Research Article

Queen Arwa University Scientific Refereed Journal

مجلة جامعة الملكة أروى العلمية المحكمة

QAUSRJ

Research Article Data:

PUBLISHER	Queen Arwa University
DOI	**10.58963/qausrj.v28i28.306**
P-ISSN	**2226-5759**
E-ISSN	**2959-3050**
Reception Date	01 Dec 2024
Accepted Date	20 Dec 2024
Published Date	31 Dec 2024
License ©	(CC BY 4.0)

Citation:

Aleryani, R. ., Alsabry, A. ., & Algawani, R. S. . (2024). Critical Success Factors in Scrum Software Development: A Comprehensive Literature Review. *Queen Arwa University Journal*, 28(28), 22. https://journal.qau.edu.ye/index.php/srj/article/view/306

Main contact:

Author: *Rania Aleryani*

Phone: 9677742652262+

Published Email: raniaalaryani@iutt.edu.ye

Organization/University/ Center:

Affiliation: *International University of Technology Twintech .*

Research funder: *Not found.*

Research field/specialization:

General Management

QR code:

Scan QR code to visit this journal on your mobile device.

امسح الكود لزيارة موقع المجلة

Critical Success Factors in Scrum Software Development: A Comprehensive Literature Review

Rania Aleryani[1], Dr. Ayman Alsabry[1], Dr.Emad Alramada[1], Rana Algawani[2]

[1] *International University of Technology Twintech, Sana'a, Yemen.*

[2] *Federation of Yemen Chambers of Commerce and Industry, Sana'a, Yemen.*

2024

Abstract:

This paper presents a comprehensive literature review on the Critical Success Factors influencing the successful implementation of Scrum in software development projects. Scrum, as a widely adopted agile methodology, enhances flexibility, collaboration, and iterative delivery of software solutions. However, its effectiveness is contingent on multiple factors across various dimensions, including organizational, people, process, technical, and project factors. This review synthesizes existing research to categorize and analyze these factors, providing valuable insights for researchers and practitioners alike. Key findings highlight the critical role of management commitment, team dynamics, customer involvement, agile software techniques, and adaptable project management processes in achieving successful Scrum outcomes. The paper also identifies challenges such as scope creep and stakeholder misalignment.

Keywords:

Scrum, Agile Methodology, Critical Success Factors, Software Development, Project Management, Team Collaboration, Iterative Delivery, Organizational Factors, Customer Involvement, Management Commitment

ROR: 03ygqq617

ISSN : 2226 - 5759

ترجمة الى العربية

العوامل الحرجة لنجاح تطوير البرمجيات باستخدام سكرم: مراجعة شاملة للأدبيات

رانيا الارياني[1]، ايمن الصبري[1]،
عماد الرمادة[1]، رنا صالح الجعوني[2]

[1] جامعة تونتك الدولية للتكنولوجيا، صنعاء، اليمن.
[2] الاتحاد العام للغرف التجارية والصناعية اليمنية، صنعاء، اليمن.

2024

الملخص

تقدم هذه الورقة مراجعة شاملة للأدبيات المتعلقة بالعوامل الحرجة لنجاح تنفيذ منهجية سكرم في مشاريع تطوير البرمجيات. تُعتبر سكرم، كمنهجية رشيقة واسعة الانتشار، وسيلة لتعزيز المرونة والتعاون وتسليم الحلول البرمجية بشكل تدريجي. ومع ذلك، تعتمد فعالية سكرم على العديد من العوامل عبر أبعاد مختلفة، بما في ذلك العوامل التنظيمية، البشرية، العملية، التقنية، وعوامل المشروع. يقوم هذا الاستعراض بتجميع وتحليل الأبحاث القائمة لتصنيف هذه العوامل، مما يوفر رؤى قيمة للباحثين والممارسين على حد سواء. وتبرز النتائج الرئيسية الدور الحاسم لالتزام الإدارة، ديناميكيات الفريق، مشاركة العملاء، تقنيات البرمجيات الرشيقة، وعمليات إدارة المشاريع القابلة للتكيف في تحقيق نتائج ناجحة باستخدام سكرم. كما تحدد الورقة التحديات مثل تمدد نطاق المشروع وسوء توافق أصحاب المصلحة.

الكلمات المفتاحية

سكرم، المنهجية الرشيقة، العوامل الحرجة للنجاح، تطوير البرمجيات، إدارة المشاريع، تعاون الفريق، التسليم التدريجي، العوامل التنظيمية، مشاركة العملاء، التزام الإدارة

Introduction

Scrum is one of the most widely adopted agile methodologies for software development, designed to promote flexibility, collaboration, and iterative progress in project management. Originating in the early 1990s, Scrum was developed as a framework to address the limitations of traditional, linear project management models such as the Waterfall method (Schwaber & Sutherland, 2011). By emphasizing adaptive planning, continuous improvement, and rapid delivery of functional software, Scrum has become an essential approach for managing complex and dynamic software development projects.

At its core, Scrum operates through a series of iterative cycles known as sprints, typically ranging from two to four weeks. Each sprint involves the development of a potentially deliverable product increment. Scrum encourages cross-functional collaboration, with roles clearly defined to ensure accountability and ownership of tasks. The key roles within Scrum include the Product Owner, who is responsible for maximizing product value through effective backlog management; the Scrum Master, who facilitates the Scrum process and ensures adherence to Scrum principles; and the Development Team, a self-organizing group responsible for delivering the product increments ((Brose et al., 2023; Kadenic, de Jesus Pacheco, et al., 2023; Salin et al., 2023).

One of the primary strengths of Scrum is its focus on communication and transparency. Through events like daily stand-up meetings (daily Scrums), sprint reviews, and retrospectives, teams are able to maintain constant communication, address potential issues early, and continuously refine both the product and their development processes. This iterative approach, combined with frequent stakeholder feedback, enables Scrum teams to respond effectively to changing requirements, making it a highly adaptive methodology suited for projects where the scope may evolve over time (Kostin & Strode, 2022; Sokolowski, 2022).

The adoption of Scrum has expanded beyond the software industry into various sectors, including manufacturing, marketing, and product development, demonstrating its versatility as a project management framework (Goins, 2024). However, despite its advantages, Scrum also faces challenges, particularly in large-scale projects or organizations that are not accustomed to agile practices. The successful implementation of Scrum relies heavily on understanding and addressing the critical success factors that drive its effectiveness, which this study aims to explore through a comprehensive review of the literature.

Importance of Identifying Critical Success Factors (CSFs)

The identification of Critical Success Factors (CSFs) is essential for the successful implementation of the Scrum software development methodology (Junior & Aquino, 2024). As an iterative and adaptive framework, Scrum depends heavily on a variety of factors that influence its success across different projects and organizations. While the methodology provides a structured approach to managing uncertainty and complexity, the degree to which it can achieve its objectives is contingent on several key organizational, technical, and project-specific factors. Understanding these factors is critical to ensuring that Scrum delivers the desired outcomes in software development environments.

identifying CSFs provides valuable insights for both researchers and practitioners, enabling them to understand the conditions that foster successful Scrum implementations (Ekechi et al., 2024). This knowledge can be applied to refine agile practices, develop targeted training and development programs for Scrum teams, and design organizational structures that support agile methodologies. Furthermore, recognizing these factors contributes to the broader understanding of how agile approaches can be scaled and adapted to various industries and project types, enhancing the overall body of knowledge in the field of software development.

Research Problem and Rationale

Despite the widespread adoption of Scrum as an agile methodology for managing software development projects, the success of its implementation varies significantly across organizations and projects. While Scrum provides a structured framework designed to enhance adaptability, collaboration, and efficiency, its successful deployment is not guaranteed in every context. Research has shown that numerous factors, both internal and external to project teams, can directly influence the outcomes of Scrum projects. These factors, often referred to as Critical Success Factors (CSFs), determine whether the methodology will be effective in delivering high-quality results within the specified constraints. However, the lack of a comprehensive understanding of these CSFs represents a significant gap in the literature and in practice.

The primary research problem addressed by this study is the absence of a unified and thorough examination of the critical success factors that drive Scrum's effectiveness in software development. Although numerous studies have examined individual aspects of Scrum success—such as team dynamics, customer collaboration, or technical processes—there remains a need for a holistic review that synthesizes the diverse factors identified in existing literature. Without a clear and consolidated framework of CSFs, organizations may struggle to prioritize the elements that are most crucial for the success of their Scrum implementations. Furthermore, the complexity and variability of software development projects mean that success factors may differ depending on the organizational environment, project type, and technical requirements, adding to the challenge of establishing universally applicable guidelines for Scrum success.

The rationale for conducting this research stems from the need to provide a more complete and evidence-based understanding of the factors that contribute to the success of Scrum projects. By reviewing and analyzing the existing literature, this study aims to identify the key organizational, people, process, technical, and project-related factors that influence Scrum outcomes. In doing so, it seeks to fill a gap in the current knowledge base and provide practitioners with actionable insights that can be applied to enhance the effectiveness of their Scrum practices. Additionally, this research will contribute to the academic discourse by offering a comprehensive synthesis of the critical elements necessary for Scrum success, facilitating the development of best practices and frameworks that can be used to guide future projects.

The growing reliance on agile methodologies, particularly Scrum, in software development highlights the practical importance of this research. As organizations increasingly move toward iterative, flexible development approaches, understanding the factors that drive Scrum's success is vital for ensuring project efficiency, quality, and stakeholder satisfaction. By addressing the research problem, this study aims to provide both theoretical and practical contributions to the field, offering organizations the tools they need to better implement and sustain successful Scrum practices across varying project environments.

Research Aims and Objectives

The primary aim of this study is to conduct a comprehensive literature review on the CSFs in Scrum software development projects, with the goal of synthesizing and categorizing the findings from existing research. As a review paper, the study seeks to provide an organized and coherent overview of the factors identified in prior studies that contribute to the successful implementation of Scrum. By consolidating the current body of knowledge, the study aims to offer both academic and practical insights that can guide future research and inform Scrum practitioners about the key determinants of success.

The specific objectives of this review are:

- To systematically collect and analyze the existing literature on CSFs in Scrum software development projects, focusing on research from academic journals, conference papers, and empirical studies.
- To categorize the identified success factors into distinct groups, such as **organizational, people, process, technical, and project-related factors**, providing a structured understanding of the factors that influence Scrum project outcomes.
- To explore the relationships between these factors, identifying common themes and patterns across different studies and highlighting how various factors interact to affect Scrum success.
- To summarize the key challenges and barriers to successful Scrum implementation as discussed in the literature, and to propose strategies or best practices that can address these challenges.
- To provide recommendations for future research based on gaps identified in the reviewed literature, with a focus on areas that require further investigation or empirical validation.

Through this literature review, the study aims to contribute to the existing body of knowledge on Scrum by offering a well-rounded analysis of the factors that drive success, while also providing practical guidelines for improving Scrum practices in software development environments.

Literature Review

In today's business landscape, information systems and software solutions play a crucial role in operational efficiency, leading to a swift growth of the software development and outsourcing sectors globally (Sutrisno et al., 2023). As businesses increasingly rely on advanced software products, having state-of-the-

~~art technology has become a necessity for~~ achieving success. This rising demand has made the success of IT projects a significant concern for industry experts. Furthermore, the concept of "project success" has emerged as a key focus for research, especially since many projects still struggle with issues related to budget, timelines, and quality (Pinto et al., 2022). Despite substantial investments in software solutions, many organizations have not seen the anticipated returns, as noted by various researchers. Consequently, software development firms are actively seeking strategies to enhance project success, enabling businesses to optimize their investments through effective software applications in their everyday operations.

The Agile development methodology has gained significant traction in software projects as a way to boost success rates (Natarajan & Pichai, 2024). Before Agile's emergence, the software industry primarily relied on traditional, plan-driven approaches like the Waterfall model, Unified Process, and Spiral methodology. These conventional methods adhered to a linear and structured framework for software creation, placing a strong emphasis on formal communication and detailed documentation at every project phase. Any major modifications required approval from a change control board, and project milestones needed formal sign-offs to advance, with clearly defined roles and a strict division of responsibilities. Agile, in contrast, disrupted these rigid processes by prioritizing collaboration with customers. Many researchers have pointed out that traditional methods often led to frustration and challenges due to their linear nature, making it costly and time-consuming to revisit earlier stages for corrections. As a result, industry experts began to explore and develop alternative software development methodologies that allowed for incremental improvements throughout the project lifecycle (Al-Ashmoery et al., 2023).

~~Consequently, the Agile development~~ methodology was introduced in 2001 and has since been recognized for its flexibility and iterative approach to software development. The "Agile Manifesto," created in the same year, outlines fundamental principles designed to steer Agile practices. This methodology emphasizes delivering customer value, facilitating iterative and incremental progress, fostering intense collaboration, and promoting small, integrated teams that are self-organizing (Kakar & Kakar, 2023). It also encourages ongoing, minor improvements to effectively address the challenges that arise from changes, which are more prevalent in IT projects compared to other types of projects.

Agile development is a software development approach that prioritizes people and interactions over extensive documentation. The term "Agile methods" encompasses various processes that align with this methodology. Some of the most commonly utilized Agile methods include Scrum, Extreme Programming (XP), Feature-Driven Development (FDD), Test-Driven Development (TDD), and Lean software development (Alqudah et al., 2024). These methods each offer unique practices and frameworks that facilitate the Agile philosophy.

Scrum is an iterative and incremental approach to project management characterized by a straightforward framework that emphasizes continuous inspection and adaptation (ALSHURIDEH et al., 2024). It is viewed as a modern, collaborative, and lean model for software development, enabling developers to accommodate late changes to requirement specifications during the implementation phase. Industry professionals often consider various project attributes—such as objectives, scope, requirements, resources, software architecture, and project size—when choosing a software development methodology. According to research and successful project case studies, many scholars and practitioners have found that Scrum is particularly effective in managing projects, especially in the face of the complexities inherent in contemporary software systems and the fast-paced changes within the business landscape (Ekechi et al., 2024).

Methodology

1. Search Strategy

A systematic search was conducted to identify relevant literature on the Critical Success Factors (CSFs) influencing Scrum software development projects. The search targeted peer-reviewed journals, conference papers, and empirical studies from reputable databases, including IEEE Xplore, Scopus, and Google Scholar. Keywords used in the search included "Scrum," "Agile Methodology," "Critical Success Factors," "Software Development," "Project Success," and "Team Collaboration." Boolean operators were employed to refine the results, combining terms like "Scrum AND Critical Success Factors" and "Agile AND Project Management." The search was restricted to studies published in English from 2010 to 2024 to ensure contemporary relevance.

2. Inclusion Criteria

- Studies that explicitly focus on Scrum as the primary software development methodology.
- Literature that identifies or analyzes Critical Success Factors (CSFs) in Scrum implementations.
- Empirical studies, literature reviews, and case studies related to Scrum project success.
- Papers that discuss both organizational and technical factors influencing Scrum outcomes.
- Studies published in peer-reviewed journals or conference proceedings.

3. Exclusion Criteria

- Studies focused on general agile methodologies without a specific emphasis on Scrum.
- Papers that do not address success factors in software development projects.
- Non-peer-reviewed sources, including blog posts, white papers, and opinion pieces.
- Studies published before 2010 to avoid outdated practices and findings.
- Literature in languages other than English, to maintain consistency in analysis.

Analysis of Critical Success Factors

1. Organization factors

1.1. Relationship between management commitment and scrum software development project success

The success of Scrum software development projects is often contingent upon multiple factors, with management commitment emerging as a crucial element influencing project outcomes. Scrum, as an agile framework, relies heavily on the active participation of all stakeholders, especially management, to foster an environment conducive to collaborative decision-making, iterative progress, and flexibility. In this context, management commitment refers to the sustained support, engagement, and involvement of senior leadership in facilitating Scrum processes and ensuring alignment with organizational goals. This discussion critically examines the relationship between management commitment and the success of Scrum projects, drawing from empirical research and theoretical perspectives.

1.1.1. Management Commitment as a Critical Success Factor (CSF)

Several studies emphasize management commitment as a CSF for Scrum projects. Management plays a pivotal role in establishing the organizational culture and infrastructure required for Scrum to thrive. Without robust management support, teams may encounter difficulties in securing the necessary resources, managing stakeholder expectations, and navigating organizational constraints.

Research by (Doan et al., 2020) found that management commitment is directly correlated with higher project success rates in agile environments.

The importance of leadership in fostering a culture of trust and openness within the Scrum team was also highlighted, as this can significantly impact the team's ability to self-organize and innovate. Management's ability to remove organizational barriers and encourage continuous learning further strengthens Scrum’s adaptive capabilities, which is critical for success in dynamic project environments.

1.1.2. Impact on Scrum Principles and Practices

Scrum’s effectiveness is rooted in its core principles of self-organization, iterative development, and cross-functional collaboration. For these principles to be fully realized, management must demonstrate a commitment to empowering teams and granting them the autonomy to make decisions. Without such empowerment, the core agile principle of team self-organization can be undermined, leading to ineffective execution of Scrum practices. (Kalluri, 2022) argue that management must actively champion Scrum values, supporting teams in adopting iterative processes and facilitating the rapid delivery of functional software.

Leadership’s role in facilitating Scrum ceremonies (such as sprint planning, sprint reviews, and retrospectives) is also critical. Research by (Landaeta et al., 2011) suggests that management's involvement should be strategic rather than intrusive, enabling teams to maintain ownership over their processes while ensuring alignment with broader organizational objectives.

When management demonstrates commitment through participation in key Scrum events, it reinforces the importance of these ceremonies and sets a precedent for the team’s adherence to Scrum practices.

Furthermore, management commitment extends to the provision of adequate tools and resources. In agile methodologies, access to the right technology and support systems is essential for ensuring efficiency in communication, testing, and continuous integration. Studies have shown that when management is proactive in providing these resources, the teams experience smoother project execution and higher productivity (Gmach et al., 2009). In contrast, a lack of commitment in this area can lead to technical debt and delays, negatively impacting project outcomes.

1.1.3. Challenges of Insufficient Management Commitment

The absence of strong management commitment can have significant adverse effects on Scrum project success. (Stettina & Hörz, 2015) highlight that in organizations where management is not fully invested in agile transformation, teams often struggle with competing priorities and insufficient buy-in from key stakeholders. This can lead to a breakdown in communication, misalignment of goals, and limited support for agile practices, which in turn results in project delays, scope creep, and diminished product quality. Furthermore, without visible and consistent commitment from leadership, the motivation of Scrum teams may wane, particularly in challenging phases of a project.

One of the key challenges identified in the literature is the misalignment between traditional management mindsets and Scrum’s agile values. (Luna et al., 2014) argue that management often struggles to shift from a command-and-control style of leadership to one that supports collaboration, flexibility, and iterative delivery. This disconnect can create friction between teams and management, undermining the trust and autonomy necessary for Scrum teams to perform effectively. When management fails to embrace the principles of agility fully, teams may be forced to revert to traditional, hierarchical approaches, which conflict with Scrum's emphasis on empowerment and adaptability.

1.1.4. Recommendations for Enhancing Management Commitment

To enhance the role of management in Scrum project success, several best practices have been proposed. First, agile training and coaching for management is essential to align leadership with Scrum values and principles. (Pop, 2022) suggest that organizations should invest in educating managers about the benefits and challenges of Scrum, as well as their role in facilitating an agile culture. This training can help managers better understand the importance of providing autonomy to teams, offering strategic support, and fostering a learning-oriented environment.

Second, management should play an active role in communicating the value of Scrum to the broader organization. Clear communication about the goals of Scrum, its processes, and the expected outcomes can help align stakeholders and reduce resistance to change. (Almeida & Espinheira, 2022) emphasize the importance of management acting as Scrum advocates within the organization, actively supporting the agile transformation process and addressing any organizational impediments that may hinder Scrum practices.

Lastly, sustained involvement in agile governance and continuous feedback loops are critical. Management should not only commit to Scrum at the outset of a project but should remain engaged throughout the lifecycle of the project, providing ongoing feedback, resources, and support. This sustained commitment ensures that Scrum practices are upheld and that any challenges encountered during the project are addressed in a timely and effective manner.

1.2. Relationship Between Organizational Environment and Scrum Software Development Project Success

The success of Scrum software development projects is not solely determined by internal team dynamics or technical expertise; the organizational environment plays a critical role in shaping the conditions under which Scrum can effectively thrive. The organizational environment refers to the broader structural, cultural, and procedural context in which Scrum is implemented, encompassing factors such as leadership support, communication structures, decision-making processes, and the degree of flexibility within the organization. This discussion critically examines how the organizational environment influences the success of Scrum software development projects, drawing on empirical studies and theoretical perspectives to highlight both enabling factors and potential obstacles.

1.2.1. Organizational Culture as a Determinant of Scrum Success

Organizational culture is a central component of the environment in which Scrum operates, with studies indicating that an agile-friendly culture significantly enhances the likelihood of Scrum success (Adeniyi et al., 2024; Handri et al., 2024). Agile culture typically emphasizes values such as adaptability, continuous improvement, collaboration, and customer focus—qualities that align closely with Scrum principles. When an organization fosters a culture of trust, empowerment, and openness to change, Scrum teams are more likely to self-organize effectively, communicate openly, and make iterative improvements.

In contrast, organizations with rigid hierarchical structures and bureaucratic decision-making processes may find it difficult to adopt Scrum practices. (Dwi Harfianto et al., 2022) argue that in such environments, team members often struggle to exercise the level of autonomy required in Scrum, as they may face resistance from upper management or conflicting demands from different departments. This misalignment between organizational culture and Scrum values can lead to friction, misunderstandings, and reduced effectiveness of the Scrum framework. Therefore, an organizational culture that supports agile values is not only beneficial but essential for the success of Scrum projects.

Moreover, (Jones, 2010) highlights the importance of leadership's role in shaping organizational culture. Leaders who actively promote agile values create an environment conducive to Scrum by fostering collaboration, encouraging transparency, and providing teams with the freedom to experiment and innovate. In such organizations, leadership is seen as an enabler rather than a controller of the development process, empowering teams to make decisions and take ownership of project outcomes.

1.2.2. Organizational Structure and Its Influence on Scrum

The structure of an organization—how it organizes its people, processes, and resources—can either facilitate or hinder Scrum adoption. (Andrei, 2024) assert that a flexible, cross-functional organizational structure is key to the successful implementation of Scrum. In Scrum, teams are expected to be self-sufficient, meaning that they must include all the necessary skills and expertise to deliver increments of a product without constantly relying on external resources or departments. When organizational structures are too siloed, with clear separations between departments such as development, testing, and operations, Scrum teams may find it challenging to operate autonomously. This can lead to delays, communication breakdowns, and difficulty in maintaining a continuous flow of work.

A cross-functional team structure, on the other hand, enables collaboration across different disciplines, which is a cornerstone of agile methodologies like Scrum. This structure allows Scrum teams to work more efficiently and fosters a culture of shared responsibility for project outcomes. (Shakir et al., 2023) emphasize that such team configurations are essential for the type of close collaboration and communication that Scrum demands, and they help in breaking down the silos that can obstruct progress in traditional, hierarchical organizations.

However, organizational flexibility goes beyond team composition. (Hron & Obwegeser, 2022) notes that for Scrum to be truly effective, organizations must be willing to adapt their processes and policies to fit the agile framework. For example, if an organization's existing processes are too rigid or are based on extensive documentation and approval hierarchies, Scrum teams may find it difficult to move quickly and iterate effectively. The ability of an organization to adjust its processes to accommodate Scrum's emphasis on adaptability and continuous delivery is a critical factor in project success.

1.2.3. Communication Structures and Information Flow

Effective communication is another critical element within the organizational environment that affects Scrum project success. Scrum relies on frequent, clear communication through mechanisms such as daily stand-ups, sprint reviews, and retrospectives. In a supportive organizational environment, these communication practices are reinforced by open, transparent channels of communication across the entire organization. (Sharp et al., 2012) found that when information flows freely between Scrum teams and upper management, as well as across departments, it enables better decision-making and ensures that the team's work remains aligned with organizational goals.

However, poor communication structures can inhibit the flow of information and create disconnects between Scrum teams and the broader organization. For example, if there is a lack of clarity in reporting lines or if there is insufficient engagement from stakeholders, Scrum teams may lack the context and feedback they need to adjust their work according to changing priorities. (Jeleel-Ojuade, 2024) highlight that this disconnect can lead to misalignment between the product delivered by the team and the organization's strategic objectives, reducing the project's overall success.

Moreover, (Luca, 2022) argue that communication barriers often arise when organizations adopt hybrid models, attempting to blend traditional waterfall methodologies with agile practices. In such environments, Scrum teams may struggle with differing expectations and reporting requirements, which can lead to confusion and inefficiencies. For Scrum to succeed, the organizational environment must support a consistent, agile-friendly communication structure, ensuring that all stakeholders—regardless of their role—are on the same page regarding project goals and processes.

1.2.4. *Organizational Policies and Agile Governance*

Another aspect of the organizational environment that influences Scrum success is the presence (or absence) of agile-friendly organizational policies and governance structures. (Ekechi et al., 2024) suggest that successful Scrum implementations often occur in organizations that have adjusted their policies to support agile governance. This may include reducing the reliance on traditional metrics (such as time and cost adherence) in favor of metrics that capture agile success, such as customer satisfaction, product quality, and team velocity.

Organizations that maintain rigid governance models, focusing primarily on controlling scope, cost, and deadlines, may struggle to reap the full benefits of Scrum. (Arif, 2024) assert that for Scrum to succeed, governance models must embrace flexibility, allowing for changing requirements and iterative delivery. This flexibility must extend to how projects are funded and resourced, as well as how performance is evaluated. Agile governance structures should also include mechanisms for continuous feedback, both from customers and from internal stakeholders, to ensure that Scrum teams can adapt their work in response to evolving needs and conditions.

1.2.5. *Resistance to Change and Organizational Transformation*

One of the key challenges in aligning the organizational environment with Scrum's requirements is resistance to change. Agile transformations often require significant shifts in mindset, processes, and organizational structures, and these changes are not always welcomed by all stakeholders. (Weichbroth, 2022) found that resistance to adopting agile methodologies is a common barrier in organizations with deeply entrenched traditional processes. Employees may feel uncomfortable with the loss of familiar practices, while managers may be reluctant to relinquish control over decision-making processes. This resistance can create friction within the organization, impeding the successful implementation of Scrum.

Change management strategies are therefore crucial to ensuring that the organizational environment is conducive to Scrum's success. (Jovanović et al., 2017) argue that agile transformations should be accompanied by targeted efforts to educate stakeholders about the benefits of agile methodologies and to address concerns about the transition. Leadership must also play a central role in advocating for change, demonstrating a clear commitment to the agile vision and providing ongoing support to teams as they adjust to new ways of working.

1.3. *Relationship Between Team Environment and Scrum Software Development Project Success*

The team environment plays a pivotal role in the success of Scrum software development projects, as Scrum's framework heavily relies on the interaction, collaboration, and self-organization of cross-functional teams. The team environment encompasses the interpersonal dynamics, communication practices, and collaborative culture within the development team, all of which are critical for enabling effective Scrum practices. This discussion delves into how the team environment impacts the success of Scrum projects, analyzing the factors that contribute to or impede project outcomes and drawing on empirical research and theoretical perspectives to provide a critical assessment(Moe et al., 2010).

1.3.1. Self-Organizing Teams and Autonomy in Scrum Success

One of the core tenets of Scrum is the concept of self-organizing teams, where members are expected to manage their own work without constant supervision or micromanagement. This autonomy is essential for fostering creativity, accountability, and rapid decision-making, which are crucial for adapting to changing project requirements. (Hoda et al., 2010) emphasize that the success of self-organizing teams is highly dependent on the team environment. A supportive and trust-based environment enables team members to take initiative, experiment, and make critical decisions without fear of failure or retribution. This level of empowerment enhances team productivity and morale, leading to better project outcomes.

However, the effectiveness of self-organization can be undermined in a toxic or unsupportive team environment. (Tyagi et al., 2022) found that when team members lack trust or exhibit poor communication, the benefits of self-organization are greatly diminished. In such environments, teams may struggle to coordinate their efforts, leading to inefficiencies and delays. Additionally, a lack of trust between team members can lead to reluctance in sharing ideas or providing honest feedback, which stifles innovation and continuous improvement, both of which are central to the Scrum methodology.

Furthermore, (Deci & Ryan, 2012) self-determination theory suggests that autonomy is a key psychological need for individuals to perform at their best. In the context of Scrum, when teams are granted autonomy and supported by a positive environment, they are more likely to experience intrinsic motivation, which drives higher performance and better project outcomes. Therefore, fostering a team environment that encourages autonomy, trust, and self-management is critical for Scrum project success.

1.3.2. Psychological Safety and Open Communication

A critical factor in a successful team environment is the presence of psychological safety, defined as a shared belief that the team is safe for interpersonal risk-taking. In psychologically safe environments, team members feel comfortable voicing their opinions, sharing new ideas, and admitting mistakes without fear of embarrassment or punishment. (Alami et al., 2023) argues that psychological safety is essential for fostering open communication, collaboration, and learning within teams—attributes that are central to Scrum's emphasis on continuous improvement through retrospectives and daily stand-ups.

The absence of psychological safety can lead to communication breakdowns, with team members hesitating to raise concerns or suggest improvements. (Mogård et al., 2022) points out that in environments where psychological safety is lacking, individuals are more likely to disengage or withhold valuable information, which can lead to poor decision-making and hinder the team's ability to respond to changing project demands. In Scrum, where feedback loops and iterative adjustments are vital, such a breakdown in communication can have serious consequences for project success.

Empirical studies reinforce the importance of psychological safety in agile teams. (Jain et al., 2016) found that teams with higher levels of psychological safety were more likely to engage in effective communication, resulting in better decision-making and higher-quality deliverables. Moreover, these teams were more resilient in the face of challenges, as they were able to adapt quickly and collaboratively when problems arose. Thus, creating a team environment that prioritizes psychological safety is critical for maximizing the effectiveness of Scrum practices.

1.3.3. Collaboration and Cross-Functional Team Dynamics

Scrum is designed to be executed by cross-functional teams that possess all the necessary skills to complete a project increment from start to finish. In such teams, members are expected to collaborate closely, with individuals from different disciplines (e.g., developers, testers, and product owners) working together to achieve common goals. (Ames Zegarra & Sabanovic, 2022; Andrei, 2024) argue that cross-functional collaboration is one of the key drivers of success in agile methodologies, including Scrum, as it enables teams to deliver high-quality products quickly and efficiently.

However, the success of cross-functional collaboration depends on the quality of the team environment. (Ames Zegarra & Sabanovic, 2022; Holland et al., 2000) highlight that collaboration in cross-functional teams can be challenging when there are interpersonal conflicts, a lack of role clarity, or misalignment of goals. For example, if team members from different functional areas (such as development and testing) do not communicate effectively or if there is tension between team members, it can lead to delays, poor quality, and a lack of cohesion. Therefore, fostering a collaborative team environment where members from different disciplines respect each other's expertise and work together toward shared objectives is crucial for Scrum success.

Additionally, (Tutty & Klein, 2008) found that teams that engaged in regular face-to-face communication and collaborative problem-solving were more likely to deliver successful project outcomes. In Scrum, practices such as daily stand-ups, sprint planning, and retrospectives are designed to facilitate this type of ongoing collaboration and communication. A team environment that supports and encourages these practices is essential for ensuring that Scrum teams can work effectively and efficiently.

1.3.4. Conflict Resolution and Team Cohesion

While some level of conflict is inevitable in any team, the way conflicts are managed within the team environment has a significant impact on the success of Scrum projects. (Elenurm & Fabritius, 2023; van Greunen et al., 2021) note that while task-related conflict (conflict about the content and direction of the work) can be beneficial for promoting new ideas and improvements, relational conflict (conflict arising from interpersonal issues) is generally harmful to team performance. In the context of Scrum, where collaboration and open communication are paramount, unresolved relational conflicts can be particularly detrimental.

A healthy team environment is one in which conflicts are addressed constructively, and team members work together to resolve disagreements in a way that benefits the project. (Coleman et al., 2021) emphasize that high-performing teams are those that can engage in productive conflict resolution, where differing viewpoints are discussed openly, and solutions are reached through consensus. In Scrum, this type of conflict resolution is often facilitated through retrospectives, where the team reflects on what went well, what didn't, and how to improve in the future.

On the other hand, unresolved conflicts can lead to a breakdown in team cohesion, which is critical for Scrum success. (Varma & Gupta) found that teams with higher levels of cohesion were more likely to perform well and deliver successful project outcomes, as team members were more willing to support each other, share knowledge, and work together toward common goals. Therefore, fostering a team environment that encourages open communication and constructive conflict resolution is key to ensuring that Scrum projects remain on track.

1.3.5. Leadership and Team Support

The role of leadership in shaping the team environment cannot be overlooked. In Scrum, the Scrum Master is responsible for facilitating the Scrum process and ensuring that the team environment is conducive to productivity and collaboration. (Majka, 2024) argues that effective Scrum Masters act as servant leaders, focusing on removing obstacles that impede the team’s progress and fostering an environment of trust and support. By doing so, they help create a team environment where members feel empowered to take ownership of their work and collaborate effectively.

However, the influence of leadership extends beyond the Scrum Master. (Jerab & Mabrouk, 2023) found that leadership support from upper management also plays a crucial role in shaping the team environment. When leaders at all levels demonstrate a commitment to agile values, provide resources, and offer encouragement, teams are more likely to feel supported and motivated to succeed. Conversely, a lack of leadership support can lead to a demotivated team environment, where members feel isolated and unsupported, ultimately undermining the success of the Scrum project.

2. People factors

2.1. Relationship Between High-Caliber Teams and Scrum Project Success

The success of software development projects in Scrum is significantly influenced by the composition and skills of the teams involved. High-caliber teams, noted for their exceptional skills, experience, and collaboration, enhance Scrum's effectiveness. Conversely, lower-caliber teams may struggle with the agile framework. Skilled team members utilize Scrum principles—self-organization, collaboration, iterative delivery, and adaptability—to drive successful outcomes. This discussion examines how these high-caliber teams contribute to project success while considering potential challenges.

2.1.1. Expertise and Technical Proficiency

High-caliber teams possess deep technical proficiency and domain expertise, allowing them to produce high-quality software efficiently. According to (Zeng et al., 2024), this expertise is vital for informed decision-making during Scrum processes. Such teams can handle complex challenges and swiftly adapt to new technologies, which is crucial for maintaining productivity in dynamic Scrum environments. However, reliance on technical expertise can lead to overconfidence and resistance to external feedback, potentially misaligning the product with stakeholder needs.

2.1.2. Collaboration and Communication Skills

Effective collaboration and communication are essential for Scrum success. High-caliber teams excel in these areas, utilizing practices like daily stand-ups and retrospectives to enhance teamwork. (Ewim et al., 2024) highlight that strong communication helps keep teams aligned with project goals. Nonetheless, conflicts may arise among highly skilled individuals, which can disrupt collaboration. Managing these dynamics is crucial to maintaining team cohesion.

2.1.3. Self-Organization and Decision-Making

Self-organization is a core Scrum principle, and high-caliber teams typically embrace it effectively. They can plan, manage work, and make decisions autonomously, which boosts engagement and productivity (Moe et al., 2008). However, such autonomy can lead to resistance to guidance from Scrum Masters or Product Owners, risking misalignment with project objectives. Balancing self-organization with oversight is essential for maintaining focus.

2.1.4. Continuous Improvement and Learning Culture

High-caliber teams often prioritize continuous improvement by regularly reflecting on their processes and seeking new knowledge. This culture of learning helps teams maintain a competitive edge. However, there's a risk of focusing too heavily on technical improvements at the expense of customer value, which can lead to misaligned products. It is critical to ensure that improvement efforts align with customer needs (Ahmed et al., 1999).

2.1.5. Team Diversity and Knowledge Sharing

Diverse skills and experiences within high-caliber teams enhance Scrum project outcomes by fostering innovation and effective problem-solving (Abadir et al., 2019; Chow & Cao, 2008). Additionally, a culture of knowledge sharing is essential for collaborative growth. However, the presence of "heroes"—highly skilled individuals reluctant to share—can lead to knowledge silos, compromising project success. Actively fostering a culture of sharing is necessary to empower all team members.

2.2. Relationship Between Customer Involvement and Scrum Software Development Project Success

Customer involvement is fundamental to Scrum and agile methodologies. It fosters collaboration throughout development, ensuring products meet evolving customer needs. The Scrum Guide (Eboh, 2024) emphasizes regular communication with customers to refine the product backlog and adjust project scope. This relationship between customer involvement and project success is critical, highlighting both benefits and challenges.

2.2.1. Customer Involvement and Agile Principles

Agile principles prioritize customer collaboration over contract negotiation. Scrum incorporates customer feedback through key phases like sprint planning and reviews, ensuring alignment with customer expectations. The iterative nature of Scrum encourages ongoing feedback, reducing miscommunication risks common in traditional models (Liu et al., 2019). Effective customer involvement helps maintain focus on delivering value, allowing teams to adapt to changing needs without losing relevance.

2.2.2. Benefits of Early and Continuous Customer Involvement

Early and continuous customer involvement helps prevent scope creep and misalignment. Traditional methods often seek feedback only at the project's start and end, risking misaligned features. Scrum's approach ensures that priorities are revisited regularly, clarifying requirements and enhancing project success However, (Larusdottir et al., 2016) note that this requires commitment from both teams and customers; lack of involvement can hinder iterative benefits.

2.2.3. Quality of Customer Feedback and Its Impact on Project Success

The quality of customer feedback is crucial. In Scrum, the Product Owner gathers this feedback, which should be clear and actionable. High-quality feedback allows teams to prioritize effectively, while unclear feedback can lead to confusion and rework. Timeliness is also vital; delays in feedback can slow progress and disrupt development (Haseli et al., 2023).

2.3. Relationship Between Customer Satisfaction and Scrum Software Development Project Success

Customer satisfaction is a central pillar of agile methodologies, and specifically Scrum, where the goal is to deliver value to the customer incrementally and iteratively. Scrum emphasizes continuous feedback, close collaboration with customers, and adaptability to ensure that the final product meets or exceeds customer expectations. In this context, customer satisfaction is closely linked to the success of Scrum software development projects, as it is an indicator of whether the product aligns with customer needs, preferences, and business goals.

This discussion critically explores the relationship between customer satisfaction and Scrum project success, examining both the direct and indirect factors that influence this relationship. While Scrum inherently fosters an environment that should enhance customer satisfaction, the dynamics of this relationship are complex and shaped by various internal and external factors.

2.3.1. Defining Customer Satisfaction in the Context of Scrum

Customer satisfaction in the Scrum framework is often viewed as the degree to which the product meets the expectations and requirements of the customer. In Scrum, the customer is typically represented by the Product Owner, who works closely with the development team to ensure that customer feedback is incorporated into the product through the management of the product backlog. As (Ekechi et al., 2024) emphasize, Scrum is designed to ensure that development efforts align with customer needs through regular interactions and frequent increments.

However, defining customer satisfaction in Scrum is multifaceted. It is not just about delivering the features the customer requested; it also includes how well the team engages with the customer, communicates progress, and addresses any concerns that arise during development. In this sense, customer satisfaction is both a product and a process outcome—satisfaction depends not only on the final product but also on how well the project progresses and whether the customer feels engaged, heard, and valued throughout the development cycle.

2.3.2. Challenges in Achieving Customer Satisfaction in Scrum Projects

Despite the theoretical benefits, achieving customer satisfaction in Scrum projects is not without its challenges. A central issue is changing requirements, which can either be a result of evolving market conditions or customer indecision. Scrum's flexibility to accommodate changes is a strength, but it can also lead to uncertainty and frustration for customers if their expectations are not managed properly. Scope creep often occurs due to the continuous influx of new customer feedback. If this feedback is not effectively managed or filtered, it can result in the addition of unnecessary features or frequent revisions, which may delay the project and diminish customer satisfaction (Aizaz et al., 2021).

2.3.3. Measuring Customer Satisfaction

Measuring customer satisfaction in Scrum projects can be difficult because the iterative nature of Scrum means that feedback is constantly changing, and satisfaction levels may fluctuate throughout the development process. Traditional methods of measuring customer satisfaction—such as surveys or post-delivery assessments—are often inadequate for agile projects, where customer feedback is integrated continuously.

(Eboh, 2024) suggest that customer satisfaction in Scrum should be evaluated through continuous engagement and retrospective reviews, where the Product Owner and team reflect on the customer's feedback at the end of each sprint. Other methods, such as regular surveys or customer interviews, can also provide valuable insights into satisfaction levels. However, the most important measure of satisfaction in Scrum is whether the customer perceives that the product delivers value and solves their problems.

3. Process factors

3.1. Relationship Between Project Management Process and Scrum Software Development Project Success

The relationship between the project management process and Scrum software development project success is a critical area of study within the context of agile methodologies. Scrum, a widely adopted agile framework, offers a flexible and iterative approach to managing software projects. In Scrum, the project management process is characterized by predefined roles, events, and artifacts that guide the team in delivering increments of the product. Understanding how these processes influence project success is crucial for evaluating the effectiveness of Scrum in meeting project objectives, especially regarding efficiency, adaptability, and stakeholder satisfaction.

3.1.1. Defining the Project Management Process in Scrum

In traditional project management methodologies, such as Waterfall, the project management process often follows a linear progression of steps—from planning and executing to closing the project. However, Scrum departs from this structured, sequential approach by embracing an iterative process focused on flexibility, collaboration, and frequent delivery of working software. The project management process in Scrum is shaped by several key elements, including:

- **Roles:** Scrum defines three core roles—Product Owner, Scrum Master, and Development Team. These roles ensure that the project is effectively managed through clear responsibilities and accountability (Kadenic, Koumaditis, et al., 2023; Niva, 2022; Petrescu et al., 2023).
- **Events:** Scrum specifies a series of recurring events or ceremonies, such as Sprint Planning, Daily Scrum, Sprint Review, and Sprint Retrospective, all of which facilitate coordination, monitoring, and reflection throughout the development process (Hron & Obwegeser, 2022; Patrucco et al., 2022; Przybyłek et al., 2022).
- **Artifacts:** Scrum's artifacts—Product Backlog, Sprint Backlog, and Increment—serve as tangible representations of the project's progress and future work, providing transparency and ensuring alignment between team members and stakeholders (Ekechi et al., 2024).

3.1.2. Iterative Planning and Delivery

One of the fundamental features of Scrum is its iterative planning and delivery process, where work is broken down into small, manageable chunks called sprints. Each sprint lasts typically between one and four weeks, during which the Scrum team works on a subset of features defined in the Product Backlog. This iterative process allows teams to deliver tangible increments of the product regularly, providing stakeholders with working software that can be assessed and reviewed for quality and functionality. This incremental approach is a significant departure from traditional methods, where a project progresses through a series of phases, often with little stakeholder involvement until the final delivery (Schwaber, 1997).

3.1.3. *Collaboration and Communication*

Another cornerstone of Scrum's project management process is collaboration and communication among team members, stakeholders, and customers. Scrum facilitates collaboration through frequent communication at various events, such as the Daily Scrum, Sprint Review, and Sprint Retrospective. These events allow the team to discuss progress, share concerns, identify obstacles, and reflect on ways to improve their processes. The open and transparent nature of Scrum encourages collaboration not only within the Scrum team but also with external stakeholders, such as the customer or Product Owner, who provide continuous feedback on the product's development (Holzmann & Panizel, 2013).

3.1.4. Transparency and Continuous Monitoring

Scrum places a significant emphasis on transparency, both within the development team and with external stakeholders. The use of Scrum artifacts, such as the Sprint Backlog and Burndown Charts, provides visibility into the work completed and work remaining, helping to monitor progress and manage expectations. Transparency enables teams to identify potential risks or delays early, allowing for timely intervention to keep the project on track. The role of the Scrum Master is particularly important in ensuring that impediments are removed and that the team stays focused on delivering value (Sokolowski, 2022).

3.1.5. Flexibility and Adaptability in Project Management

Scrum's iterative process and commitment to responding to change are crucial for managing projects in dynamic environments where requirements frequently evolve. The ability to adapt to change is facilitated by Scrum's lightweight processes, such as the Sprint Retrospective, which allows teams to reflect on what went well and what could be improved. These retrospectives create opportunities for continuous improvement, enabling teams to adjust their approach based on previous experiences(Popova, 2019).

3.1.6. Risk Management and Decision Making

Risk management is another critical aspect of project management in Scrum (Ali et al., 2016). Scrum's approach to managing risks is inherently proactive, as risks are continuously identified and addressed throughout each sprint cycle. The use of the Sprint Planning event helps teams assess potential risks and uncertainties early, while regular Daily Scrums provide opportunities to detect and mitigate emerging issues.

3.2. Relationship Between Project Definition Process and Scrum Software Development Project Success

The project definition process is crucial for the success of software development initiatives. In Scrum, an agile framework focused on iterative development, the definition phase must align with principles of flexibility, collaboration, and transparency. Unlike traditional project management, Scrum encourages early stakeholder involvement, evolving requirements, and continuous feedback, making effective project definition essential for aligning with user needs and business goals (Dvir et al., 2003).

3.2.1. The Role of Project Definition in Scrum

In traditional frameworks, project definition occurs during the Initiation phase, detailing scope, objectives, and resources. In contrast, scrum treats project definition as a dynamic process, evolving through ongoing collaboration with stakeholders. Key elements include:

- **Product Vision and Backlog**: The Product Owner articulates the vision and maintains a dynamic product backlog, which evolves based on stakeholder feedback during sprints (Balasubramaniyam, 2024).
- **Sprint Goals and Backlogs**: Each sprint begins with defined goals, ensuring continuous progress toward objectives (Schwaber & Sutherland, 2011).
- **Stakeholder Involvement**: Regular interactions during Sprint Reviews allow for reassessment of priorities based on emerging needs (Kathram & Nersu, 2024).

4. Technical factors

4.1. Relationship Between Agile Software Techniques and Scrum Software Development Project Success

Agile software techniques form the backbone of the Scrum framework, fostering flexibility, collaboration, and continuous delivery in software development projects. The relationship between agile techniques and the success of Scrum software development projects is integral, as these techniques directly influence the project's efficiency, adaptability, and ability to meet customer needs. This discussion critically examines the role of agile techniques in Scrum and their impact on project success.

One of the core agile techniques used within Scrum is iterative development, which enables teams to deliver small, incremental updates to the product at the end of each sprint. This method is particularly effective in addressing changing requirements and enabling frequent feedback from stakeholders, ensuring that the product aligns with customer expectations (Morin, 2020). The iterative process helps to manage uncertainty and mitigate risks by breaking the project into manageable segments, allowing teams to adjust to evolving needs.

Another essential agile technique in Scrum is collaborative decision-making, which is fostered through daily stand-ups, sprint planning, and sprint reviews. This collaboration ensures that the development team and stakeholders are aligned and can make real-time adjustments based on feedback, thus driving the success of the project (Moe et al., 2012; Ștefan et al., 2022). The emphasis on transparency and communication is crucial for addressing issues early, avoiding costly mistakes, and maintaining stakeholder engagement throughout the project lifecycle.

5. Project factors

5.1. Relationship between non-life-critical projects and scrum software development project success

Scrum's flexibility and adaptability make it particularly effective for non-life-critical projects. In such settings, where failure does not have severe consequences, Scrum's iterative approach allows for experimentation and continuous feedback. This environment enables teams to focus on high-quality software delivery while managing risks through frequent evaluations. However, without urgent deadlines, there is a potential for scope creep, which can hinder project success if not managed effectively (Kulathunga & Ratiyala, 2018).

5.2. Relationship between projects of variable scope and scrum software development project success

Scrum's flexibility makes it well-suited for projects with variable scope. Its iterative process allows for continuous reevaluation of priorities, ensuring alignment with evolving customer needs. The dynamic nature of the product backlog supports efficient prioritization as requirements shift. However, constant scope changes can introduce scope creep, requiring careful management to avoid delays and maintain focus (Kulathunga & Ratiyala, 2018).

5.3. Relationship between dynamic schedules and scrum software development project success

Scrum's ability to adapt to changing timelines makes it ideal for managing projects with dynamic schedules. Its time-boxed sprints create predictability and allow teams to focus on incremental improvements, even as priorities shift. Regular reviews and retrospectives ensure alignment with evolving goals, but dynamic schedules pose risks such as misalignment or scope creep, which require proactive communication and stakeholder engagement(Kulathunga & Ratiyala, 2018).

5.4. Relationship between project acceptability and scrum software development project success

Project acceptability, defined by meeting stakeholder expectations, is a critical success factor in Scrum. The framework's iterative feedback loops and ongoing customer involvement help address issues early, ensuring that the final product aligns with stakeholder needs. Scrum's adaptability to feedback enhances acceptability, but failing to manage expectations effectively can lead to product rejection and project failure (Kulathunga & Ratiyala, 2018).

5.5. Relationship between intention to use and scrum software development project success

The success of Scrum projects is significantly influenced by stakeholders' intention to adopt Scrum practices. A strong commitment to Scrum's principles of collaboration, transparency, and iterative development leads to improved outcomes. A clear intention to use Scrum fosters a culture of continuous improvement, but resistance to Scrum can result in delays or ineffective implementation (Kulathunga & Ratiyala, 2018).

Conclusion

This comprehensive literature review has synthesized and categorized the Critical Success Factors influencing the successful implementation of Scrum in software development projects. The study highlights the crucial roles of organizational factors such as management commitment, team environment, and customer involvement, alongside process and technical factors like agile software techniques and delivery strategies. It also underscores the importance of flexibility and adaptability in managing variable scope and dynamic schedules inherent in Scrum projects. However, challenges such as scope creep and misalignment between stakeholders must be carefully managed to sustain project success. This review provides actionable insights for both researchers and practitioners, offering a foundation for improving Scrum practices and enhancing project outcomes through better understanding and application of these critical factors. Further research is recommended to explore how these factors interact in different organizational contexts and project types to build a more robust framework for Scrum success.

References:

Abadir, S., Batsa, E., Neubert, M., & Halkias, D. (2019). Leading multicultural teams in agile organizations. Available at SSRN 3507635.

Adeniyi, I. S., Al Hamad, N. M., Adewusi, O. E., Unachukwu, C. C., Osawaru, B., Onyebuchi, C. N., Omolawal, S. A., Aliu, A. O., & David, I. O. (2024). Organizational culture and leadership development: A human resources review of trends and best practices.

Ahmed, P. K., Loh, A. Y., & Zairi, M. (1999). Cultures for continuous improvement and learning. Total Quality Management, 10(4-5), 426-434.

Aizaz, F., Khan, S. U. R., Khan, J. A., & Akhunzada, A. (2021). An empirical investigation of factors causing scope creep in agile global software development context: a conceptual model for project managers. IEEE Access, 9, 109166-109195.

Al-Ashmoery, Y., Nasser, N., Chaabi, Y., Haider, H., Haider, A., & Alwesabi, K. (2023). A systematic study on Traditional software development models and Agile Software Development Methodologies. Alrazi University Journal of Computer Science and Technology, 1(1), 1-13.

Alami, A., Zahedi, M., & Krancher, O. (2023). How does psychological safety influence agile teams' pursuit of achieving software quality?

Ali, M., Hafeez, Y., & Hamid, B. (2016). An Empirical Study and a Framework for Effective Risk Management in Scrum: Effective Risk Management in Scrum. Proceedings of the Pakistan Academy of Sciences: A. Physical and Computational Sciences, 53(4), 417-429.

Almeida, F., & Espinheira, E. (2022). Adoption of large-scale scrum practices through the use of management 3.0. Informatics,

Alqudah, M. K., Razali, R., Alqudah, M. K., Al Dalaien, M. N., Alabool, H. M., & Alkhazaleh, H. A. (2024). A grounded theory of selecting lean and agile practices for software development. Journal of Software: Evolution and Process, 36(4), e2539.

ALSHURIDEH, M., Akour, I. A., Al Kurdi, B., & Hamadneh, S. (2024). Exploring the Impact of Scrum Framework on Project Effectiveness: A Quantitative Analysis of Agile Implementation Challenges and Benefits. International Journal of Theory of Organization and Practice (IJTOP), 4(2), 161-174.

Ames Zegarra, C., & Sabanovic, S. (2022). Cross-functional team in a project with an agile management: Effective strategies and common challenges in a cross-functional team while using agile project management. In.

Andrei, N. (2024). CROSS-FUNCTIONAL TEAMS AS A KEY TO SUCCESSFUL PRODUCT MANAGEMENT. Universum: технические науки, 9(4 (121)), 24-27.

Arif, M. (2024). Agile Management: Adapting to Change in Dynamic Markets. Governance Accounting Archive Review, 1(4), 145-161.

Balasubramaniyam, A. (2024). Role of Product Owners in incremental adoption of new IT operating model.

Brose, W. D., Cabral, P. M. F., Junior, J. C. d. S. F., & de David, C. (2023). Scrum team self-organization: an understanding in the light of the systemic-complex paradigm. International Journal of Scientific Management and Tourism, 9(6), 3594-3619.

Chow, T., & Cao, D.-B. (2008). A survey study of critical success factors in agile software projects. Journal of Systems and Software, 81(6), 961-971.

Coleman, D. M., Dossett, L. A., & Dimick, J. B. (2021). Building high performing teams: opportunities and challenges of inclusive recruitment practices. Journal of Vascular Surgery, 74(2), 86S-92S.

Deci, E. L., & Ryan, R. M. (2012). Self-determination theory. Handbook of theories of social psychology, 1(20), 416-436.

Doan, T. T. T., Nguyen, L. C. T., & Nguyen, T. D. N. (2020). Emotional intelligence and project success: The roles of transformational leadership and organizational commitment. The Journal of Asian Finance, Economics and Business, 7(3), 223-233.

Dvir, D., Raz, T., & Shenhar, A. J. (2003). An empirical analysis of the relationship between project planning and project success. International Journal of Project Management, 21(2), 89-95.

Dwi Harfianto, H., Raharjo, T., Hardian, B., & Wahbi, A. (2022). Agile transformation challenges and solutions in bureaucratic government: a systematic literature review. Proceedings of the 2022 5th International Conference on Computers in Management and Business,

Eboh, M. F. (2024). Effective customer engagement strategies for agile project management Vilniaus universitetas.].

Ekechi, C. C., Okeke, C. D., & Adama, H. E. (2024). Enhancing agile product development with scrum methodologies: A detailed exploration of implementation practices and benefits. Engineering Science & Technology Journal, 5(5), 1542-1570.

Elenurm, T., & Fabritius, J. (2023). Managing task and relationship conflicts in international online team learning. Innovations in Education and Teaching International, 60(3), 426-435.

Ewim, C., Achumie, G. O., Adeleke, A. G., Okeke, I. C., & Mokogwu, C. (2024). Developing a cross-functional team coordination framework: A model for optimizing business operations. International Journal of Frontline Research in Multidisciplinary Studies, 4(01), 15-34.

Gmach, D., Rolia, J., Cherkasova, L., & Kemper, A. (2009). Resource pool management: Reactive versus proactive or let's be friends. Computer Networks, 53(17), 2905-2922.

Goins, P. (2024). Understanding the Transparency Construct of the Project Management Scrum Framework in Continuous Improvement Projects in the Manufacturing Industry Capella University].

Handri, E. Y., Sensuse, D. I., & Tarigan, A. (2024). Developing an agile cybersecurity framework with organizational culture approach using Q methodology. IEEE Access.

Haseli, G., Ranjbarzadeh, R., Hajiaghaei-Keshteli, M., Ghoushchi, S. J., Hasani, A., Deveci, M., & Ding, W. (2023). HECON: Weight assessment of the product loyalty criteria considering the customer decision's halo effect using the convolutional neural networks. Information Sciences, 623, 184-205.

Hoda, R., Noble, J., & Marshall, S. (2010). Organizing self-organizing teams. Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1,

Holland, S., Gaston, K., & Gomes, J. (2000). Critical success factors for cross-functional teamwork in new product development. International journal of management reviews, 2(3), 231-259.

Holzmann, V., & Panizel, I. (2013). Communications management in Scrum projects. International Conference on Information Management and Evaluation,

Hron, M., & Obwegeser, N. (2022). Why and how is Scrum being adapted in practice: A systematic review. Journal of Systems and Software, 183, 111110.

Jain, A. K., Fennell, M. L., Chagpar, A. B., Connolly, H. K., & Nembhard, I. M. (2016). Moving toward improved teamwork in cancer care: the role of psychological safety in team communication. Journal of oncology practice, 12(11), 1000-1011.

Jeleel-Ojuade, A. (2024). The Role of Information Silos: An analysis of how the categorization of information creates silos within financial institutions, hindering *effective communication and collaboration. Available at SSRN 4881342.*

Jerab, D., & Mabrouk, T. (2023). The role of leadership in changing organizational culture. Available at SSRN 4574324.

Jones, M. (2010). Leadership's role in shaping organizational culture: the key to the future. The opinions and statements expressed throughout this volume are those of the individual authors and contributors and should not be considered an endorsement or a reflection of the official position of the Federal Bureau of Investigation, the Society of Police Futures International, or any other institution or organization for any policy, 53.

Jovanović, M., Mas, A., Mesquida, A.-L., & Lalić, B. (2017). Transition of organizational roles in Agile transformation process: A grounded theory approach. Journal of Systems and Software, 133, 174-194.

Junior, A., & Aquino, P. T. (2024). Critical Success Factors for Agile Software Development. IEEE Transactions on Engineering Management.

Kadenic, M. D., de Jesus Pacheco, D. A., Koumaditis, K., Tjørnehøj, G., & Tambo, T. (2023). Investigating the role of Product Owner in Scrum teams: Differentiation between organisational and individual impacts and opportunities. Journal of Systems and Software, 206, 111841.

Kadenic, M. D., Koumaditis, K., & Junker-Jensen, L. (2023). Mastering scrum with a focus on team maturity and key components of scrum. Information and software technology, 153, 107079.

Kakar, A. K., & Kakar, A. (2023). Have the Agile Principles endured? An empirical investigation post 20th anniversary of the Agile Manifesto (2001).

Kalluri, R. (2022). A Human Factors Study of Risk Management of Complex Agile Scrum Projects in Large Enterprises. International Journal of Business & Management Studies, 3(8).

Kathram, S. R., & Nersu, S. R. K. (2024). Enhancing Stakeholder Engagement through Agile Project Transparency: A Roadmap for Modern Project Managers. Revista de Inteligencia Artificial en Medicina, 15(1), 1358-1389.

Kostin, D., & Strode, D. (2022). Effective communication in globally distributed Scrum teams.

Kulathunga, K., & Ratiyala, S. (2018). Key success factors of scrum software development methodology in Sri Lanka.

Landaeta, R. E., Viscardi, S., & Tolk, A. (2011). Strategic management of scrum projects: An organizational learning perspective. First International Technology Management Conference,

Larusdottir, M., Cajander, Å., Gregory, P., Cockton, G., Salah, D., Kuusinen, K., & Nauwerck, G. (2016). Stakeholder involvement in agile software development. Proceedings of the 9th Nordic Conference on Human-Computer Interaction,

Liu, J.-W., Ho, C.-Y., Chang, J. Y., & Tsai, J. C.-A. (2019). The role of Sprint planning and feedback in game development projects: Implications for game quality. Journal of Systems and Software, 154, 79-91.

Luca, C. (2022). Hybrid Methodologies: Integrating Waterfall and Agile Approaches.

Luna, A. J. d. O., Kruchten, P., Pedrosa, M. L. d. E., Neto, H. R., & De Moura, H. P. (2014). State of the art of agile governance: a systematic review. arXiv preprint arXiv:1411.1922.

Majka, M. (2024). Common Challenges for New Scrum Masters.

Moe, N. B., Aurum, A., & Dybå, T. (2012). Challenges of shared decision-making: A multiple case study of agile software development. Information and software technology, 54(8), 853-865.

Moe, N. B., Dingsøyr, T., & Dybå, T. (2008). Understanding self-organizing teams in agile software development. 19th australian conference on software engineering (aswec 2008),

Moe, N. B., Dingsøyr, T., & Dybå, T. (2010). A teamwork model for understanding an agile team: A case study of a Scrum project. Information and software technology, 52(5), 480-491.

Mogård, E. V., Rørstad, O. B., & Bang, H. (2022). The relationship between psychological safety and management team effectiveness: the mediating role of behavioral integration. International journal of environmental research and public health, 20(1), 406.

Morin, J. (2020). Scrum Iterative Development Practices for Team-Based Active Learning. Journal of Faculty Development, 34(3), 96-99.

Natarajan, T., & Pichai, S. (2024). Transition from Waterfall to Agile Methodology-An Action Research Study. IEEE Access.

Niva, P. (2022). Freedom in the role of a Product Owner: a case study of the role and requirements of a Product Owner in a large-scale agile environment.

Patrucco, A. S., Canterino, F., & Minelgaite, I. (2022). How do scrum methodologies influence the team's cultural values? A multiple case study on agile teams in Nonsoftware industries. IEEE Transactions on Engineering Management, 69(6), 3503-3513.

Petrescu, M. A., Motogna, S., & Berciu, L. (2023). Women in Scrum Master Role: Challenges and Opportunities. 2023 IEEE/ACM 4th Workshop on Gender Equity, Diversity, and Inclusion in Software Engineering (GEICSE),

Pinto, J. K., Davis, K., Ika, L. A., Jugdev, K., & Zwikael, O. (2022). Coming to terms with project success: Current perspectives and future challenges. International Journal of Project Management, 40(7), 831-834.

Pop, M. C. (2022). Agile Virtualization: the importance of Scrum framework in creating synergies in global organizations.

Popova, O. (2019). Adaptation of flexible project management models based on Scrum and Kanban technologies. Технологический аудит и резервы производства(4 (2)), 4-10.

Przybyłek, A., Albecka, M., Springer, O., & Kowalski, W. (2022). Game-based Sprint retrospectives: multiple action research. Empirical Software Engineering, 27(1), 1.

Salin, H., Albrecht, F., & Skov, J. (2023). The Power of Scrum Mastery: An Analysis of Agile Team Performance and Scrum Master Influence. ICSOFT,

Schwaber, K. (1997). Scrum development process. Business Object Design and Implementation: OOPSLA'95 Workshop Proceedings 16 October 1995, Austin, Texas,

Schwaber, K., & Sutherland, J. (2011). The scrum guide. Scrum Alliance, 21(1), 1-38.

Shakir, M., Jusoh, M., Azam, S. F., Shakir, M., Jusoh, M., & Azam, S. F. (2023). THE RIPPLE EFFECT: SILO MENTALITY'S INFLUENCE ON COMMUNICATION AND COLLABORATION PATTERNS IN THE IT INDUSTRY OF SRI LANKA. Journal of Data Acquisition and Processing, 38(4), 886.

Sharp, H., Giuffrida, R., & Melnik, G. (2012). Information flow within a dispersed agile team: a distributed cognition perspective. Agile Processes in Software Engineering and Extreme Programming: 13th International Conference, XP 2012, Malmö, Sweden, May 21-25, 2012. Proceedings 13,

Sokolowski, L. (2022). Investigating transparency in collaborative learning and its delivery through Scrum University of West London].

Ştefan, I. A., Ştefan, A., Tsalapatas, H., Heidmann, O., & Gheorghe, A. F. (2022). Collaborative decision-making in software research projects: the innovation challenge. Procedia Computer Science, 199, 1318-1326.

Stettina, C. J., & Hörz, J. (2015). Agile portfolio management: An empirical perspective on the practice in use. International Journal of Project Management, 33(1), 140-152.

Sutrisno, S., Kuraesin, A. D., Siminto, S., Irawansyah, I., & Ausat, A. M. A. (2023). The Role of Information Technology in Driving Innovation and Entrepreneurial Business Growth. Jurnal Minfo Polgan, 12(1), 586-597.

Tutty, J. I., & Klein, J. D. (2008). Computer-mediated instruction: A comparison of online and face-to-face collaboration. Educational technology research and development, 56, 101-124.

Tyagi, S., Sibal, R., & Suri, B. (2022). Empirically developed framework for building trust in distributed agile teams. Information and software technology, 145, 106828.

van Greunen, C., Venter, E., & Sharp, G. (2021). The influence of relationship and task conflict on the knowledge-sharing intention in knowledge-intensive organisations. South African Journal of Business Management, 52(1), 9.

Varma, V. S., & Gupta, R. CONFLICT RESOLUTION STRATEGIES IN THE WORKPLACE: EMPIRICAL STUDY OF MANAGING INTERPERSONAL AND TEAM CONFLICTS.

Weichbroth, P. (2022). A case study on implementing agile techniques and practices: Rationale, benefits, barriers and business implications for hardware development. Applied Sciences, 12(17), 8457.

Zeng, J., Wen, J., Cai, B., & Xiao, Q. (2024). Research on Collaborative Innovation Ability Training of Software Engineering Talents Based on the Industry-Education Integration. 2024 IEEE International Conference on Software Services Engineering (SSE),

Contents

الملخص 2
INTRODUCTION 2
IMPORTANCE OF IDENTIFYING CRITICAL SUCCESS FACTORS (CSFS) 3
RESEARCH PROBLEM AND RATIONALE 3
RESEARCH AIMS AND OBJECTIVES 4
LITERATURE REVIEW 4
METHODOLOGY 6
1. SEARCH STRATEGY 6
2. INCLUSION CRITERIA 6
3. EXCLUSION CRITERIA 6
ANALYSIS OF CRITICAL SUCCESS FACTORS 6
1. ORGANIZATION FACTORS 6
2. PEOPLE FACTORS 13
3. PROCESS FACTORS 16
4. TECHNICAL FACTORS 18
5. PROJECT FACTORS 18
CONCLUSION 19
REFERENCES: 19
CONTENTS 22