selected studies in this category address each of the previously identified issues (Table 1). It indicates whether an issue is addressed (✓) or not addressed (✗) and the extent to which each study contributes to the overall problem.

3.2.9.2. SP2: how do agile practices integrate with the context of open-source development in collaborative OSS environments?. Category CAT2 (Agile context and open-source context) includes six studies (Table 25) that analyze the common dynamics between agile principles and the values of open-source development. These studies examine how agile project management practices can be integrated with open-source development practices, highlighting the synergy between the two approaches in collaborative environments.

A comparative analysis was conducted (Table 26) to examine how the selected studies in this category address each of the previously identified issues (Table 1). It indicates whether an issue is addressed (✓) or not addressed (✗) and the extent to which each study contributes to the overall problem.

Table 23
Category CAT1 studies.

Study	Contributions	Limitations
[96]	Work models and the various ways in which companies and community professionals contribute to OSS projects, highlighting the channels used for contributions and interactions between organizations and communities.	Restricts the scope of work practices used by companies and does not address the broader topic of non-technical contributions, excluding code-free contributions to OSS projects developed by the community.
[97]	Company participation in large community projects, considering aspects such as activity level, neutrality, and type of collaboration.	Limited to analyzing collaboration performance between companies and communities using performance metrics.
[103]	Open business models that enable successful collaboration and serve as a source of innovation between organizations and communities.	Restricted to the IoT ecosystem, with a reduced scope in terms of the number of companies analyzed, and a bias due to interviews including only the most active community participants.

3.2.9.3. SP3: What agile scaling methods are applied in OSS contexts, and how do they adapt to the culture and structure of the organization?. Category CAT3 (Organizational agile scaling methods) comprises four studies (Table 27) that analyze the application of popular agile scaling frameworks within organizations. These studies examine how scaling methods adapt to organizational culture and structure, and how they impact the agile scaling process, the project lifecycle, and the day-to-day operations of software development teams.

A comparative analysis was conducted (Table 28) to examine how the selected studies in this category address each of the previously identified issues (Table 1). It indicates whether an issue is addressed (✓) or not addressed (✗) and the extent to which each study contributes to the overall problem.

3.2.9.4. SP4: What are the main challenges and difficulties encountered during the adoption of agile practices in OSS organizations and communities?. Category CAT4 (Challenges of the agile change process) comprises six studies (Table 29) that identify the primary problems organizations encounter when scaling agility. These studies highlight the challenges in adapting the implementation roadmaps prescribed in agile methods to the specific needs of the organization, emphasizing the importance of effectively managing these challenges to achieve a successful agile transformation. Understanding these challenges is vital in implementing effective agile solutions that address organizational and operational problems.

A comparative analysis was conducted (Table 30) to examine how the selected studies in this category address each of the previously identified issues (Table 1). It indicates whether an issue is addressed (✓) or not addressed (✗) and the extent to which each study contributes to the overall problem.

3.2.9.5. SP5: What is the level of maturity in terms of agile adoption of OSS organizations and communities, and how is this maturity assessed or measured?. The Category CAT5 (Agile Maturity Level) includes a study (Table 31) that analyzes organizational agile maturity models. This study highlights the importance of assessing the current maturity levels of both the organization and its associated community, providing a solid foundation from which to plan and implement effective agile solutions tailored to their respective maturity levels.

A comparative analysis was conducted (Table 32) to examine how the selected studies in this category address each of the previously identified issues (Table 1). It indicates whether an issue is addressed (✓) or not addressed (✗) and the extent to which each study contributes to the overall problem.

4. Discussion

The review reveals a growing research interest in agility and open-source software, highlighting their transformative potential in organizational operations and structures. Emerging agile models and innovative applications developed in open-source software in sectors as varied as AI, robotics, and IoT emphasize the need to promote academic research aligned with the interests of organizations, supporting and accompanying them in an increasingly dynamic environment that requires innovation.

4.1. Interpretation of findings

The analysis of the 20 studies, based on RQ1 (with its five SPs) and the MQ1-MQ8 mapping questions, provides a clear view of the state of the literature and shows how the identified problems are addressed, offering information on the practices, methods, and organizational approaches used to address them, as well as possible research gaps.

Table 24
Analysis of category CAT1 studies in relation to the problem.

	Study	P1	P2	P3	P4	P5	P6	P7	P8	P9	Total	Media	Deviation	Variance
CAT1	[96]	(✓)	(✓)	(✗)	(✗)	(✓)	(✗)	(✗)	(✓)	(✗)	4	0,444	0.527	0.2778
	[97]	(✗)	(✗)	(✓)	(✗)	(✓)	(✗)	(✗)	(✗)	(✗)	2	0.222	0.441	0.1944
	[103]	(✗)	(✓)	(✗)	(✓)	(✗)	(✗)	(✗)	(✗)	(✗)	2	0.222	0.441	0.1944

Table 25
Category CAT2 studies.

ID	Contributions	Limitations
[86]	Achieving open-source sustainability through reuse, presenting a business model focused on agile open-source development based on reuse and quality.	Focused only on achieving sustainability as a driver of organizational growth in OSS development, an aspect not currently addressed by agile methodologies.
[87]	Challenges of using and integrating Kanban agile practices in open-source development teams, providing practical insights into combining ASD with FOSS.	Limited external validity, as the study is based on a hybrid FOSS project designed for students; results should not be generalized to other OSS development scenarios.
[88]	Benefits of synthesizing agile and waterfall methodologies in OSS development projects involving OSS communities.	OSSD may not be the best option when organizations and communities do not share goals or are not aligned; applicability is restricted to product-driven contexts emphasizing the synthesis of agile and waterfall methods.
[89]	Reliability and security of agile practices applied to open-source development in critical healthcare environments.	Limited to the healthcare domain, reporting a practical trial across 27 hospitals and research centers worldwide, but lacking the rigor required for a critical safety domain.
[90]	Success and failure factors in adopting an agile culture within open-source development teams in an OSS company.	Restricted to applying Agile at the operational level of development teams and to the consistency of the methodology used as a success factor in adoption.
[91]	Challenges of adopting iterative agile methods in a distributed community-driven OSS project, highlighting similarities between PyPy development practices and those recommended in agile methodologies.	Limited to the specific context and characteristics of the PyPy project developed by the Python community.

4.1.1. *Evaluation of findings from the mapping questions (MQ1–MQ8)*

The overall findings from the selected studies (Table 33), classified according to each of the mapping questions (MQ1-MQ8), are summarized below.

Most of the findings are from the period from 2020 to 2023, representing 50 % of the studies, which reflects a growing interest in emerging contexts, such as agility and open-source software. Studies are almost equally divided between organizational (55 %) and academic (45 %) contexts, with scientific articles being the preferred publication format (75 %) by authors.

Specialized databases (60 %), such as Scopus, ACM, WOS, and IEEE Xplore, dominate indexed sources (40 %), including Google Scholar, due to their advanced search capabilities and peer-reviewed content. In terms of context, 85 % of studies address agility, compared to 15 % that focus on free/open-source, with SAFe and Scrum being the most

Table 27
Category CAT3 studies.

ID	Contributions	Limitations
[93]	Challenges of developing software at scale using agile methodologies such as Scrum, XP or Kanban.	Limited by the lack of independent studies on agile scaling methodologies.
[94]	Lessons learned from scaling agility with Scrum@Scale and its organizational effects.	A key limitation of this scaling method is the difficulty of maintaining an overall view of scalability during continuous development, which negatively affects transparency in scalability management.
[100]	Effectiveness of applying agile scaling with SAFe in organizations with waterfall-based project management.	Constraints related to the number of organizations and participants involved in the study; inability to compare and contrast results from organizations transitioning to SAFe from other methods; and limitations in the obtained metrics due to differences between Waterfall and SAFe in measuring outcomes.
[102]	SAFe implementation process based on a maturity model and improved software processes.	Restricted to a single case-study organization, making results context-specific and not generalizable; the proposed model may also face industry resistance as it is based on predefined agile adoption plans.

Table 26
Analysis of category CAT2 studies in relation to the problem.

	Study	P1	P2	P3	P4	P5	P6	P7	P8	P9	Total	Media	Deviation	Variance
CAT2	[86]	(✗)	(✓)	(✗)	(✗)	(✗)	(✗)	(✗)	(✗)	(✗)	1	0.111	0.3333	0.1111
	[87]	(✗)	(✗)	(✓)	(✗)	(✗)	(✓)	(✗)	(✗)	(✗)	2	0.222	0.441	0.1944
	[88]	(✗)	(✗)	(✓)	(✗)	(✗)	(✗)	(✓)	(✗)	(✗)	2	0.222	0.441	0.1944
	[89]	(✗)	(✓)	(✓)	(✗)	(✗)	(✗)	(✗)	(✗)	(✗)	2	0.222	0.441	0.1944
	[90]	(✗)	(✓)	(✗)	(✗)	(✗)	(✗)	(✗)	(✗)	(✗)	1	0.111	0.3333	0.1111
	[91]	(✗)	(✓)	(✓)	(✗)	(✗)	(✗)	(✗)	(✗)	(✗)	2	0.222	0.441	0.1944

Table 28
Analysis of category CAT3 studies in relation to the problem.

	Study	P1	P2	P3	P4	P5	P6	P7	P8	P9	Total	Media	Deviation	Variance
CAT3	[93]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✗)	(✗)	(✓)	3	0.3333	0.5	0.25
	[94]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✓)	(✗)	(✗)	3	0.3333	0.5	0.25
	[100]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✗)	(✗)	(✓)	3	0.3333	0.5	0.25
	[102]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✗)	(✗)	(✓)	3	0.3333	0.5	0.25

Table 29
Category CAT4 studies.

ID	Contributions	Limitations
[85]	Solutions to challenges identified within the agile development cycle, particularly from the perspective of agile testing.	Limited to application within the agile development team context and code testing.
[92]	Challenges encountered during agile transformation in organizations and recommendations to address them.	Constraints on full implementation of recommendations due to cost, cultural, structural, and scope-related restrictions.
[95]	Identification of 154 situational factors that impact and influence the success of agile transformation and adoption.	Scope is too general when comparing existing agile adoption frameworks up to 2019.
[98]	Challenges and difficulties encountered when scaling agility using SAFe.	Restricted to the operational and cultural specifics of a single aerospace organization and a particular scaling method.
[99]	Methods, interrelations, leadership, coordination, and problematic factors involved in implementing SAFe in large organizational projects.	Limited applicability of the case study to other organizational contexts, as it focuses on a single project.
[101]	Challenges faced by organizations when scaling agility through prescriptive frameworks.	Scope restricted to the most popular prescriptive agile frameworks within a generic scaling context.

frequently analyzed agile methods (36 %) and being present in the studies.

These results are consistent with the concentration of studies in the CAT2, CAT3, CAT4, and CAT5 categories (85 %), which examine agile scaling methods, challenges in adoption and scaling, common dynamics with open source development, and agile maturity assessment, highlighting the current trend of the research community to prioritize agility over the context of open source development (15 %), both in theoretical and practical domains.

4.1.2. Evaluation of findings from RQ1 and its sub-questions (SPs)

All the selected studies address the content of RQ1 and its sub-questions, and through their findings, contribute to addressing the research problems identified (Table 1) in various ways and to varying degrees (Table 34, Figs. 12 and 13).

CAT1 – Open-source organizations and communities: Studies by Refs. [96,97], and [103] focus on practices, interactions, and contributions in OSS projects. Companies' governance models, business strategies, and collaboration mechanisms are analyzed, highlighting how organizations and communities align to foster open innovation. These findings provide insights into how the literature addresses the nature of contributions and interactions between organizations and communities (P5), identifying roles (P8), governance models (P1), business models (P2), operational models (P3), and participation channels for idea and innovation contributions (P4) used in these collaborative contexts.

CAT2 – Agile and open-source context: The research examines the integration of agile methodologies with OSS practices. All the findings extracted from these proposals highlight the complementarity and shared dynamics between the principles and values of agility and open-source software, demonstrating a considerable increase in the value delivered by organizational projects when agile practices are integrated with open-source development practices.

These findings offer insight into how the literature addresses business models (P2) and operating models (P3) within this category, particularly in contexts where organizations adopt agile management and integrate it into open-source development [86–91]. They also show

Table 31
Category CAT5 studies.

ID	Contributions	Limitations
[84]	Demonstrates the usefulness of a maturity model for assessing the level of SAFe adoption within an organization.	Limited by the number of participants in the Delphi expert round and the organizational members involved in evaluating the model, which may affect the results obtained.

Table 30
Analysis of category CAT4 studies in relation to the problem.

	Study	P1	P2	P3	P4	P5	P6	P7	P8	P9	Total	Media	Deviation	Variance
CAT4	[85]	(✗)	(✗)	(✗)	(✗)	(✓)	(✗)	(✗)	(✗)	(✗)	1	0.111	0.333	0.11
	[92]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✓)	(✗)	(✗)	3	0.333	0.5	0.25
	[95]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✓)	(✗)	(✗)	3	0.333	0.5	0.25
	[98]	(✗)	(✗)	(✗)	(✗)	(✓)	(✗)	(✗)	(✗)	(✗)	1	0.111	0.333	0.11
	[99]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✗)	(✓)	(✗)	3	0.333	0.5	0.25
	[101]	(✗)	(✗)	(✗)	(✗)	(✓)	(✓)	(✗)	(✗)	(✗)	2	0.222	0.441	0.194

Table 32
Analysis of category CAT4 studies in relation to the problem.

	Study	P1	P2	P3	P4	P5	P6	P7	P8	P9	Total	Media	Deviation	Variance
CAT5	[84]	*(X)*	*(X)*	*(X)*	*(V)*	*(X)*	*(X)*	*(X)*	*(X)*	*(V)*	2	0.222	0.441	0.1944

Table 33
Total findings from mapping questions.

ID			Studies	Percentage
MQ1	Year	**2020**	[85,86,94,95,98,99,101]	**35 %**
		2017,2019,2021	[84,87,92,93,96,100]	30 %
		2006,2009,2011,2012,2013,2015,2023	[88–91,97,102,103]	35 %
MQ2	Method	**Scientific Paper**	[84–92,94–98,103]	**75 %**
		Doctoral Thesis	[99–101]	15 %
		Master Thesis	[102]	5 %
		Book	[93]	5 %
MQ3	Context	**Agility**	[59,60,65,66,68,69,84,85,94,95,100]	**55 %**
		Agility and free/open-source	[86–91]	30 %
		free/open-source	[96,97,103]	15 %
MQ4	Source	**Google Scholar**	[84,97–103]	**40 %**
		SCOPUS	[85–88]	20 %
		ACM	[89–91]	15 %
		WOS	[92–94]	15 %
		IEEE Xplore	[95,96]	10 %
MQ5	Domain	**Organizational**	[90,92,94,96–103]	**55 %**
		Academic	[84–89,91,93,95]	45 %
MQ6	Sector	**No specific type of organization**	[84–87,91,93]	**30 %**
		Companies in general	[96–99,101,102]	**30 %**
		Open sources and Communities	[88,89,94,95]	20 %
		Sector specific	[90,92,100,103]	20 %
MQ7	Methodology	**SAFe**	[96–99,101]	**18 %**
		Scrum	[87,89–92]	**18 %**
		XP	[85,86,89,91]	14 %
		Combined agile methodologies	[93,98,100,102]	14 %
		None	[84,94,95,103]	14 %
		Kanban	[85,91]	7 %
		ASD	[85,88]	7 %
		DSDM	[86]	4 %
		Scrum@Scale	[92]	4 %
MQ8	Category	**CAT2**	[86–91]	**30 %**
		CAT4	[85,92,95,98,99,101]	**30 %**
		CAT3	[93,94,100,102]	20 %
		CAT1	[96,97,103]	15 %
		CAT5	[84]	5 %

Table 34
Contribution of studies by category to the problem.

	Study	Problem Scope Treaty	Total	Percentage per study	Percentage by Category
CAT1	[96]	P1, P2, P5, P8	4	9 %	18 %
	[97]	P3, P5	2	4 %	
	[103]	P2, P4	2	4 %	
CAT2	[86]	P2	1	2 %	23 %
	[87]	P3, P6	2	4 %	
	[88]	P3, P7	2	4 %	
	[89]	P2, P3	2	4 %	
	[90]	P2	1	2 %	
	[91]	P2, P3	2	4 %	
CAT3	[93]	P5, P6, P9	3	7 %	25 %
	[94]	P5, P6, P7	3	7 %	
	[100]	P5, P6, P9	3	7 %	
	[102]	P5, P6, P9	3	7 %	
CAT4	[85]	P5	1	2 %	30 %
	[92]	P5, P6, P7	3	7 %	
	[95]	P5, P6, P7	3	7 %	
	[98]	P5	1	2 %	
	[99]	P5, P6,P8	3	7 %	
	[101]	P5, P6	2	4 %	
CAT5	[84]	P4, P9	2	4 %	4 %

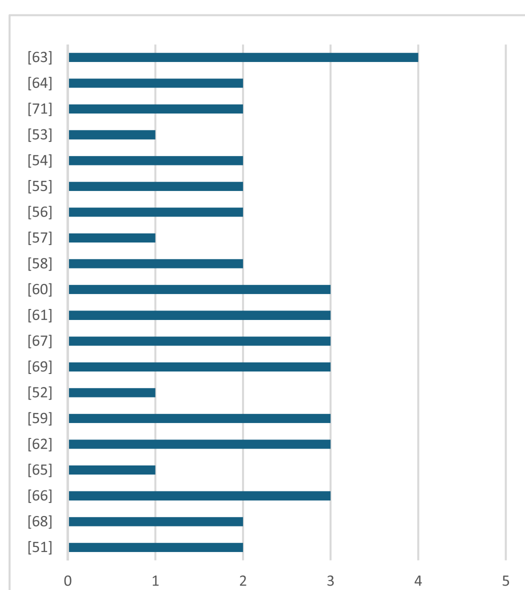

Fig. 12. Problematic contribution by study.

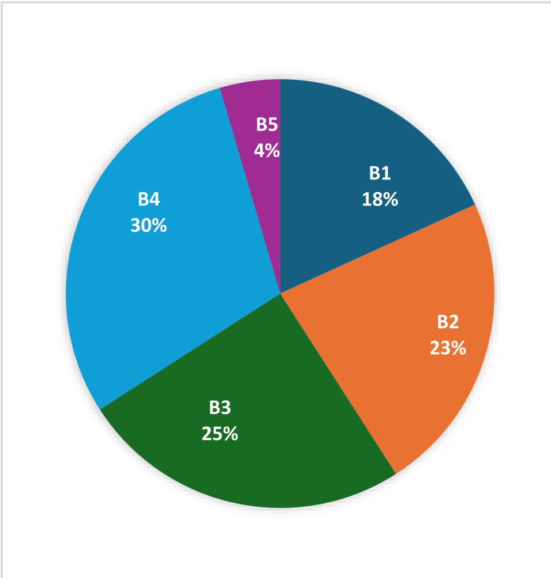

Fig. 13. Percentage of study contribution by category.

how workflow (P6) in FOSS teams is significantly improved through the application of agile practices [87] and how teams achieve better synchronization and reduce dependencies (P7) when agile management is introduced into the open-source development cycle [88].

CAT3 – Organizational agile scaling methods: The studies [93,94,100,102] investigate the application of scaling frameworks such as Scrum, XP, Scrum@Scale, and SAFe. Their findings reveal the operational and cultural challenges associated with adopting traditional agile methods, as well as the benefits achieved through adoption and scaling, highlighting the early delivery of value, improved workflow, cost reduction, and enhanced software quality. These results highlight the predominant role of Scrum and SAFe in the market as the most widely used prescriptive scaling frameworks, as they can comprehensively manage the lifecycle of organizational change and its operational processes (P5, P6, P7), while allowing progress and results to be measured throughout the process (P9).

CAT4 – Challenges of the agile transformation process: The research identifies the most common difficulties in scaling agility, including adapting the framework to the company's context, resistance to change, and situational factors. The evidence highlights the complexity of agile adoption and scaling, as well as the significant cultural challenges that accompany it, offering guidance on how to address organizational and operational obstacles. The findings emphasize the most frequent challenges across key aspects (P5, P6, P7, P8) for organizations [95,101] when scaling widely adopted frameworks such as SAFe, LeSS, and Scrum [92,98,99], and how these challenges directly impact software development processes [85].

CAT5 – Agile maturity assessment: In this category, the research highlights the evaluation of an organization's agile maturity as a foundation for successful transformation. Maturity models, such as the SAFe adoption assessment validated through the Delphi method [84], provide practical insights for establishing the initial paradigm before implementing agile changes. Furthermore, they enable organizations to identify their level of innovation (P4) and to determine which progress measurement systems (P9) are being used in relation to the level of maturity achieved**.**

This review highlights how the integration of agility and OSS, agile scaling, challenge management, and maturity assessment complements each other as critical elements for achieving sustainable and successful agile transformations within organizations.

Table 35 illustrates the distribution of studies cited in relation to their contribution (Fig. 14) to the identified problems (Table 1), providing valuable insights into the current state of the literature.

Additionally, it highlights gaps in areas that remain underexplored, particularly in key areas where the existing literature offers limited knowledge. Leadership (P8) is one such area, with few agile solutions for community-based meritocratic roles such as the Benevolent Dictator for Life (BDFL). Similarly, Governance (P1) remains underexplored: only a few studies, such as Butler et al. [96], examine the channels for contributions and interactions between companies and communities, and analyze governance systems. Zanotti and Vélez [116] describe three levels of governance (software, community, and ecosystem) that underpin community participation in FLOSS ecosystems; yet, practical guidance on implementing these governance models in real organizational contexts is scarce. These findings suggest that community governance, although more participatory and open compared to traditional management structures, remains an area with limited empirical evidence, where the interplay between organizational and community dynamics is still poorly understood [117].

Team alignment and dependency management (P7) also receive limited attention. Existing studies examine approaches to managing dependencies through synchronization between organizational teams and community contributors, including alignment via Communities of Practice (CoPs) as described by Wenge- [118]. According to Giraldo Marín and Atehortúa Correa [119], these communities can serve as spaces for democratizing knowledge and aligning groups of individuals with shared interests within organizations, potentially evolving into informal entities with their own learning life cycle [120]. Other underexplored aspects include community-driven open innovation (P4) and the roles of organizations and communities (P8).

In contrast, the dimensions that have received the most academic attention include the challenges in the contribution lifecycle (P5) and workflow management (P6), as both address aspects related to adoption, agility, and scaling. Also noteworthy are the number of studies that deal with business models (P2) within the software industry, in line with Juncos and Borrastero [121], as well as the study of the different operating models of work (P3) applicable to collaborative contexts.

These patterns indicate where the research community has focused its efforts on finding solutions to specific problems and where further research is still needed.

4.2. Research limitations

This review is limited to literature found in English in the period between 2000 and onwards. Another limitation is the scope of the identified problem (Table 1) and the objectives of the change of the proposals put forward (Fig. 2), since these may vary from one organization to another in the context investigated, causing that the complete elimination of the difficulties and dynamics identified is not possible to carry out in all the areas described, due to the intrinsic particularity of each organization in aspects related to its culture, structure and operations.

The review highlights several critical gaps in the literature that limit

Table 35
Distribution of studies by problem area.

ID	Problem area	Studies	Total	Percentage
P1	Governance	[96]	1	2 %
P2	Business Model	[86,89–91,96,103]	6	13 %
P3	Operational Model	[87–89,91,97]	5	11 %
P4	Innovation	[84,103]	2	4 %
P5	Contribution Lifecycle	[59,60,85,94–102]	12	26 %
P6	Workflow	[59,60,66,67,87,94,95,101,102]	9	19 %
P7	Alignment and Synchronization	[59,61,88,95]	4	9 %
P8	Leadership	[96,99]	2	4 %
P9	Measurement Culture	[84,93,100,102]	4	9 %

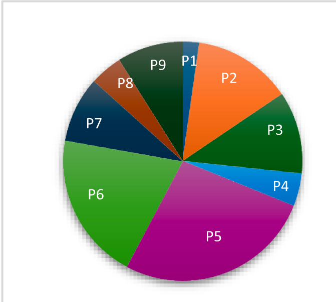

Fig. 14. Percentage by problem area.

a comprehensive understanding of agile adoption and open-source collaboration in organizational contexts. First, leadership (P8) and governance (P1) remain underexplored areas of focus. Few studies provide actionable guidance on community-based meritocratic roles or practical frameworks for implementing governance models in real organizational settings [96,116,117]. Similarly, community-driven open innovation (P4) and organizational and community roles (P8) are insufficiently addressed, leaving unclear how organizations can leverage collective knowledge effectively in agile OSS projects.

Another significant gap lies in the study of team alignment and dependency management (P7). While approaches such as Communities of Practice (CoP) are mentioned [118–120], empirical evidence on their practical effectiveness in aligning organizational and community contributors is limited. Moreover, the literature rarely explores the creation and role of symbiotic communities that work closely with organizations on open-source projects, despite their potential impact on project sustainability and profitability [97,122]. Research on the integration of FOSS solutions with agile development in collaborative contexts is also sparse, with most studies treating these domains separately. Studies on agile scaling methods are limited, with SAFe dominating due to its prescriptive roadmap, while other frameworks, such as LeSS, DAD, and Nexus, are less examined. There is a pressing need for more case studies across diverse organizations and contexts to understand how agile transformations can be effectively adapted to specific organizational challenges, especially considering the success of organizations that design custom frameworks to meet their unique needs, such as Spotify, Nokia, or Ericsson.

Finally, there is a notable lack of research on the combined scaling of agile practices within open-source organizations and free communities. This indicates a broader gap in understanding how large-scale transformations impact both operational and organizational dimensions, highlighting the need for future studies to provide integrated, practical approaches for sustainable and effective adoption of agile methods in collaborative OSS environments.

5. Threats to validity

Both FOSS and agile methodologies generate numerous conflicting opinions and prejudices, making it necessary to avoid what authors such as Kahneman [123] define as cognitive biases, which arise from previous experiences, insecurities, and internalized preconceptions. Failure to mitigate these biases could result in a partial or biased interpretation of the results. To identify and mitigate potential threats to the validity of this systematic review, we adopted the categories proposed by Wohlin et al. [124], which cover construct validity, internal validity, external validity, and conclusion validity. These categories are widely used to address and reduce biases in numerous systematic reviews, as highlighted by Franco-Bedoya et al. [78].

5.1. Construct validity

To ensure the rigor and reliability of this systematic review, potential threats to construct validity were carefully considered and addressed. Construct validity refers to the degree to which the research design, instruments, and procedures accurately capture the concepts intended to be studied. In the context of this review, three main threats were identified: the specificity of the search protocols, the alignment of the research question with the study objectives, and the precise definition of the research contexts. Table 36 summarizes these threats and the measures implemented to mitigate them, demonstrating the systematic efforts undertaken to enhance the accuracy, comprehensiveness, and relevance of the study's findings.

5.2. Internal validity

Table 37 identifies two threats to internal validity.

5.3. External validity

No threats to external validity were identified, as the results and conclusions drawn are confined to specific and well-defined domains, such as agile and open-source software.

5.4. Conclusion validity

To ensure the reproducibility of the results by future researchers, this document provides detailed descriptions of all search protocols and terms used in the two bibliographic reviews conducted. Additionally, links are provided to access all extracted results, which are available in two Excel files located at umbrella review [125] and systematic review [126].

6. Conclusions

In this multifaceted review, we analyzed both peer-reviewed articles and grey literature, as excluding the latter would have omitted significant information on the applicability of agile methods within the

Table 36
Description of threats and mitigation measures.

Threats	Mitigation Measures
The defined search protocols are not sufficiently specific to identify all published articles.	✓ Inclusion of customized search protocols according to the characteristics of each database. ✓ Inclusion of a wide range of terms in these protocols to cover all aspects of the research. ✓ Establishment of a broad search period for the studies. ✓ Inclusion of the most relevant databases with the highest number of scientific publications in the research area. ✓ Use of Google Scholar to further expand the search scope. ✓ Definition of concise inclusion and exclusion criteria aligned with the review objectives. ✓ Definition of quality criteria. ✓ Definition of a checklist.
The research question may not accurately reflect the objectives of the study.	✓ Formulation of a research question based on the author's knowledge and experience, faithfully reflecting the objectives to be achieved.
The contexts have not been precisely defined.	✓ Detailed inclusion in the search protocols of all possible concepts and terms relevant to the two contexts considered in the research (agility and free/open-source software).

Table 37
Description of threats and their mitigation measures.

Threats	Mitigation Measures
The lack of a clear definition of the agile approach by many authors.	✓ In the proposals related to the agile scaling process, a diversity of concepts and terms is used, such as methods, methodologies, maturity models, and frameworks.
Considerable uncertainty regarding whether the proposals are intended for an organizational or academic context.	✓ While some proposals clearly identify the type of organization they target, many are difficult to classify within a real-world organizational context or an academic research context.

research categories (CAT1–CAT5) considered in this study. The literature still shows limitations due to the scarcity of scientific production specifically addressing these categories, as much of the knowledge on agile implementation processes, applied methods, challenges encountered, and valuable feedback is located in the repositories of official agile method sites and the organizations themselves. This study allowed us to identify several relevant findings related to the research problem, with the most significant contributions situated within the operational dimension of organizations and in relation to agility (P5, P6, P7, P8, P9). Equally notable were contributions to the organizational dimension, specifically regarding business models (P2) and operating models (P3) employed by these organizations and communities, which offer a broad perspective on how these topics are currently addressed in the literature.

However, research on the application of combined agile solutions in collaborative open-source development remains limited, particularly regarding the integration of collective intelligence and open innovation from OSS communities. Open and hybrid approaches that combine prescriptive and non-prescriptive agile methods offer promising avenues to address these challenges, as demonstrated by organizations such as Spotify [127] and Zappos [128]. These organizations not only became agile with decentralized structures but, in Spotify's case, also transformed and customized their agile culture and operations to adapt to the challenges of their specific context. Survey evidence from Digital [110] and Scrum Network [111] further indicates a growing organizational preference for customized or non-prescriptive frameworks over traditional prescriptive methods such as SAFe or Scrum@Scale.

Future research should focus on improving the understanding of the various actors involved in open-source software development projects, particularly the role played by symbiotic communities. In addition, studies should explore new ways to encourage and manage the participation of the OSS community in more organizational projects, identifying disruptive innovations that can be incorporated into organizational portfolios. Expanding independent studies on prescriptive and non-prescriptive agile scaling frameworks is essential to inform organizational decision-making and enhance agile adoption practices, particularly in high-variability contexts where synergies between various agile solutions are beneficial in addressing specific issues.

Overall, achieving sustainable organizational agility in collaborative contexts requires aligning the various solutions proposed in this study with an organizational design that is sensitive to the problems of this context, which includes an understanding of the operational and structural dynamics of symbiotic communities.

CRediT authorship contribution statement

José Luis González-Blázquez: Writing – original draft. **Alicia García-Holgado:** Writing – review & editing, Supervision. **Francisco José García-Peñalvo:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The dataset generated by the research is available at: umbrella review: https://doi.org/10.5281/zenodo.17338512 and Systematic review: https://doi.org/10.5281/zenodo.17338592.

References

[1] Morero HA, Fernandez V, Sonnenberg Palmieri J. International surveys of free and open source software (FLOSS) companies [Online]. Available, http://hdl.handle.net/11086/4591. [Accessed 29 August 2025].
[2] Morero HA. The economy of free and open-source software: multinationals, pymes, and communities. Estudios Sociológicos Editora 2024 [Online]. Available: http://estudiosociologicos.org/-descargas/eseditora/La-economia-del-software-libre/La-economia-del-software-libre_Morero-Motta.pdf.
[3] Gamalielsson J, Lundell B. Sustainability of open source software communities beyond a fork: how and why has the LibreOffice project evolved? J Syst Software Mar. 2014;89:128–45. https://doi.org/10.1016/j.jss.2013.11.1077.
[4] González-Blázquez JL, García-Holgado A, García-Peñalvo FJ. Open approach of scaled agile for organizations and communities dedicated to the development of open-source projects. In: Ninth international conference on technological ecosystems for enhancing multiculturality (TEEM'21). Barcelona Spain: ACM; Oct. 2021. p. 752–7. https://doi.org/10.1145/3486011.3486543.
[5] Tessore C, Página CP. 'Project Management - contemporary Developments'. 2022. https://doi.org/10.13140/RG.2.2.16914.12480.
[6] de Jiménez RLM. Agile software development methodologies applied to business project management. Rev Tecnol 2015;(8).
[7] Valderrama B. Digital transformation and agile organizations. ARANDU UTIC 2019;6(1):15–50.
[8] Kotter J. The leader of change. McGraw-Hill; 1997.
[9] Laloux F. Reinventar las organizaciones. first ed. Barcelona: Arpa Editores; 2016.
[10] Izquierdo JLC, Cabot J. For a more transparent governance of open source. Commun ACM Aug. 2023;66(8):32–4. https://doi.org/10.1145/3570635.
[11] García-García J, Alonso de Magdaleno MI. Open source models and social responsibility [Online]. Available: https://mpra.ub.uni-muenchen.de/34684/. [Accessed 3 December 2024].
[12] Meissner JO, Heike M, Sigrist D. Agile management. In: Organizational design in a complex and unstable world. Wiesbaden: Springer Fachmedien Wiesbaden; 2024. p. 71–89. https://doi.org/10.1007/978-3-658-45022-9_5.
[13] Flores-Cerna F, Sanhueza-Salazar V-M, Valdés-González H-M, Reyes-Bozo L. Agile methodologies: an analysis of organizational challenges for their implementation. Rev Científica (Maracaibo) Dec. 2021;43(1):38–49. https://doi.org/10.14483/23448350.18332.
[14] Díaz JR. 'New business models based on free software', studylib.es [Online]. Available: https://studylib.es/doc/4495150/nuevos-modelos-de-negocio-basados-en-software-libre. [Accessed 29 August 2025].
[15] Kucherbaev P, et al. 'Business processes for the crowd computer', in business process management workshops. In: La Rosa M, Soffer P, editors. Lecture notes in business information processing, vol. 132. Berlin, Heidelberg: Springer Berlin Heidelberg; 2013. p. 256–67. https://doi.org/10.1007/978-3-642-36285-9_31. 132.
[16] Cooper RG, Sommer AF. Agile-stage-gate: new idea-to-launch method for manufactured new products is faster, more responsive. Ind Mark Manag Nov. 2016;59:167–80. https://doi.org/10.1016/j.indmarman.2016.10.006.
[17] Cánovas Izquierdo JL, Cabot J. On the analysis of non-coding roles in open source development: an empirical study of NPM package projects. Empir Softw Eng Jan. 2022;27(1):18. https://doi.org/10.1007/s10664-021-10061-x.
[18] Bozheva T, Anderson D. Kanban maturity model. DYNA Manag June 2021;7(1). https://doi.org/10.6036/MN8805 [10 pp.]-[10 pp.].
[19] García Pozo J. '35 years of open source software, pitfalls. Vulnerabilities and Tech Debt', Sept 2024 [Online]. Available: https://hdl.handle.net/10016/46257. [Accessed 29 August 2025].
[20] Schneider N. Admins, mods, and benevolent dictators for life: the implicit feudalism of online communities. New Media Soc 2022;24(9):1965–85. https://doi.org/10.1177/1461444820986553. Sept.
[21] Ramirez J, Reyes CJ, Gil GD. Source code metrics and evolution of F/OSS: an exploratory study. XIV Simp Argent Ing Softw ASSE-JAIIO 2013;42.
[22] Lozano Ardila JA. Holacracy, an organizational management framework to drive strategy. A literature review'; Oct. 2024 [Online]. Available: https://hdl.handle.net/10818/63540. [Accessed 6 October 2025].
[23] Hirth M, Hoßfeld T, Tran-Gia P. Anatomy of a crowdsourcing platform - using the example of microworkers.com. In: 2011 fifth international conference on innovative Mobile and internet services in ubiquitous computing; June 2011. p. 322–9. https://doi.org/10.1109/IMIS.2011.89.
[24] Chen X, Usman M, Badampudi D. Using InnerSource for improving internal reuse: an industrial case study. In: Proceedings of the 27th international conference on evaluation and assessment in software engineering. Oulu Finland: ACM; June 2023. p. 348–57. https://doi.org/10.1145/3593434.3593466.
[25] D. Cooper and K.-J. Stol, 'Adopting InnerSource'.

[26] Morgan L, Gleasure R, Baiyere A. Is inner source the next stage in the agile revolution? In: Elbanna A, McLoughlin S, Dwivedi YK, Donnellan B, Wastell D, editors. Co-creating for context in the transfer and diffusion of IT. Cham: Springer International Publishing; 2022. p. 130–6. https://doi.org/10.1007/978-3-031-17968-6_10.

[27] Wangsa K, Chugh R, Karim S, Sandu R. A comparative study between design thinking, agile, and design sprint methodologies. Int J Agile Syst Manag 2022;15 (2):225. https://doi.org/10.1504/IJASM.2022.124916.

[28] Klaus-Rosińska A, Gąsiorowska A. Project management in IT with LeSS scrum methodology. Procedia Comput Sci 2025;256:1962–8. https://doi.org/10.1016/j.procs.2025.02.339.

[29] Almeida F, Espinheira E. Large-scale agile frameworks: a comparative review. J Appl Sci Manag Eng Technol Mar. 2021;2(1):16–29. https://doi.org/10.31284/j.jasmet.2021.v2i1.1832.

[30] Cooper RB. Queueing theory. In: *Proceedings of the ACM '81 conference on - ACM 81*, not known. ACM Press; 1981. p. 119–22. https://doi.org/10.1145/800175.809851.

[31] Senapathi M, Drury-Grogan ML. Systems thinking approach to implementing Kanban: a case study. J Softw Evol Process 2021;33(4):e2322. https://doi.org/10.1002/smr.2322.

[32] Antolinez SV, Trujillo MV. Communities of practice and their strategy according to life cycle stage: a systematic review based on PRISMA guidelines. Económicas CUC 2023;44(1):175–202. https://doi.org/10.17981/econcuc.44.1.2023.Org.5.

[33] Brown JS, Duguid P. Organizational learning and Communities-of-Practice: toward a unified view of working, learning, and innovation. Organ Sci Feb. 1991; 2(1):40–57. https://doi.org/10.1287/orsc.2.1.40.

[34] Farnsworth V, Kleanthous I, Wenger-Trayner E. Communities of practice as a social theory of learning: a conversation with Etienne Wenger. Br J Educ Stud Apr. 2016;64(2):139–60. https://doi.org/10.1080/00071005.2015.1133799.

[35] unFIX, 'Coordination & integration', Large Scale Scrum (LeSS). Accessed: October. 10, 2025. [Online]. Available: https://less.works/less/framework/coordination-and-integration.

[36] Prasongko A, Adianto T. The role of the Agile Leadership model as a competitive advantage for the future leader in the era of globalization and industrial revolution 4.0. J Pertahanan Media Inf Tentang Kaji Dan Strategi Pertahanan Yang Mengedepankan Identity Nasionalism Dan Integr Dec. 2019;5(3):126–33. https://doi.org/10.33172/jp.v5i3.596.

[37] Appelo J. Management 3.0: Leading Agile Developers, Developing Agile Leaders. Pearson Education; 2011.

[38] Hernández G, Martínez Á, Jiménez R, Jiménez F. Productivity metrics for agile software development teams: a systematic review. TecnoLógicas Dec. 2019;22: 63–81. https://doi.org/10.22430/22565337.1510.

[39] Wannes A, Ghannouchi SA. KPI-based approach for business process improvement. Procedia Comput Sci 2019;164:265–70. https://doi.org/10.1016/j.procs.2019.12.182.

[40] Valencia FD, Vega JP, Barrientos A. Agile maturity model for banking sector companies. presented at the Décima Segunda Conferencia Iberoamericana de Complejidad, Informática y Cibernética. Conferencia Virtual; Mar. 2022. p. 201–5. https://doi.org/10.54808/CICIC2022.01.201.

[41] Nonaka I, Kodama M, Hirose A, Kohlbacher F. Dynamic fractal organizations for promoting knowledge-based transformation – a new paradigm for organizational theory. Eur Manag J Feb. 2014;32(1):137–46. https://doi.org/10.1016/j.emj.2013.02.003.

[42] unFIX, 'The unFIX model', unFIX. Accessed: October. 6, 2025. [Online]. Available: https://unfix.com/blog/the-unfix-model.

[43] J. Appelo, 'Startup, scaleup, screwup: 42 tools to accelerate lean and agile business growth', Goodreads. Accessed: October. 6, 2025. [Online]. Available: https://www.goodreads.com/book/show/41755006-startup-scaleup-screwup.

[44] unFIX, 'Scaling unFIX', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/scaling-unfix.

[45] unFIX, 'Self-similar scaling', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/self-similar-scaling.

[46] unFIX, 'Value Streams', unFIXAccessed: October. 10, 2025. [Online]. Available: https://unfix.com/value-streams.

[47] unFIX, 'Role Attributes', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/role-attributes.

[48] unFIX, 'Queues and Backlogs', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/queues-and-backlogs.

[49] unFIX, 'Dependency breakers', unFIX. Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/dependency-breakers.

[50] unFIX, 'Forum Types', unFIX. Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/forum-types.

[51] unFIX, 'Collaboration Moments', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/collaboration-moments.

[52] unFIX, 'Delegation levels', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/delegation-levels.

[53] unFIX, 'Bets and objectives', unFIX.Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/bets-and-objectives.

[54] unFIX, 'Goal-setting aspects', unFIX. Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/goal-setting-aspects.

[55] unFIX, 'Measures and metrics', unFIX. Accessed: October. 10, 2025. [Online]. Available: https://unfix.com/measures-and-metrics.

[56] Kitchenham B. Procedures for performing systematic reviews [Online]. Available: https://www.semanticscholar.org/paper/Procedures-for-Performing-Systematic-Reviews-Kitchenham/29890a936639862f45cb9a987dd599dce9759bf5. [Accessed 3 December 2024].

[57] García Holgado A, Marcos Pablos S, García Peñalvo FJ. Guidelines for performing systematic research projects reviews. Int J Interact Multimed Artif Intell June 2020;6(2):136–44. https://doi.org/10.9781/ijimai.2020.05.005.

[58] García-Peñalvo FJ. Development of robust state-of-the-art reviews: systematic literature reviews. Educ Knowl Soc EKS Apr. 2022;23:e28600. https://doi.org/10.14201/eks.28600.

[59] Stallman R. Free software for a free society [Online]. Available: http://archive.org/details/softwarelibreparaunasociedadlibre. [Accessed 29 August 2025].

[60] unFIX, 'unFIX - versatile organization design', unFIX. Accessed: August. 29, 2025. [Online]. Available: https://unfix.com.

[61] Castiñeiras JR. Proyect freedone. presented at the XV Concurso de Trabajos Estudiantiles (EST 2012) (XLI JAIIO, La Plata, 27 al 31 de agosto de 2012). 2012 [Online]. Available: http://sedici.unlp.edu.ar/handle/10915/124917. [Accessed 29 August 2025].

[62] Arya A, Chang O, Metzman J, Serebryany K, Liu D. OSS-Fuzz. Shell 2025. Aug. 29, https://github.com/google/oss-fuzz. [Accessed 29 August 2025].

[63] Github. Project google/oss-rebuild. Go. Google; Aug. 29, 2025 [Online]. Available: https://github.com/google/oss-rebuild. [Accessed 29 August 2025].

[64] Sánchez Ruiz JM, Olivero González MÁ, Domínguez Mayo FJ, Benavides Cuevas DF. A collaborative comparison of performance in free software projects [Online]. Available: https://hdl.handle.net/11441/159714. [Accessed 29 August 2025].

[65] OpenLogic. State of open source report' [Online]. Available: https://www.openlogic.com/resources/state-of-open-source-report. [Accessed 3 December 2024].

[66] Karima Moumane, Ali Idri, Khalid Nafil. An empirical evaluation of Mobile software usability using ISO 9126 and QoS DiffServ model. In: Frontiers in artificial intelligence and applications. IOS Press; 2016. https://doi.org/10.3233/978-1-61499-674-3-177.

[67] Business A. KFC UK&I case study [Online]. Available: https://www.agilebusiness.org/resource/casestudy-kfc-uk-i-case-study.html. [Accessed 29 August 2025].

[68] Whittemore R, Chao A, Jang M, Minges KE, Park C. Methods for knowledge synthesis: an overview. Heart Lung 2014;43(5):453–61. https://doi.org/10.1016/j.hrtlng.2014.05.014. Sept.

[69] Mucambe B, Tereso A, Faria J, Mateus T. Large-scale agile frameworks. Dealing with interdependences', SCOPUS; Apr. 2019.

[70] Kalenda M, Hyna P, Rossi B. Scaling agile in large organizations: practices, challenges, and success factors. J Softw Evol Process Oct. 2018;30(10):e1954. https://doi.org/10.1002/smr.1954.

[71] Gandomani TJ, Zulzalil H, Ghani AAA, Sultan ABM. A systematic literature review on relationship between agile methods and open source software development methodology. 2013. https://doi.org/10.48550/ARXIV.1302.2748.

[72] Putta A, Paasivaara M, Lassenius C. Adopting scaled agile framework (SAFe): a multivocal literature review. In: *Proceedings of the 19th international conference on agile software development: companion*, Porto Portugal. ACM; May 2018. p. 1–4. https://doi.org/10.1145/3234152.3234164.

[73] Saeeda H, Khalid H, Ahmed M, Sameer A, Arif F. Systematic literature review of agile scalability for large scale projects. Int J Adv Comput Sci Appl 2015;6(9). https://doi.org/10.14569/IJACSA.2015.060908.

[74] Dikert K, Paasivaara M, Lassenius C. Challenges and success factors for large-scale agile transformations: a systematic literature review. J Syst Software Sept. 2016; 119:87–108. https://doi.org/10.1016/j.jss.2016.06.013.

[75] Abrar MF, et al. Motivators for large-scale agile adoption from management perspective: a systematic literature review. IEEE Access 2019;7:22660–74. https://doi.org/10.1109/ACCESS.2019.2896212.

[76] Hauge Ø, Ayala C, Conradi R. Adoption of open source software in software-intensive organizations – a systematic literature review. Inf Software Technol Nov. 2010;52(11):1133–54. https://doi.org/10.1016/j.infsof.2010.05.008.

[77] Sanchez VR, Ayuso PN, Galindo JA, Benavides D. Open source adoption factors—A systematic literature review. IEEE Access 2020;8:94594–609. https://doi.org/10.1109/ACCESS.2020.2993248.

[78] Franco-Bedoya O, Ameller D, Costal D, Franch X. Open source software ecosystems: a systematic mapping. Inf Software Technol Nov. 2017;91:160–85. https://doi.org/10.1016/j.infsof.2017.07.007.

[79] Edison H, Wang X, Conboy K. Comparing methods for large-scale agile software development: a systematic literature review. IEEE Trans Software Eng Aug. 2022; 48(8):2709–31. https://doi.org/10.1109/TSE.2021.3069039.

[80] García-Peñalvo FJ. Systematic reviews and literature mapping [Online]. Available: http://repositorio.grial.eu/handle/grial/1551. [Accessed 3 December 2024].

[81] García-Peñalvo FJ. Systematic literature mappings, systematic literature reviews, and benchmarking of training programs. '; 2017.

[82] Brereton P, Kitchenham BA, Budgen D, Turner M, Khalil M. Lessons from applying the systematic literature review process within the software engineering domain. J Syst Software Apr. 2007;80(4):571–83. https://doi.org/10.1016/j.jss.2006.07.009.

[83] Page MJ, et al. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. BMJ Mar. 2021:n71. https://doi.org/10.1136/bmj.n71.

[84] Turetken O, Stojanov I, Trienekens JJM. Assessing the adoption level of scaled agile development: a maturity model for scaled agile framework. J Softw Evol Process June 2017;29(6):e1796. https://doi.org/10.1002/smr.1796.

[85] Rehman AU, Nawaz A, Ali MT, Abbas M. A comparative study of agile methods, testing challenges, solutions & tool support. In: 2020 14th international conference on open source systems and technologies (ICOSST). Lahore, Pakistan: IEEE; Dec. 2020. p. 1–5. https://doi.org/10.1109/ICOSST51357.2020.9332965.

[86] Galán OAA, Luis J, Ferreira H, A G, Leonardo J. Proposal of a sustainable agile model for software development. Int J Adv Comput Sci Appl 2020;11(1). https://doi.org/10.14569/IJACSA.2020.0110105.

[87] Harzl A. Can FOSS projects benefit from integrating kanban: a case study. J Internet Serv Appl Dec. 2017;8(1):7. https://doi.org/10.1186/s13174-017-0058-z.

[88] Okoli C, Carillo K. The best of adaptive and predictive methodologies: open source software development, a balance between agility and discipline. Int J Inf Technol Manag 2012;11(1/2):153. https://doi.org/10.1504/IJITM.2012.044071.

[89] Gary K, et al. Agile methods for open source safety-critical software. Software Pract Ex Aug. 2011;41(9):945–62. https://doi.org/10.1002/spe.1075.

[90] Tsirakidis P, Kobler F, Krcmar H. Identification of success and failure factors of two agile software development teams in an open source Organization. In: *2009 fourth IEEE international conference on global software engineering*, Limerick, Ireland. IEEE; July 2009. p. 295–6. https://doi.org/10.1109/ICGSE.2009.42.

[91] Düring B. 'Sprint driven development: agile methodologies in a distributed open source Project (PyPy)', in extreme programming and agile processes in software engineering. In: Abrahamsson P, Marchesi M, Succi G, editors. Lecture notes in computer science, vol.4044. Berlin, Heidelberg: Springer Berlin Heidelberg; 2006. p. 191–5. https://doi.org/10.1007/11774129_22. 4044.

[92] Conboy K, Carroll N. Implementing large-scale agile frameworks: challenges and recommendations. IEEE Softw Mar. 2019;36(2):44–50. https://doi.org/10.1109/MS.2018.2884865.

[93] Dingsoeyr T, Falessi D, Power K. Agile development at scale: the next frontier. IEEE Softw Mar. 2019;36(2):30–8. https://doi.org/10.1109/MS.2018.2884884.

[94] Brataas G, Hanssen GK, Herbst N, Van Hoorn A. Agile scalability engineering: the ScrumScale method. IEEE Softw Sept. 2020;37(5):77–84. https://doi.org/10.1109/MS.2019.2923184.

[95] Jovanovic M, Mesquida A-L, Mas A, Colomo-Palacios R. Agile transition and adoption frameworks, issues and factors: a systematic mapping. IEEE Access 2020;8:15711–35. https://doi.org/10.1109/ACCESS.2020.2967839.

[96] Butler S, et al. On company contributions to community open source software projects. IEEE Trans Software Eng July 2021;47(7):1381–401. https://doi.org/10.1109/TSE.2019.2919305.

[97] Gonzalez-Barahona JM, Izquierdo-Cortazar D, Maffulli S, Robles G. Understanding how companies interact with free software communities. IEEE Softw Sept. 2013;30(5):38–45. https://doi.org/10.1109/MS.2013.95.

[98] Salikhov D, Succi G, Tormasov A. An empirical analysis of success factors in the adaption of the scaled agile framework – first outcomes from an empirical study 2020. https://doi.org/10.48550/ARXIV.2012.11144. *arXiv*.

[99] Collins W. Understanding scaled agile framework coordination methods for reducing failure rates [Online]. Available: https://www.proquest.com/docview/2450166180?fromunauthdoc=true&sourcetype=Dissertations%20&%20Theses. [Accessed 3 December 2024].

[100] Carilli JF. The perceived effectiveness of the scaled agile framework® in software development organizations', phd. USA: Indiana State University; 2021.

[101] Srinivas Sandur Madhu M. Qualitative study of scaled agile framework: an organizational approach. Capella University; 2020.

[102] Stojanov I. Scaling agile using scaled agile framework [Online]. Available: https://www.semanticscholar.org/paper/Scaling-agile-using-scaled-agile-framework-Stojanov/17c05c6c074e2b34fcee7cf0e48ef14008d26644. [Accessed 3 December 2024].

[103] Send H, Hölzner H, Hassan M. Community interaction in open business models: how IoT companies can handle community-generated innovation. Des Sci 2023;9:e5. https://doi.org/10.1017/dsj.2023.3.

[104] Caballero Rúa EH. 'Marco de trabajo para adaptar las metodologías ágiles e implantarlas a nivel organizacional'. Universidad Politécnica de Madrid; 2015. https://doi.org/10.20868/UPM.thesis.39391. PhD Thesis.

[105] Rozemblum C, Unzurrunzaga C, Banzato G, Pucacco C. Editorial quality and scientific quality in the parameters for inclusion of scientific journals in open access and commercial databases. Palabra Clave Apr. 2015;4(2):64–80.

[106] Torres-Salinas D, Ruiz-Pérez R, Delgado-López-Cózar E. *Google Scholar* como herramienta para la evaluación científica. El Prof Inf Sept. 2009;18(5):501–10. https://doi.org/10.3145/epi.2009.sep.03.

[107] Merchán-Narváez NJ, Palma-Peralta EE, Poma-Japón DX. Comparison of agile methodologies for software development. MQRInvestigar Mar. 2024;8(1):5052–74. https://doi.org/10.56048/MQR20225.8.1.2024.5052-5074.

[108] E. Ntt Data, '85% of companies in LATAM are joining the new trend of Agile methodologies', Talento y Empresa. Accessed: Sept. 02, 2025. [Online]. Available: https://www.talentoyempresa.com.ar/article/el-85-de-las-companias-en-latam-se-suman-a-la-nueva-tendencia-de-metodologias-agiles.

[109] Bedoya F. Open source software ecosystems quality analysis from data sources [Online]. Available: https://core.ac.uk/reader/157811040. [Accessed 1 September 2025].

[110] Digital.ai. State of agile [Online]. Available: https://stateofagile.com/. [Accessed 2 September 2025].

[111] Scrum Network. Scrum Network - scrum Original en Colombia [Online]. Available: https://www.scrumnetwork.com. [Accessed 2 September 2025].

[112] Schwartz F. Scrum Las Estrategias del Juego: es Póker, No Ajedrez de. 2020.

[113] Knaster R, Leffingwell D. SAFe 5.0 distilled: achieving business agility with the scaled agile framework. Addison-Wesley Professional; 2020.

[114] Alqudah M, Razali R. A review of scaling agile methods in large software development. Int J Adv Sci Eng Inf Technol Dec. 2016;6(6):828. https://doi.org/10.18517/ijaseit.6.6.1374.

[115] Morales-Carrillo J, Cedeño-Valarezo L. Software development methodologies and their scope of application: a systematic review [Online]. Available: https://www.proquest.com/openview/937c0ffcf92bf2c2761d76165f34090e/1?pq-origsite=gscholar&cbl=1006393. [Accessed 1 September 2025].

[116] Zanotti A, Vélez JG. Floss development and peer governance: the case of the gnome desktop environment. Int J Innov Dec. 2020;8(3):438–65. https://doi.org/10.5585/iji.v8i3.17114.

[117] Vélez JG, Zanotti A. Production in FLOSS communities: companies, foundations, and governance. Estud Sociol 2020. Sept. 03, 2025. [Online]. Available: https://ri.conicet.gov.ar/handle/11336/146566.

[118] Wenger E. Communities of practice: learning, meaning, and identity. Cambridge: Cambridge University Press; 1998.

[119] Giraldo Marín LM, Atehortúa Correa LF. Communities of practice: a strategy for the democratization of knowledge in organizations, a reflection. Rev Ing Univ Medellín Jan. 2010;9(16):141–50.

[120] Castrillón G, Alfonso M. Communities of practice in organizations. Desarro Gerenc 2020;12(1):1–22.

[121] Juncos I, Borrastero C. In: Morero H, Motta J, editors. Large software corporations and FLOSS: cooperating in open environments as a competitive strategy. Hernán Morero y Jorge Motta; 2020. p. 77–111. Sept. 03, 2025. [Online]. Available: http://estudiosociologicos.org/portal/la-economia-del-software-libre-y-open-source-multinacionales-pymes-y-comunidades/.

[122] K. Fogel, 'Produce open source software', UniversiTAD. Accessed: September. 3, 2025. [Online]. Available: https://universitad.com/computacion/producir-software-de-codigo-abierto-1-edicion-karl-fogel/.

[123] Kahneman D, Tversky A. Subjective probability: a judgment of representativeness. Cogn Psychol July 1972;3(3):430–54. https://doi.org/10.1016/0010-0285(72)90016-3.

[124] Wohlin C, Runeson P, Höst M, Ohlsson MC, Regnell B, Wesslén A. Experimentation in software engineering. Berlin, Heidelberg: Springer Berlin Heidelberg; 2012. https://doi.org/10.1007/978-3-642-29044-2.

[125] Umbrella review'.[Online]. Available: https://doi.org/10.5281/zenodo.17338512.

[126] Systematic review'.[Online]. Available: https://doi.org/10.5281/zenodo.17338592.

[127] Voigt K-I, Buliga O, Michl K. Passion for music: the case of spotify. In: Voigt K-I, Buliga O, Michl K, editors. Business model pioneers: how innovators successfully implement new business models. Cham: Springer International Publishing; 2017. p. 143–55. https://doi.org/10.1007/978-3-319-38845-8_12.

[128] A. Eremina, 'Comparison of organizational structures : case Zappos', laturi oulu fi. Accessed:Sept. 03, 2025. [Online]. Available: https://oulurepo.oulu.fi/handle/10024/8782.