

Optimizing Agile System Implementation: Strategies for Enhancing Team Productivity and Project Outcomes in Software Development

Arifuzzaman*

Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

E-mail : aarifuzzaman@stu.kau.edu.sa

ORCID ID: <https://orcid.org/0009-0002-0993-7409>

*Corresponding Author

Rizwan Jameel Qureshi

Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia

E-mail : rmuhammd@kau.edu.sa

ORCID ID: <https://orcid.org/0000-0002-1664-0737>

Received: 11 February, 2025; Revised: 24 May, 2025; Accepted: 26 July, 2025; Published: 08 December, 2025

Abstract: Scrum and Kanban are some of the most common methodologies that are used in software development because of flexibility and focus on the team. Nevertheless, applying Agile at a project level and within large IT projects including workforce distributed across different areas, manages some remarkable difficulties, for example, coordination, communication, or resources. This paper examines ideas on how to improve the implementation of Agile system to increase the performance of the team and results of the projects. The study focuses on four key goals: proving Agile improvements in practice via pilot surveys, applying best-practice structures, such as defence-grade SAFe or LeSS at the scale, encouraging organizations-wide Agile mindset, and using collaboration and automation technologies when working in remote environments. These issues serve as the focus of this research with the intention of preserving Agile's principles of flexibility and practicability across various size and scale projects. It presents suggestions for further research and informs practitioners and organizations wishing to obtain the most out of Agile methodologies in real environments.

Index Terms: Agile Methodologies, Team Productivity, Scalable Frameworks, Distributed Teams, Project Outcomes

1. Introduction

This paper introduces the Hybrid Distributed Agile Framework (HDAF), a novel synthesis that goes beyond repackaging existing strategies. While it builds upon established approaches such as SAFe, LeSS, pilot studies, and communication tools, the originality lies in integrating them into a structured hybrid model tailored specifically for distributed Agile environments. Unlike prior work, this framework embeds decision rules for selecting scaling methods (SAFe vs. LeSS), incorporates empirical validation through pilot studies and surveys, and establishes a measurable process using normalized Agile Efficiency System (AES) metrics. By combining these elements with an iterative feedback loop, HDAF contributes a validated, actionable model that balances scalability with agility—providing practical novelty for organizations facing challenges in large-scale IT projects.

The design of the Hybrid Distributed Agile Framework (HDAF) is grounded in established models from software engineering (SE) and organizational behavior. Specifically, we draw on Boehm's Spiral Model for iterative risk-based development [1], Kotter's 8-Step Change Model for organizational transformation [2], and Lewin's Change Management Model [3]. These theoretical foundations provide rigor and ensure that HDAF not only addresses technical scalability but also organizational readiness and cultural change.

Scrum, Kanban, and other blended models have become globally popular as a suitable approach to dealing with today's diverse and rapidly evolving software development challenges. It enables teams to respond quickly to alteration in demand and provide ongoing enhancement to projects, which makes Agile the most suitable option for current software development projects. But as the features of projects become more intricate and scope gets bigger, especially in the distributed large scale IT environment, Agile has some real challenges. Tasks like organizing interdisciplinary

work, communication protocols, and resource management particularly exert pressure on the very fundamental notions of flexibility and responsiveness and as a result become slow and cumbersome.

While we have seen new tools woven into ITSM such as DevOps, Lean, and automation, the actual real-world outcomes of these enhancements are still largely unknown. Corporations are often challenged with achieving the Agile flexibility, bringing in the necessity of structure and governance to accommodate the large number of teams. Also, the social and technical changes that need to occur in organizations for Agile to work, more so when implemented in environments where people work remotely, or are distributed across geographical locations, make Agile's adoption even more challenging. This mandates empirical research and Hybrid solutions that would indeed prove out and enhance Agile for use, to cover various situations. This paper aims to address these challenges by focusing on four critical goals: For instance, empirically validation Agile enhancements carried out through small-scale pilot studies, integrating best and better-known practices within the SaFe and LeSS, agile training and organizational cultural change, and leveraging on over communication and automation tools in distributed group formation. Through the systematic investigation of these areas, this study aims at offering recommendations to the relevant practitioners and organizations to enhance Agile methodologies, so that they will continually serve as a viable solution for developing high-tech, large-scale software projects.

Section 2 covers related work followed by the problem definition in section 3. The proposed hybrid framework is presented in section 4. Section 5 illustrates the validation of the proposed framework.

2. Literature Review

Specifically, this review aims to enhance the conventional Agile practices for high-technology software development for better performance and flexibility of high technology software development in today's rapidly growing tech environment. They discovered that practices like Scrum and Kanban may be inadequate in large projects and proposed ways to improve these practices including integrating DevOps to improve collaboration Lean to minimize waste, the use of configurable Agile that suits projects' needs and encourage constant feedback. Nonetheless, the study needs long-term testing to support these changes, does not address cultural integration into the system, and can introduce some complexity, mainly for a small number of teams. The last issue is maintaining agility flexibility when addressing Agile, Lean, and DevOps currents in high-speed contexts [4].

The study aims at identifying effort estimation and on-time delivery problematics in large-scale Agile software development, specifically at schedule variance. It defines problems such as project delivery delay, and budget increase, which arise from poor estimations of effort during its planning phase. The authors merge survey data with data gathered from software repositories and to model and explain more intricate factors such as timing of requirements refinement, task dependencies and organizational politics that drive project times/maps. They present a set of guidelines as intervention points to assist practitioners mitigate risks associated with delays and do this while admitting to some weaknesses which include those arising from survey bias, low external validity, rudimentary representation of interactions and the possibility of automated risk indicators struggling to predict delays [5].

Determining the way Agile methodologies have been put into practice and the impact they have on projects; this research aims at identifying successful factors in Agile software development. Agile project results are sometimes inconsistent and Binboga and Gumussoy discuss this issue and present the Agile Software Project Success Model from the literature and expert respondents. Using statistical analysis of the 596 practitioners' data, they confirm the model and identify the customer and Agile process factors as the most significant determinants of project success. However, there are the following potential limitations: the data were obtained using questionnaires; it is difficult to generalize the results to other organizations; proposed hypotheses are not always supported; and, finally, the study focuses on the customer and process perspectives that can hide such significant factors as team or technology [6].

As a result, the study's core area of interest is the development of an additional method for increasing Agile team performance, namely through the adoption of the Kanban system in Agile development. Education of Psarov and Druzhinin gives an idea about how they address issues such as improvement of lengthy business processes, which prevent teams from providing a constant stream of valuable work. CFDs and flow metrics are also suggested to recognize the work status, put restriction on WIPs, and pursue constant enhancement. However, potential issues belong to the following: The problem can be narrowed down just to the metrics and disregard the team aspects, some metrics may be misinterpreted, such as CFDs, the given strategies may not be incorporated into different teams especially large and distributed ones [7].

The key subject for the study is Scrum in the context of Agile product development and their application, usage, and development profile in remote and distributed teams. In their paper, Ekechi et al. analyze the challenges of implementing Scrum in different contexts and consider the problem of resistance to its application. They have recommended developing practices regarding scrum that would respond to the requirements of the project, the promotion of culture adaptation, and addressing the challenges of distributed teams. First, the study focuses on discussing how Scrum is trying to develop: the integration with DevOps functions and the tendencies to expand the framework's application to other industries. Some of the important suggestions are training, customization, and culture integration that has to be continued over a period of time. However, as the study is highly general, general conclusions can be derived, and further issues of implementation can be left unnoticed. Further, it does not offer a set of guidelines

on how an organization can analyze the effectiveness of the Scrum adaptations it implements and may overemphasize the ease of applying changes across an organization [8].

The questions of interest concern data management concern in Agile software development, including such aspects as data integration, data quality changes, and real-time analysis. These challenges are reflected by Fawzy et al. [9], who employs a systematic literature review complemented by an e-survey, completed by 32 industry practitioners. They suggest that steps should be taken to define good data management practices, share knowledge and educate individuals about these good data management practices, and to develop new approaches for better agility, data integration, and automation. However, the results cannot be generalized because the number of participants is small, and relations in the work might not cover real-life conditions like legacy systems and resource scarcity. Further, the proposed solutions may not address problems that change quickly, and Agile environments often have quickly changed data requirements [9].

Table 1. Limitations and restrictions on the related work

Title Of the Paper	Limitations
“Optimizing agile project management methodologies in high-tech software development” [4]	The study calls for further testing to validate these changes, may overlook the cultural shifts needed for integration, and could add complexity, especially for smaller teams. Balancing Agile, Lean, and DevOps methodologies in fast-paced environments also remains a challenge.
“Factors Affecting On-Time Delivery in Large-Scale Agile Software Development” [5]	They propose a framework to help practitioners manage risks related to delays but acknowledge limitations, including potential survey bias, limited generalizability, oversimplification of complex interactions, and the challenges automated tools may face in predicting schedule overruns.
“Factors Affecting Agile Software Project Success” [6]	Potential limitations include reliance on self-reported data, limited generalizability, unsupported hypotheses, and an emphasis on customer and process factors that may overlook other critical aspects like team dynamics or technology.
“Enhancing Agile Team Productivity with Metrics” [7]	Potential issues include an overemphasis on metrics, which may overlook team dynamics, the risk of misinterpreting metrics like CFDs, and the difficulty of applying these strategies across different teams, especially larger or distributed ones.
“Enhancing agile product development with scrum methodologies: A detailed exploration of implementation practices and benefits” [8]	The study's broad scope may lead to vague insights, and it may not fully address real-world implementation challenges. Additionally, it lacks a concrete framework for measuring the success of Scrum adaptations and may underestimate the complexity of organizational change management.
“Exploring Data Management Challenges and Solutions in Agile Software Development: A Literature Review and Practitioner Survey” [9]	The small survey sample limits the findings' generalizability, and the study may not fully address real-world complexities, such as legacy systems and resource constraints. Additionally, the proposed solutions may not account for the fast-paced, evolving nature of Agile environments where data needs shift quickly.
“An Insight into the Capabilities of Professionals and Teams in Agile Software Development: An Update of the Systematic Literature Review” [10]	The study leans more on technical metrics than social or innovative aspects, which may limit its relevance to human dynamics. Additionally, the lack of established prediction models and reliance on existing literature may restrict practical application, and the focus on metrics could oversimplify complex human behaviors.
“An effective framework to improve the managerial activities in global software development” [11]	The framework may lack detail on adapting to different organizational contexts, overlook cultural and time zone differences, and focus too much on high-level issues without addressing specific operational tools. Additionally, the framework may struggle to keep pace with evolving technologies and practices in GSD.
“Agile Project Planning and Execution in Large-Scale IT Projects” [12]	The research may oversimplify the challenges of scaling Agile for remote teams, underplay coordination difficulties, and overlook the organizational changes needed for Agile adoption. Additionally, it may not fully emphasize the need for balancing Agile flexibility with structured governance in large IT projects.
“Analysis of Strategies for the Integration of Security Practices in Agile Software Development: A Sustainable SME Approach” [13]	Only 20.2% of studies focus on SMEs, limiting relevant insights. Additionally, SMEs may struggle to implement robust security due to resource constraints, and the study may not fully address balancing security with the speed of Agile development. Emerging practices could also be overlooked by relying heavily on existing literature

The present review paper concentrates on the assessment and prognosis of human competencies in relation to ASD; specifically, the effects of individual and team on effectiveness and success. More recently, Cunha et al., have published an update of what was previously a review of the knowledge on this topic, and while listing personality dimensions, they have added new elements like creativity and adaptability to this approach to the composition of successful teams. Mentally, they propose the use of these traits for enhancing the formation of teams and predictions hence developing ways on using these traits in a real-life capability base in the later studies. Still, it lacks comprehensive focus on social and innovative aspects, so for human-centered conceptions, the study more relies on technical parameters. Finally, since there is no well-defined set of parameters for predicting unobservant behavior, there is an overreliance on prior work and certain aspects of the paper may be limited to practical use [10].

The study therefore investigates project management issues in GSD environments to explain why distributed teams experience coordination, communication, and control problems that affect software quality. Considering these challenges, Siddique et al. have provided a solution to the crisis recommending strategies and a framework toward enhancing collaboration and project outcomes in GSD settings. Nevertheless, the framework could be weak in explaining how different organizational settings may be managed, there could be no sufficient attention paid to differences in cultural and time zone environments, and there could be over-emphasis on solving general strategic

problems of change, while substantive attention is paid to concrete operational instruments. However, the framework might not easily be adaptable to metamorphosing technologies and practices in GSD [11].

The study is dedicated to applying the Agile approach in large-scale IT projects, the issues with conventional methodologies in the complicated and rapidly evolving context. Challenges are resource management, collaboration impedance, risk management, and Agile at scale with Distributed teams and are described by Avancha et al. They suggest that such frameworks as SAFe present large projects as a set of components, which enables iterative planning and integration of efforts of various specialized teams. However, the research can reduce the complexity of scaling Agile for a remote team, underestimate the issues in coordination, and ignore organizational implications of Agile adoption. Furthermore, it also may neglect some of the aspects in that it does not place equal importance between agility and governance formality in large IT projects [12].

This research considers the question of how small and medium-sized firms (SMEs) can adopt Agile methodologies for the integration of security into SDLC. A study by Valdés-Rodríguez et al. points to the difficulty of implementing security in Agile environments due to the higher pace and vulnerability to cyber threats a situation that burdens SMEs. Subsequently, using a literature-based approach, the authors provided various measures for SMEs to protect and maintain their applications in the future. Yet, concerning the SMEs, only 20.2% of the research concentrates on them, as it is indicated. Further, SMEs may experience limitations in their ability to integrate a strong security infrastructure because of the lack of resources, and the research may not meet exploring security and efficiency and the Agile development speed optimally. New practices could also not be considered by relying mainly on prior literature and research [13].

While the reviewed studies provide valuable insights into Agile methodologies, several limitations remain that directly inform the focus of this research. For example, studies on Agile-DevOps integration highlight the need for long-term empirical validation but lack cultural integration strategies [14]. Research on delivery performance identifies estimation and coordination issues, yet offers limited solutions for distributed environments. Success models emphasize customer and process factors but often overlook cultural readiness and tooling [15]. Similarly, productivity metrics and Scrum adaptations provide useful guidance but are either too narrow or too general, making them difficult to apply at scale. Finally, data management and SME-focused Agile studies expose critical gaps in automation, scalability, and contextual adaptation.

These gaps directly connect to the research problem of scaling Agile in distributed, large-scale projects while preserving flexibility. They also justify the need for the Hybrid Distributed Agile Framework (HDAF), which explicitly addresses: (1) empirical validation of Agile enhancements, (2) structured adoption of scaling frameworks such as SAFe and LeSS, (3) organizational training and cultural readiness, and (4) integration of automation and communication tools. Thus, the proposed framework builds upon and extends prior work by unifying technical, cultural, and governance dimensions into a coherent, empirically validated model.

3. Problem Definition

More specifically, Scrum and Kanban methodologies are amongst the most common trends because they are flexible, can be used for quick shifts, and imply cooperation between team members. Nevertheless, the authorities admitted that there is still much scope in performing these advances in the live environment, even with the extension of the advanced integrations of DevOps, Lean, and hybrids and techniques. When dealing with complicated project environments, these improvements do not appear to offer the anticipated benefits without proper testing with real-world examples, and this seems to create confusion among organizations about their effectiveness [4]. However, when Agile is adopted at a large scale when used in large IT projects, especially involving distributed teams, this problem is exacerbated. Of course, Agile at scale brings into question new challenges in such contexts as the organization's need to coordinate its cross-functional teams, communication, and resource management, among others. These factors take a lot of Agile values, especially flexibility away, which can result in inefficiency, delays, and even conflict [12]. While frameworks such as SAFe have been developed to address scaling problems, the behavior of these improvements in reality still needs scientific study and fine tuning at best [12]. A research question is narrated to address the issues raised in the existing studies [4,12].

- How can the Hybrid Distributed Agile Framework (HDAF) address the challenges of scaling Agile in distributed IT projects by combining validated Agile enhancements, scaled frameworks, cultural readiness, and automation tools?

4. The Proposed Framework

Fig.1 presents the Hybrid Distributed Agile Framework (HDAF), which integrates empirical validation, cultural readiness, framework selection, structured implementation, measurement, and continuous feedback into one unified cycle. The framework begins by recognizing the inputs, which represent the organizational context—team size, distribution across geographies, technical complexity, and cultural factors. These inputs are crucial because they set the baseline for

deciding whether existing Agile practices can scale or whether adaptations are required. Without this step, organizations risk applying a generic Agile model that does not match their real-world challenges.

The second stage involves pilot validation and cultural readiness. Here, improvements such as automation, DevOps integration, or Lean practices are tested in small-scale pilot projects to determine their effectiveness before organization-wide adoption. In parallel, organizations assess their cultural readiness by providing Agile training and encouraging shifts in mindset among developers, managers, and leadership. This combination of technical testing and cultural alignment reduces resistance to change, ensuring smoother transformation. It emphasizes the idea that Agile at scale is not merely a technical process but a social and organizational one, where culture and mindset often determine success more than tools. This stage resonates with Kotter's framework of creating a sense of urgency and building a guiding coalition [2], and also reflects Lewin's 3-stage model of change (unfreeze–change–refreeze) [3], ensuring cultural alignment with Agile adoption. Once validated, the framework moves to a decision point where the organization selects between SAFe (Scaled Agile Framework) and LeSS (Large-Scale Scrum). This decision is based on project scale, coordination needs, and governance requirements. SAFe provides a structured, governance-heavy approach ideal for enterprises handling multiple vendors, regulatory compliance, and highly complex programs. In contrast, LeSS offers a lightweight approach suitable for smaller or moderately distributed projects, focusing on flexibility, collaboration, and reduced overhead. Including this decision step ensures that scaling is tailored, avoiding the common pitfall of forcing a single model onto all contexts.

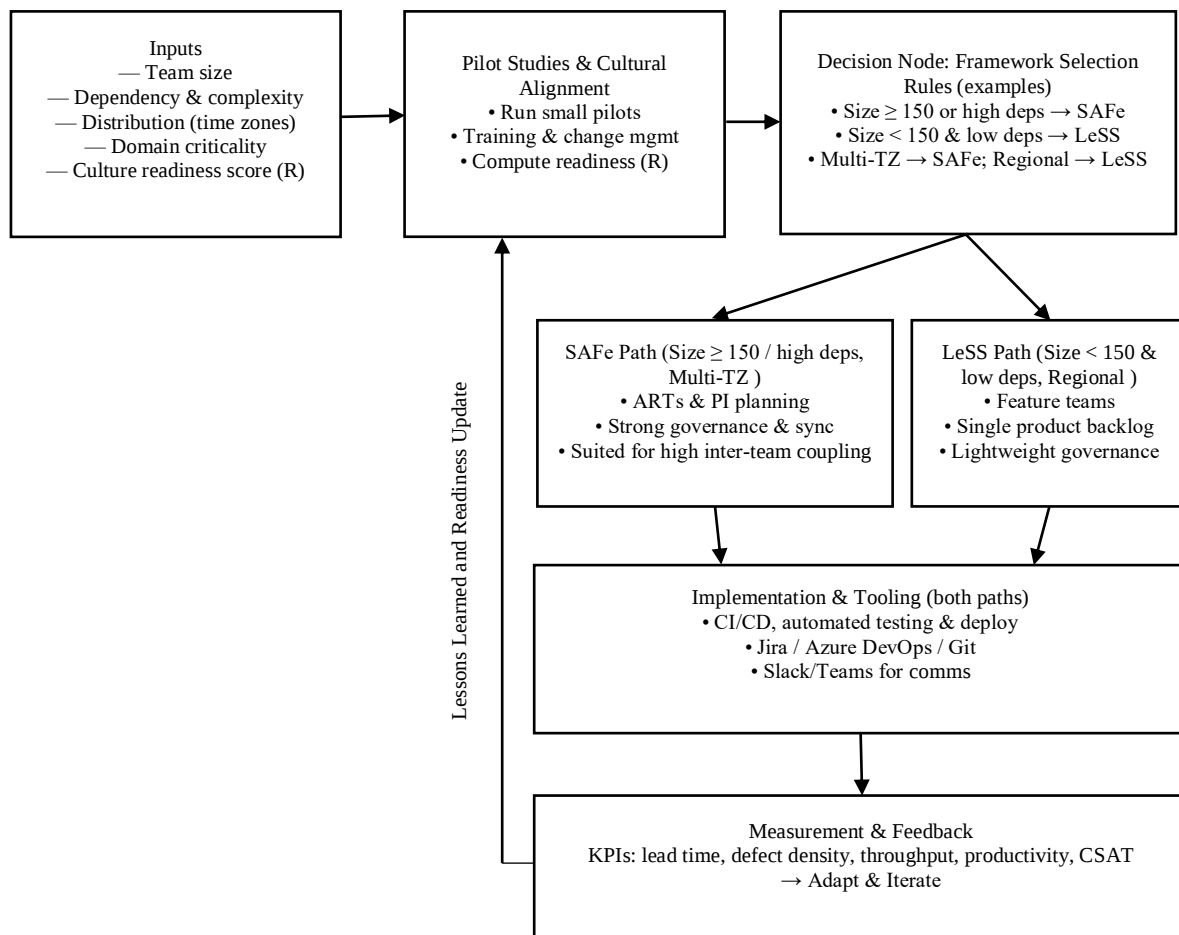


Fig. 1. Hybrid Distributed Agile Framework (HDAF)

The implementation and tooling stage converts strategy into practice. It emphasizes the use of collaboration tools (Slack, Jira, Microsoft Teams) and automation technologies (continuous integration, testing, and deployment pipelines) to support distributed teams. Coordination practices such as Scrum-of-Scrums and real-time dashboards maintain transparency across multiple teams working in different locations and time zones. This ensures that distributed teams operate as one unified system rather than fragmented silos, reducing the risks of delays, miscommunication, and duplication of work. Measurement is embedded into the framework through the Agile Efficiency System (AES) metrics: lead time, defect density, throughput, productivity, and customer satisfaction (CSAT). These metrics provide a holistic view, balancing delivery speed, quality, efficiency, and customer value. By normalizing results, organizations can compare performance across teams and projects, ensuring that scaling decisions are data-driven. This measurement stage also protects against the risk of Agile becoming symbolic rather than functional by holding teams accountable to tangible outcomes. This emphasis on measurement aligns with maturity models such as CMMI [16], ensuring that scaling follows structured improvement pathways rather than ad-hoc metrics.

Finally, the feedback loop ensures that HDAF operates as a living system. Results from measurements are fed back into the framework, leading to refinements in pilot strategies, cultural interventions, framework selection, and tooling practices. For example, if lead time remains high despite automation, backlog refinement processes may need to be restructured. If defect density increases, emphasis may shift to test automation or pair programming. This continuous feedback cycle ensures that Agile scaling does not stagnate but evolves with project complexity, team maturity, and technological advances.

In essence, fig. 1 demonstrates that HDAF is not a one-time framework but a dynamic, iterative model that balances structure and flexibility. It addresses the shortcomings of traditional Agile at scale by combining empirical evidence, cultural alignment, governance frameworks, technical tooling, and performance measurement into a coherent cycle. This makes it particularly relevant for distributed and large-scale IT projects, where Agile values must be preserved without sacrificing coordination, governance, or accountability. Fig. 2 shows workflow diagram of the proposed framework.

4.1 Empirical Validation through Pilot Studies and Small-Scale Implementations

It is recommended that organizational experiments should carry out slow pilot trials to experiment with improvements such as DevOps, Lean, and hybrid Agile in practical use laboratories. These small-scale implementations help the teams to gauge the improvements carried out in such enhancements under different circumstances. Information that can be gained from these pilots along with the performance measures would show these strengths or areas of weakness, so that through progressive improvement they can be implemented on a larger scale [4].

4.2 Adopting Scaled Agile Frameworks for Large Projects

For massive, dispersed IT undertakings, principles related to the particular scaling of Agile may be applied with the help of certain frameworks, including SAFe or LeSS. All these frameworks offer networked models in relation to the supply chain's many cross-functional teams including coordination, communication, and resource allocation. They ensure that Agile values such as flexibility is upheld while also making sure that organizational teams are on the same level. Use of Scrum-of-Scrums meetings and availability of effective real-time project management tools can improve the cross-distribution integration [12].

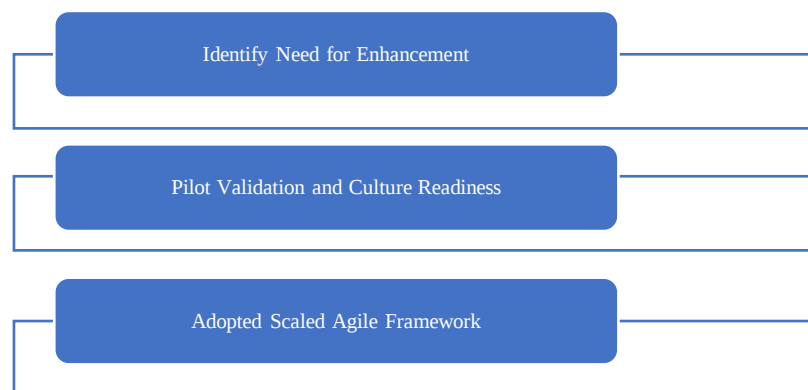


Fig. 2. Workflow Diagram of the Proposed Framework

4.3 Investing in Agile Training and Culture Transformation

Agile at scale is not just a question of technical solutions but also a problem of changing organizational culture. For training to be effective, most aspects of Agile and processes must be communicated and made understood by the developers, operations, and leadership among others. Such a cultural approach enhances teamwork and perpetuates organizational development. Also, the culture that fosters decentralized decision making can support the decentralized teams which remain independent even in huge contexts [12].

4.4 Leveraging Automation and Communication Tools

Organizations cannot afford to have individuals manually introducing changes, performing tests and deployments; therefore, the use of automation technologies for continuous Integration, testing and deployment should be encouraged to enhance the agility of the solutions that are being offered. Some of the collaboration solutions that might assist distant teams and be more transparent include Slack and Jira as well as Microsoft Teams. Students make sure that all the stakeholders of the project are informed and on the same page, so that they do not end up with major communication barriers, a common factor in large scaled distributed projects [12].

5. Validation of the Proposed Framework

One of the most critical steps in the proposed solution is therefore the validation of the proposed solution. facts that are necessary in any research focusing Students and their performance deserve to. In this paper the verification of the

presented idea with the help of the methods used in electronic survey. The rationale for using this method is that It's not too much time consuming and it means the respondent does not have to attend formal lectures most of the time to think and answer questions will suffice for credibility. So, the method of validating the proposed solution is going to be in fact, it entails accumulating a group of individuals who respond to an electronic questionnaire test. SE experts will get an electronic target audience survey via email and social media platforms like Facebook and Twitter. A Google form will be used to collect opinions. In this study, a questionnaire was employed to collect data. Four goals were created from the questions, and they were organized based on how relevant they were to the stated aims.

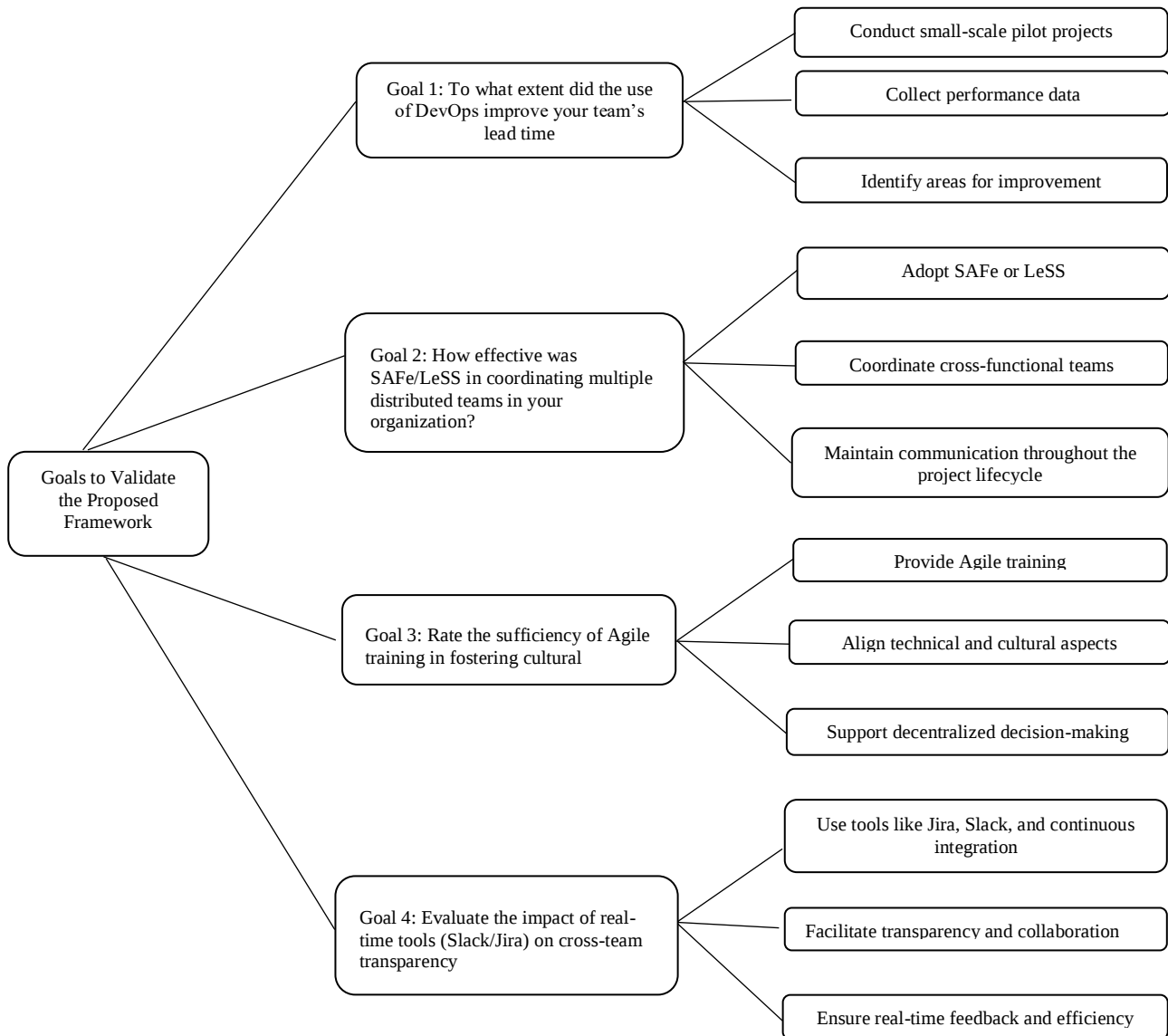


Fig. 3. The four goals to address the challenges in scaling Agile

The validation approach is consistent with Kitchenham's evidence-based SE paradigm [17], which advocates empirical grounding of proposed models to ensure credibility and applicability. A total of fifty respondents participated, with the sample size determined using a minimum recommended threshold for survey-based studies in empirical SE. Respondents included project managers (25%), Agile coaches (20%), developers (35%), and other roles (20%), with average professional experience of 6.2 years across SMEs, large enterprises, and distributed organizations. To reduce bias, purposive sampling ensured that only practitioners with ≥ 3 years of Agile experience participated, while stratified sampling balanced representation across industries and team sizes. Recruitment was carried out across professional networks (e.g., LinkedIn Agile communities, alumni groups, practitioner forums) to capture diverse organizational contexts and geographic regions. The objective of the questionnaire was to empirically validate the Hybrid Distributed Agile Framework (HDAF) by collecting practitioner insights across four goals as shown in fig. 3.

- **Goal 1 – Agile Enhancements:** Items assessed the impact of DevOps, Lean, and hybrid practices on performance metrics such as lead time, throughput, defect density, productivity, and customer satisfaction.

- **Goal 2 – Scaled Framework Adoption:** Items evaluated the effectiveness of SAFe and LeSS in coordinating distributed teams, managing dependencies, and reducing project delays.
- **Goal 3 – Training and Cultural Readiness:** Items explored the role of Agile training, leadership support, and organizational culture in facilitating Agile transformation.
- **Goal 4 – Automation and Communication Tools:** Items measured the influence of digital platforms (Slack, Jira, CI/CD pipelines) on collaboration, transparency, and efficiency.

By structuring the 32 items around these four goals, the questionnaire ensured construct validity and provided both quantitative and qualitative data for evaluating the feasibility, scalability, and organizational relevance of HDAF. A full list of items is included in Appendix A. Each set of items in the questionnaire corresponds to one of the four goals described in Section 4. An extensive statistical analysis was feasible since the questionnaires developed were standardized and also used to pass on information through participants. This approach made it possible to obtain as many data flows as possible and structure them for analysis. Frequency tables, maps, and charts were used to present the results and to emphasize some of the trends and patterns in the study to make the analysis easier to understand. These tools improved the depth of the analysis and provided better clarity in terms of the range of the results. Several sources of data were analyzed to check the validity and the relevance of the data collected to arrive at valid conclusions. By examining the quantitative validation findings, the current document provides a precise picture, supported by data outcomes, that enables researchers and decision-makers to make further decisions or conduct additional empirical studies.

5.1 Cumulative Outcomes of Goal 1

As with other Agile improvements (DevOps, Lean, combined approaches, etc.), the goal is to implement it on a small-scale using pilot projects. Such experiments will gather information on performance and define what aspects need optimization before utilizing the technology in high-tech. The findings for Goal 1 provided by the analysis are provided in Table 2. For Goal 1 on the questionnaire, the reaction is positive with 49.17% of respondents selecting ‘High’ and 32.92% of the respondents’ selecting ‘Very High’ to demonstrate a high level of optimism on the proposed solutions. 14.58% of participants offered Neutral response which we categorize as moderate perception, while 2.92% and 0.42% of participants offered ‘Low’ and ‘Very Low’ responses respectively. In aggregate, these findings underscore a high degree of confidence and approval for the strategies outlined here, with the large majority endorsing the strategies’ efficacy as shown in fig. 4.

Table 2. Detailed cumulative outcomes for goal 1

Q. No.	Very low	Low	Neutral	High	Very high
1	0	0	3	16	11
2	1	1	2	22	4
3	0	0	6	12	12
4	0	2	7	13	8
5	0	0	8	11	11
6	0	2	2	14	12
7	0	1	4	15	10
8	0	1	3	15	11
Total	1	7	35	118	79
Average	0.42	2.92	14.59	49.17	32.92

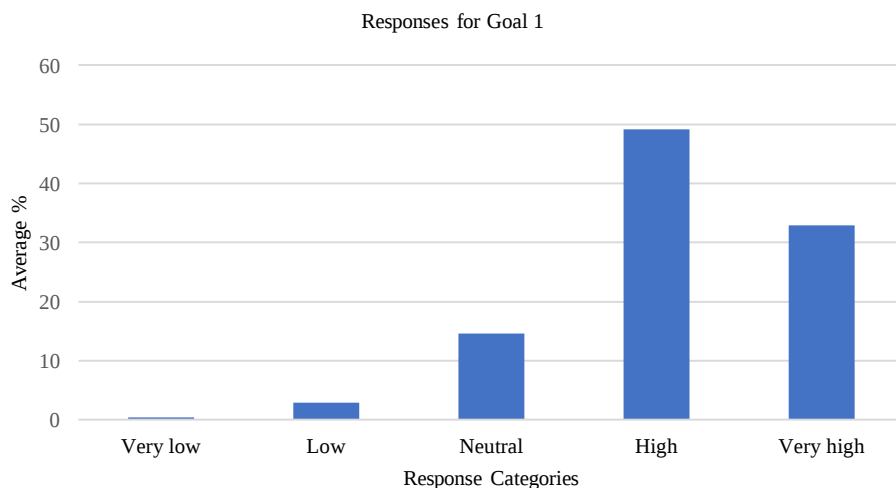


Fig. 4. Cumulative outcomes for goal 1 chart

5.2 Cumulative Analysis of Goal 2

For large and distributed IT initiatives, the target is to envisage and achieve the framework Unity SAFe or LeSS. This will help the organization to embrace Agile more effectively, connecting cross functional teams and communicating during project development. The findings of Goal 2 analysis are shown in Table 3.

Table 3. Detailed cumulative outcomes for goal 2

Q. No.	Very low	Low	Neutral	High	Very high
1	0	2	8	8	12
2	0	0	2	13	15
3	1	2	6	12	9
4	0	1	6	12	11
5	0	0	6	13	11
6	0	0	4	15	11
7	0	1	6	11	12
8	0	0	8	11	11
Total	1	6	46	95	92
Average	0.42	2.5	19.17	39.59	38.34

The findings regarding Goal 2 reveal a high level of support for the recommended strategies on average. Of all the respondents a greater percentage of 39.58% opted for “High” while 38.33% opted for “Very High”, indicating high level of optimism. The Imam received 19.17% neutral response, which showed moderate support or neutral response in addition 2.50% of the respondents opted for “Low” response while “Very Low” was 0.42% among the respondents. This work has indicated that the reception is generally positive as most of the participants showed confidence in the solutions as proposed in the research shown in fig. 5.

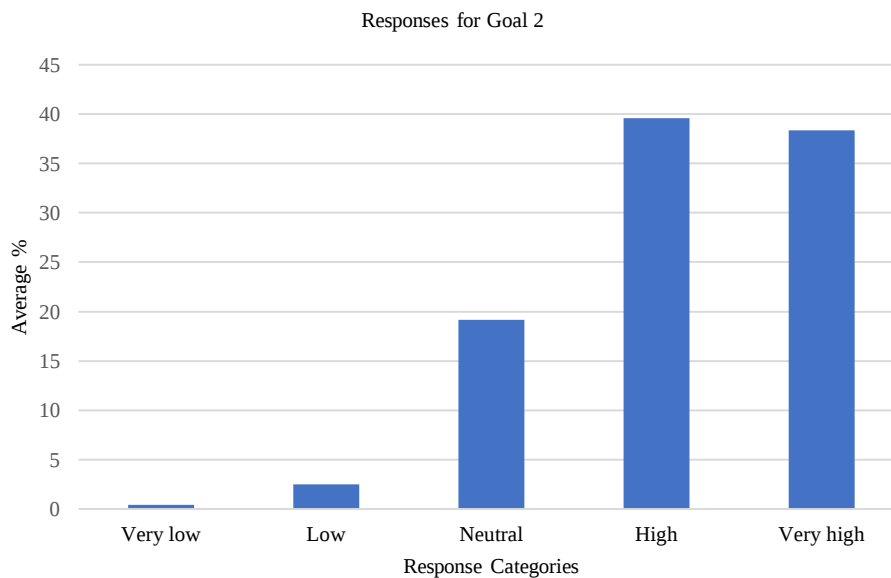


Fig. 5. Cumulative outcomes for goal 2 chart

5.3 Cumulative Analysis of Goal 3

The idea here is to educate a team that will be engaged in the Agile process to embrace Agile values. This training will teach how to deal with both the technical and cultural aspects needed for better cooperation, initiating decentralization of decision-making and engaging in constant improvement within the team. The findings of Goal 3 for the analysis results are highlighted in the tabular form in Table 4.

In considering the results on Goal 3 the levels of support for the proposed strategies are high. About the potential of business growth, 43.33% of respondents chose “High” option and 37.08% chose “Very High,” so it is possible to conclude about the subject’s high level of optimism. The percentage of moderate views was 15.42% of the neutral response type. The other 3.75% were of Low response type while 0.42% were of very low response type. From these conclusions one can state that the majority exclaimed the proposed solutions will be effective as shown in fig. 6.

Table 4. Detailed cumulative outcomes for goal 3

Q. No.	Very low	Low	Neutral	High	Very high
1	0	0	5	15	10
2	0	1	2	12	15
3	0	1	7	9	13
4	1	1	6	13	9
5	0	1	5	16	8
6	0	3	3	13	11
7	0	2	3	14	11
8	0	0	6	12	12
Total	1	9	37	104	89
Average	0.42	3.75	15.42	43.34	37.08

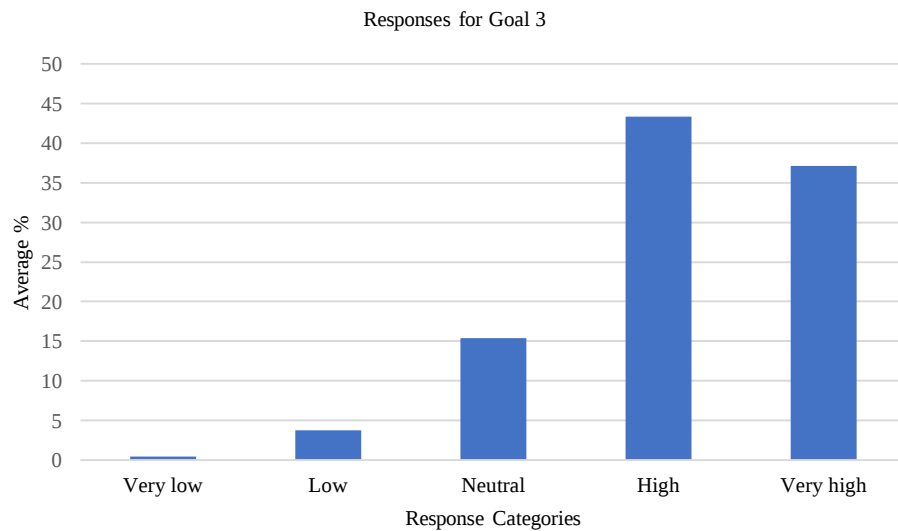


Fig. 6. Cumulative outcomes for goal 3 chart

5.4 Cumulative Analysis of Goal 4

The purpose is to bring transparency and shared work-related content between the distant groups to enhance the means of communication and automation. Products, such as Jira, Slack and continuous integration systems will assist with the organization of the work and with providing feedback in real time eliminating the potential factors that may contribute to a situation where Agile scales up in large-scale projects and eventually reduces efficiency. The results of Goals achievement analysis for Goal 4 are summarized in Table 5 below.

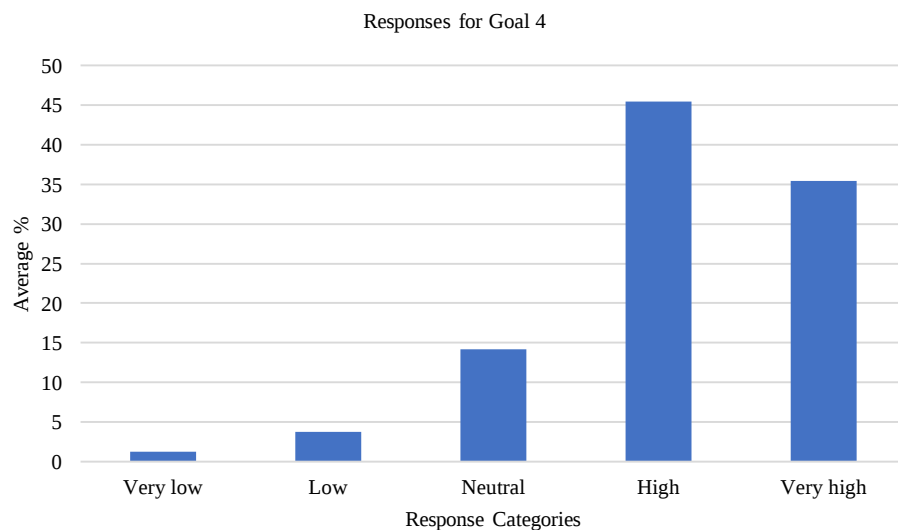


Fig. 7. Cumulative outcomes for goal 4 chart

Concerning Goal 4, the results presented show a positive attitude toward the proposed strategies. Regarding optimism level, 45.42% of the respondents chosen “High” and 35.42% of the respondents chosen “Very High”, so a considerable number of respondents were optimistic in their response. Fourteen-point seventeen percent of the participants had a neutral attitude which means that they had moderate feelings about donating to charity organizations on the other hand, 3.75% of respondents had a “Low attitude” towards the donation and 1.25% of the participants had a “Very low attitude.” All these points to a highly positive approval of the proposed solutions with the majority expressing confidence in how such solutions work as shown in fig. 7.

6. Conclusion

This study presented the Hybrid Distributed Agile Framework (HDAF), which integrates pilot validation, cultural readiness, framework selection, performance measurement, and continuous feedback into a unified model for scaling Agile in distributed environments. Beyond reiterating the proposed goals, the conclusion emphasizes a practical roadmap for implementation. Organizations can begin with small-scale pilots of DevOps and Lean practices, followed by structured adoption of SAFe or LeSS depending on project complexity, and then reinforce adoption through targeted Agile training and communication tooling. At the same time, several limitations should be acknowledged. The validation was based on survey data and small pilot studies rather than long-term organizational deployments, which restricts generalizability. The focus on four goals, while comprehensive, may still overlook other critical aspects such as security integration, cost analysis, or cross-domain applicability. For future research, empirical case studies across diverse industries and geographies should be undertaken to assess the adaptability of HDAF in real-world contexts. Comparative evaluations of SAFe and LeSS implementations under controlled conditions, integration of advanced automation tools (e.g., AI-driven backlog management), and longitudinal studies on cultural change outcomes represent promising directions. By explicitly articulating these limitations and research avenues, this paper moves from proposing a conceptual framework to providing a practical agenda for both practitioners and scholars.

Appendix A: Questionnaire

Name: Arifuzzaman

Designation: MSIT Student

Name & Address of Organization: Information Technology, KAU

Contact #: aarifuzzaman@stu.kau.edu.sa

Signature: Arifuzzaman

The objective of the questionnaire was to empirically validate the Hybrid Distributed Agile Framework (HDAF) by collecting practitioner insights across the four core goals of the proposed solution. Specifically, the questionnaire was designed to:

- Assess Agile Enhancements (Goal 1): Measure the impact of DevOps, Lean, and hybrid Agile practices on performance metrics such as lead time, throughput, defect density, productivity, and customer satisfaction.
- Evaluate Scaled Framework Adoption (Goal 2): Gather feedback on the effectiveness of SAFe and LeSS in coordinating distributed teams, managing dependencies, and reducing project delays.
- Examine Training and Cultural Readiness (Goal 3): Understand the role of Agile training, leadership support, and organizational culture in facilitating Agile transformation.
- Analyze Automation and Communication Tools (Goal 4): Determine how digital platforms (e.g., Slack, Jira, CI/CD pipelines) influence collaboration, transparency, and efficiency in distributed Agile environments.

Please check the appropriate answer.

Filling Guidelines

Lacquered scale is ranging from 1 to 5.

- **Very low** effect indicating 1
- **Low** effect indicating 2
- **Nominal/Average** effect indicating 3
- **High** effect indicating 4
- **Very high** effect indicating 5

Introduction: This questionnaire is developed to gather professional insights on strengthening Agile practices in four key domains: validating Agile enhancements, adopting scaled Agile frameworks for large and distributed projects, fostering organizational alignment through training and cultural readiness, and improving communication and automation in distributed teams. The responses will directly support the empirical validation of the Hybrid Distributed

Agile Framework (HDAF), ensuring that its components are practical, scalable, and organizationally relevant. Your participation will provide valuable evidence for advancing Agile research and practice as shown in figure 3.

1) 2) Questions	Very low	Low	Neutral	High	Very high
3) Goal 1 - To what extent did the use of DevOps improve your team's lead time					
1. How beneficial have you found Agile enhancements (e.g., DevOps, Lean principles, hybrid models) for improving team productivity?					
2. How would you rate the ease of integrating new methodologies (like DevOps or Lean) into your current Agile practices?					
3. How useful are the following metrics (e.g., velocity, cycle time, lead time) in tracking productivity in Agile-enhanced teams?					
4. Do Agile enhancements, like continuous feedback, impact project timelines?					
5. How significant are the following factors (e.g., team alignment, management support, tools) in the success of Agile enhancements in your projects?					
6. Does continuous feedback help your team achieve improved project outcomes?					
7. How effective do you consider Agile enhancements, such as iterative learning, in improving team adaptability to project changes?					
8. How much do Agile enhancements improve the reliability and consistency of project outcomes in your experience?					
4) Goal 2 - How effective was SAFe/LeSS in coordinating multiple distributed teams in your organization?					
1. How effective do you find the Scaled Agile Framework (SAFe) in managing large-scale Agile projects?					
2. Does SAFe support team alignment and collaboration across multiple departments or locations effectively?					
3. How effective is SAFe in synchronizing the work of multiple teams working on different parts of a large project?					
4. How effective is SAFe in helping manage resource allocation for large, complex projects?					
5. How much does implementing SAFe reduce project delays or schedule overruns in large projects?					
6. How significant is the role of management support in the successful implementation of SAFe in large projects?					
7. How well does SAFe accommodate changing requirements in large-scale projects?					
8. How much does SAFe contribute to achieving consistent project outcomes across multiple teams in large-scale projects?					
5) Goal 3- Rate the sufficiency of Agile training in fostering cultural readiness within your team					
1. Do you feel that organizational alignment drives the success of Agile projects in your team?					
2. How much has Agile training boosted your team's productivity and collaboration?					
3. Do you think leadership support is essential for the success of Agile training and alignment efforts?					
4. How much does cross-departmental alignment help create a smooth Agile journey across teams?					
5. Do you feel well-prepared to handle project changes after Agile training?					

6.	How much does clear communication keep everyone aligned during Agile projects?					
7.	Do you believe ongoing Agile training contributes to long-term project success?					
8.	How much have Agile training programs improved your understanding and application of Agile principles?					
5) Goal 4- Evaluate the impact of real-time tools (Slack/Jira) on cross-team transparency						
1.	How much does effective communication improve collaboration in your distributed Agile team?					
2.	Do you feel that automation tools enhance efficiency in distributed Agile projects?					
3.	How much do you rely on communication tools (e.g., Slack/JIRA, Microsoft Teams) to stay aligned with your distributed team?					
4.	Do you believe that automated processes help reduce delays and errors in distributed Agile projects?					
5.	How much does real-time communication impact project success in your distributed Agile team?					
6.	Do you think that having standardized communication protocols improves collaboration across distributed teams?					
7.	How much automation in task management (e.g., using Jira, Trello) has streamlined your workflow in a distributed setup?					
8.	Do you feel that communication and automation improvements significantly contribute to the overall productivity of distributed Agile teams?					

References

- [1] Boehm, B. (1986). A spiral model of software development and enhancement. ACM SIGSOFT Software Engineering Notes, 11(4), 14-24.
- [2] Kotter, J. P. (1996). Leading Change. Harvard Business Review Press.
- [3] Lewin, K. (1947). Frontiers in group dynamics: Concept, method and reality in social science. Human Relations, 1(1), 5-41.
- [4] Amajuoyi, P., Benjamin, L. B., & Adeusi, K. B. (2024). Optimizing agile project management methodologies in high-tech software development. GSC Advanced Research and Reviews, 19(2), 268-274.
- [5] Kula, E., Greuter, E., van Deursen, A., & Gousios, G. (2022). Factors affecting on-time delivery in large-scale agile software development. IEEE Transactions on Software Engineering, 48(9), 3573-3591.
- [6] Binboga, B., & Gumussoy, C. A. (2024). Factors affecting agile software project success. IEEE Access, 12, 95613-95629.
- [7] Psarov, O., & Druzhinin, E. (2024). Enhancing agile team productivity with metrics. Visnyk Ternopil National Technical University, 1(113), 93-104.
- [8] Ekechi, C. C., Okeke, C. D., & Adama, H. E. (2024). Enhancing agile product development with scrum methodologies. Engineering Science & Technology Journal, 5(5), 1542-1570.
- [9] Fawzy, A., et al. (2024). Exploring data management challenges and solutions in agile software development. arXiv preprint arXiv:2402.00462.
- [10] Cunha, F., et al. (2024). An insight into the capabilities of professionals and teams in agile software development. Journal of Communications Software and Systems, 20(1), 99-111.
- [11] Siddique, S., et al. (2023). An effective framework to improve managerial activities in global software development. Nonlinear Engineering, 12.
- [12] Avancha, S., Goel, O., & Pandian, P. K. G. (2024). Agile project planning and execution in large-scale IT projects. Shodh Sagar, 12(3), 239-245.
- [13] Valdés-Rodríguez, Y., et al. (2024). Analysis of strategies for integrating security practices in agile software development. IEEE Access, 12, 35204-35215.
- [14] Lenarduzzi, V., et al. (2020). A systematic literature review on the definition of DevOps. Journal of Systems and Software, 167, 110610.
- [15] Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations. Journal of Systems and Software, 119, 87-108.
- [16] CMMI Product Team. (2010). Capability Maturity Model Integration (CMMI) for Development, Version 1.3. Software Engineering Institute.
- [17] Kitchenham, B., et al. (2004). Evidence-based software engineering. In Proceedings of the 26th International Conference on Software Engineering, 273-281.

Authors' Profiles



Arifuzzaman received the Bachelor of Science in Information Technology (IT) from the University of Information Technology & Sciences, Dhaka, Bangladesh and pursuing Master of Science in Information Technology (IT) from King Abdulaziz University, Jeddah, Saudi Arabia. In addition to earning his degree, he works at the university as a Research Assistant engaging in research on IT and computer systems.



Prof. Dr. Rizwan Qureshi received his Ph.D. degree from National College of Business Administration & Economics, Pakistan 2009. He is currently working as a full Professor in the Department of IT, King Abdulaziz University, Jeddah, Saudi Arabia. This author is the best researcher awardees from the Department of Information Technology, King Abdulaziz, Jeddah, Saudi Arabia. This author is the best researcher awardees from the Department of Information Technology, King Abdulaziz University in 2013 and 2016.

How to cite this paper: Arifuzzaman, Rizwan Jameel Qureshi, "Optimizing Agile System Implementation: Strategies for Enhancing Team Productivity and Project Outcomes in Software Development", International Journal of Education and Management Engineering (IJEME), Vol.15, No.6, pp. 1-14, 2025. DOI:10.5815/ijeme.2025.06.01