ISSN: 2634 - 8853

Journal of Engineering and Applied Sciences Technology

SCIENTIFIC Research and Community

Review Article

Open Access

The Future of Scrum in AI-Driven Software Development: A Systematic Review

Gazi Sabbir Ahammad[1], Sujan Acharya[1], Ahnaf Aiman Abdi[2] and Golam Mahadi[3*]

[1]Bachelor of Software Engineering, School of Computer Science and Artificial Intelligence, Zhengzhou University, Henan, China

[2]MSc. in Information and Communication Engineering, School of Resources and Environment, University of Electronic Science and Technology of China, Chengdu, Sichuan, China

[3]MSc. in Computer Science, School of Computer science and Technology, Dalian University of Technology, Dalian, Liaoning, China

ABSTRACT

Scrum has become one of the most widely used Agile frameworks in software development because of its iterative structure, transparency, and ability to support continuous delivery in changing environments. At the same time, Artificial Intelligence (AI) is reshaping how software teams plan, coordinate, and execute development work. This systematic review examines the future of Scrum in AI-driven software development by analyzing its core principles, workflow, strengths, limitations, and emerging opportunities. The review synthesizes literature from academic databases and credible industry sources to identify how Scrum is being adapted in response to AI-enabled automation, predictive analytics, intelligent task allocation, communication support, and adaptive planning. The findings show that Scrum remains highly relevant for managing complex and evolving software projects, but its effectiveness depends on strong collaboration, disciplined execution, and organizational support. The review also highlights several challenges, including scope creep, scalability in distributed teams, resistance to change, and ethical concerns related to AI bias and human oversight. Overall, the study concludes that AI is unlikely to replace Scrum, but it is likely to reshape Scrum practices by improving efficiency, decision-making, and coordination while preserving the need for human accountability. The paper further identifies future directions such as AI-assisted backlog management, scalable Agile frameworks, responsible AI governance, and cross-domain adoption of Scrum beyond software engineering.

***Corresponding author**

Golam Mahadi, M.Sc in Computer Science, School of Computer science and Technology, Dalian University of Technology, Dalian, Liaoning, China.

Received: June 10, 2026; **Accepted:** June 16, 2026; **Published:** June 24, 2026

Keywords: Scrum, Agile Methodology, Artificial Intelligence, Software Development, Systematic Review, Distributed Teams, Human Oversight, Project Management

Introduction

Scrum, an Agile methodology developed by Jeff Sutherland and Ken Schwaber in 1993, has revolutionized software development by emphasizing iterative, incremental progress, collaboration, and adaptability [1-4]. Inspired by rugby's cooperative "scrum," it fosters team collaboration to deliver incremental value aligned with stakeholder needs [5-7]. Over time, Scrum has transitioned from niche use to widespread adoption across industries, effectively managing software development complexity through sprints, enabling frequent releases, feedback integration, and adaptability to changing requirements [8-9].

However, Scrum faces challenges in adapting to advancements in artificial intelligence (AI). The rapid pace of AI innovation, increasing software complexity, and demands for scalability raise questions about Scrum's responsiveness. While AI tools promise enhanced efficiency and decision-making, they necessitate updates to Scrum's processes, roles, and responsibilities [10-12]. Additionally, scaling Scrum for large, distributed teams remains problematic, exacerbated by remote work and global collaboration challenges. Iterative processes, while beneficial, can lead to scope creep and delays in large, complex projects [13-15]. These issues highlight the need for Scrum's evolution to remain effective in modern software development landscapes.

This review explores the future trajectory of Scrum in the era of artificial intelligence (AI) by addressing key research questions:

- Relevance of Scrum: Why does Scrum remain pertinent in software development despite competing methodologies? [1-3].
- Advantages of Scrum: What specific benefits make Scrum preferable over other Agile approaches and traditional project management techniques? [4-6].
- Weaknesses of Scrum: What are Scrum's inherent limitations, and how do they impact productivity and project success? [7-9].
- Enhancements for Competitiveness: How can Scrum be adapted and improved to remain competitive, particularly through AI-driven tools and methodologies? [10-12].

By addressing these questions, this article examines Scrum's potential to evolve alongside technological advancements, ensuring its continued relevance in software development.

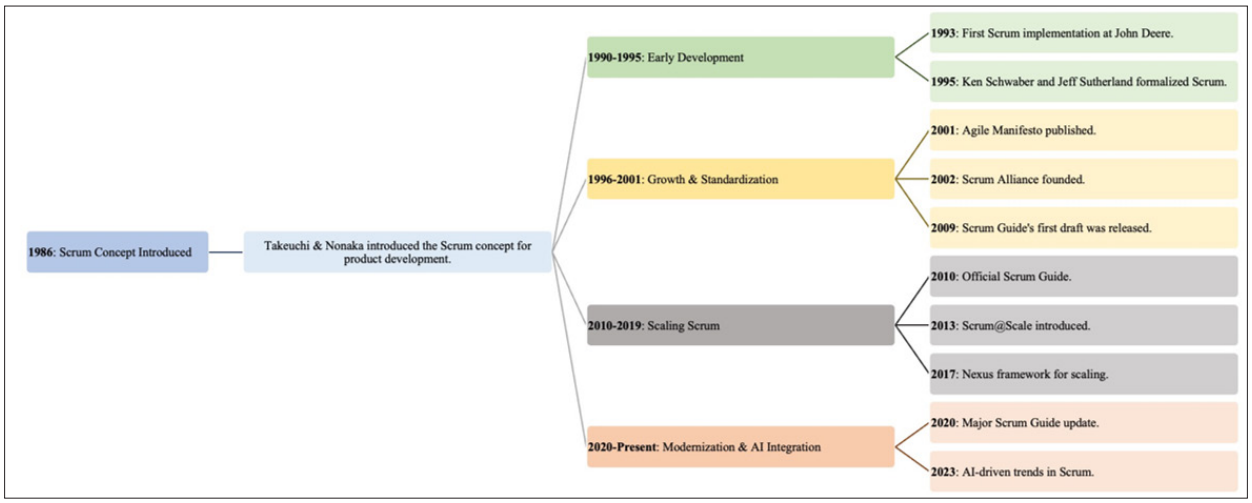

Figure 1: Scrum Evolution

This review evaluates Scrum's applicability in modern software development and its future evolution in the context of artificial intelligence (AI). By systematically analyzing its principles, workflows, and artifacts, the study highlights Scrum's enduring relevance in addressing software development complexity [1-3]. It also examines the advantages driving Scrum's widespread adoption, alongside its limitations and their impact on project management [4-6]. Furthermore, the review explores the transformative potential of integrating AI into Scrum to enhance collaboration, scalability, and innovation in Agile practices [7-9].

Focusing on software engineering, this research outlines Scrum's core principles, processes, and artifacts, critically assessing its strengths and weaknesses in contemporary development environments [10-11]. It also investigates how AI can address Scrum's inherent challenges, offering insights into the synergy between Agile frameworks and AI-driven automation [12-13]. The study concludes with future research directions and practical applications of Scrum in AI, emphasizing innovation and agility enhancement in software development [14-15].

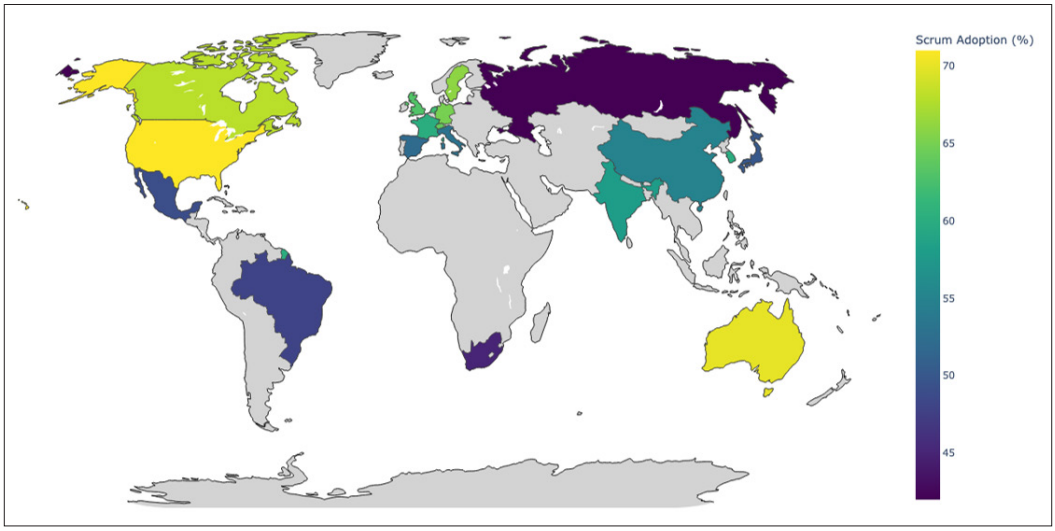

Figure 2: Global Scrum Adoption by Country

This visualization presents the global adoption of the Scrum framework, a key Agile methodology, across various countries. The choropleth map shows the percentage of organizations implementing Scrum, with color intensities indicating the level of adoption. Regions like the United States, Canada, and Australia demonstrate high adoption rates while emerging markets like India and China are experiencing rapid growth in Scrum implementation. The data is sourced from industry reports, specifically the Scrum Alliance and VersionOne State of Agile Report [79-81]. This map provides valuable insights for academic studies on the diffusion of Agile practices across different regions, highlighting trends and adoption barriers.

Literature Review

The origins of Scrum date back to the early 1980s, when Hirotaka Takeuchi and Ikujiro Nonaka introduced the "rugby-style" product development concept in their Harvard Business Review article, The New New Product Development Game [3-16]. They observed that companies like Honda and Canon achieved success through cross-functional, team-based, and iterative approaches, emphasizing flexibility, empowerment, and rapid iteration. This methodology, likened to a rugby scrum, laid the foundation for the Scrum framework [17].

Criteria	Scrum	Kanban	XP
Focus	Iterative development, team collaboration, adaptability.	Continuous flow, work-in-progress optimization.	Continuous delivery, code quality, customer feedback.
Principles	Transparency, Inspection, Adaptation.	Visualizing work, limiting WIP, flow management.	Communication, simplicity, feedback, courage.
Structure	Time-boxed sprints, Scrum events.	No time-boxed iterations, focuses on continuous flow.	Iterations with short feedback loops, pair programming, TDD.
Roles	Product Owner, Scrum Master, Development Team.	No defined roles.	Customer, Developer, Tester.
Artifacts	Product Backlog, Sprint Backlog, Increment.	Kanban Board (To Do, In Progress, Done).	User Stories, Iteration Backlog, Codebase.
Flexibility	Less flexible due to time-boxed sprints.	High flexibility, continuous flow.	Flexible within iteration cycles, focus on technical practices.
Feedback	Daily Scrums, Sprint Reviews, Retrospectives.	Continuous feedback via board updates.	Continuous feedback with frequent releases.
Adoption	Widely adopted in software and other industries.	Used in service and manufacturing.	Primarily in software, emphasizing technical excellence.
Strengths	Collaboration, adaptive planning, transparency.	Flexibility, visual management, flow optimization.	Quality, testing, customer collaboration.
Challenges	Requires dedicated roles, time-boxed sprints.	Can lack structure or direction.	High technical demands, risk of developer burnout.

Figure 3: Comparison of Scrum with Other Agile Methodologies

In 1993, Jeff Sutherland and Ken Schwaber formalized Scrum as a software development framework, inspired by Takeuchi and Nonaka's work, to address the inefficiencies of traditional waterfall methodologies [6,18]. Scrum enabled teams to tackle complex problems by breaking them into smaller tasks and delivering incremental value through iterative cycles [7]. Its principles were codified in the Scrum Guide, first published in 2010 and regularly updated [8].

Scrum is built on three pillars: transparency, inspection, and adaptation, which underpin its empirical process control [9,19]. Transparency ensures visibility of work and progress through artifacts like the product backlog, sprint backlog, and increment [11, 19]. Inspection involves regular progress assessments via Scrum events (e.g., daily scrums, sprint reviews, retrospectives) to identify challenges and improvements [3,12]. Adaptation allows teams to respond swiftly to deviations or changing requirements, ensuring alignment with project goals and continuous value delivery [9,14]. These principles enable Scrum teams to navigate complexity and uncertainty effectively.

Scrum fosters a collaborative, transparent, and responsive development environment through its three pillars: transparency, inspection, and adaptation, and five core values: commitment, focus, openness, respect, and courage. These values empower teams to take ownership, communicate transparently, and achieve shared goals [8,15]. The Scrum workflow is structured around five key events:

- Sprint Planning: The team defines the sprint goal and selects tasks from the product backlog, ensuring alignment on objectives [20].
- Daily Scrum: A time-boxed daily meeting where team members share progress, plans, and obstacles, promoting transparency and alignment [8,21,25].
- Sprint Review: Held at the end of each sprint, this event involves demonstrating completed work to stakeholders and gathering feedback to ensure alignment with expectations [8,22].
- Sprint Retrospective: A reflective meeting where the team evaluates successes, challenges, and process improvements for the next sprint. This event fosters a culture of continuous learning and iterative enhancement [23].
- The Sprint: A time-boxed iteration, typically lasting two to four weeks, where the team delivers a potentially releasable product increment. It provides a structured yet flexible framework for managing complex projects [24,25].

These events enhance collaboration, transparency, and continuous improvement, enabling teams to deliver value iteratively. By adhering to these values and structured events, Scrum enables teams to adapt to change, maximize productivity, and continuously refine their development process to deliver high-quality software efficiently.

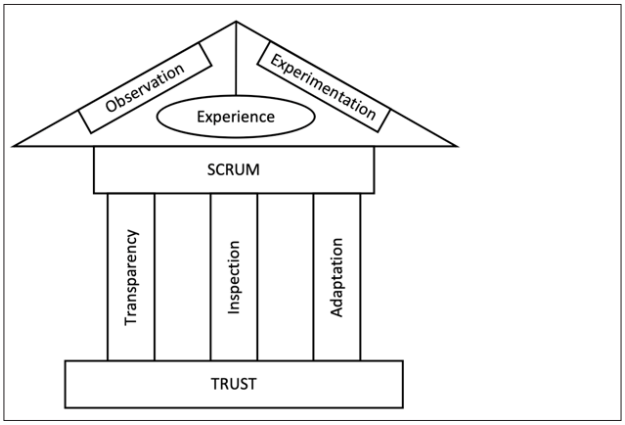

Figure 4: The Scrum Pillars & Characteristics (Modified from Scrum.org, n.d.)

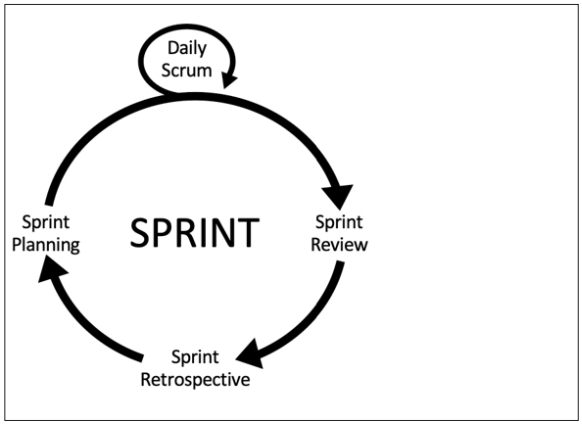

Figure 5: The Scrum Event and Sprint Cycle.

Workflow	Members	Time	Keywords
Sprint	SM, PO, DT	1 – 4 Week	• Development continuously cycle • Include all scrum events
Sprint Planning	SM, PO, DT	1 – 8 Hours	• Plan the sprint Backlog • Define the sprint Goal
Daily Scrum	DT, SM (opt)	15 Minute	• Reduce complexity • Synchronize Process
Sprint Review	SM, PO, DT	1 – 4 Hours	• Demonstrate the increment • Get feedback from Stakeholders
Sprint Retrospective	DT, SM (opt)	1 – 3 Hours	• Way of improvement • What was a mistake

Figure 6: The Scrum Workflow and Keywords

Scrum relies on three key artifacts to ensure transparency and alignment:

- Product Backlog: A prioritized, dynamic list of all work required to deliver the product, evolving as requirements and priorities change [27].
- Sprint Backlog: A subset of product backlog items the team commits to completing during the sprint, providing a clear plan to achieve the sprint goal [28].
- Increment: The sum of completed product backlog items at the end of a sprint, meeting the team's definition of done and being potentially releasable, ensuring incremental value delivery [8,29].

These artifacts support Scrum's iterative and transparent development process.

Product Backlog	Sprint Backlog
Focuses on the long-term goals of the project	Focuses on a short-term set of goals for a specific sprint
Owned by Product Owner	Owned by Development Team
Includes all features and functionalities planned for the product	Includes a subset of tasks from the product backlog chosen for the upcoming sprint
Items are prioritized but not scheduled for a specific time	Items are broken down into smaller tasks and estimated for time
Duration is the entire project	Duration is fixed for specific sprint, usually 2-4 weeks
Backlog can be change and adapted	During Sprint backlog can't change

Figure 7: Scrum Product Backlog vs Sprint Backlog

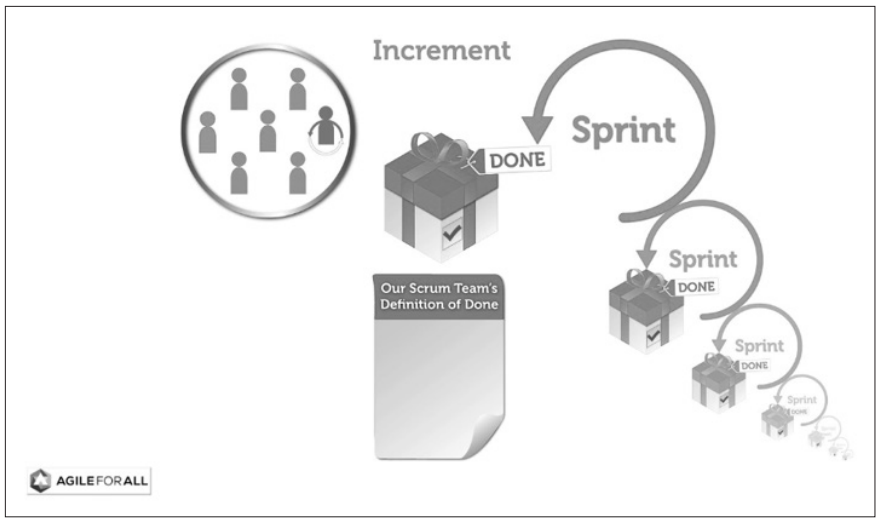

Figure 8: Visualization of the Increment and Done (Agile for All, 2017).

Scrum has expanded beyond software development into industries like healthcare, education, and manufacturing, leveraging its focus on collaboration, adaptability, and continuous improvement to manage complex projects [1,30].

A significant trend is the integration of Artificial Intelligence (AI) into Scrum workflows. AI-powered tools enhance backlog prioritization, task assignment, and progress tracking, using machine learning to predict bottlenecks and optimize processes [10,31,36]. AI also facilitates collaboration in distributed teams through real-time translation, sentiment analysis, and meeting summaries, bridging cultural and linguistic gaps [32,37,38]. Additionally, AI enables adaptive planning by analyzing market trends and customer feedback to dynamically adjust priorities [33,39].

However, integrating AI into Scrum poses challenges, including the need for teams to acquire new skills and adapt processes, representing a cultural shift for organizations [34, 40]. Ethical concerns, such as AI bias and the necessity of human oversight, also arise, ensuring decisions align with organizational values [35,41].

Methodology

A rigorous search strategy was employed to ensure a comprehensive review of Scrum and its evolution in the AI era. The review utilized academic and industry sources, including peer-reviewed journal articles, conference papers, books, and credible online publications. Primary databases such as IEEE Xplore, Scopus, Google Scholar, and PubMed were searched due to their extensive collections in software engineering, Agile methodologies, and AI [42].

Keywords and phrases included Scrum framework, Agile methodologies, Scrum in software development, Scrum and Artificial Intelligence, AI in Agile workflows, Scrum limitations and challenges, Scrum scalability, and innovation. Boolean operators (AND, OR) refined search results, e.g., "Scrum AND (Artificial Intelligence OR AI)" [42]. The search was limited to English publications from 2000 to 2024 to ensure relevance and recency.

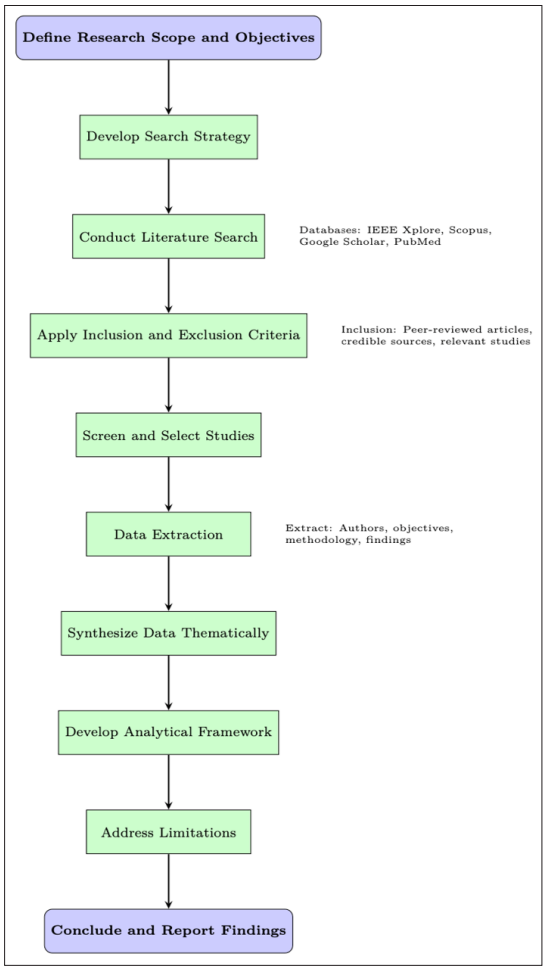

Figure 9: Flowchart of the systematic review methodology

To ensure quality and relevance, strict inclusion and exclusion criteria were applied. Inclusion criteria prioritized peer-reviewed articles, books by recognized experts, and credible online sources (e.g., Scrum.org, Agile Alliance) [43]. Studies were included if they addressed Scrum's necessity, advantages, limitations, or AI integration. Non-English publications, non-peer-reviewed sources, and studies unrelated to Scrum, Agile, or AI were excluded [44].

Data extraction focused on author(s), publication year, research objectives, methodology, and key findings. Thematic synthesis identified key themes, such as Scrum's adaptability, advantages, limitations, and AI integration potential, analyzed using a structured framework [39,45-48].

Limitations include language bias (excluding non-English studies), time frame constraints (2000–2024), source bias (reliance on peer-reviewed and credible sources), and a focus on software development, potentially limiting generalizability. Despite these, the methodology ensures a comprehensive and balanced review of Scrum's role in modern software development and its AI-era evolution.

Discussion

Why Scrum is Still Necessary: Scrum's iterative approach addresses the limitations of traditional methodologies like the waterfall model, which struggle with evolving requirements, leading to delays and misaligned deliverables [3,4]. By breaking projects into manageable sprints, Scrum enables regular delivery of functional increments, stakeholder feedback, and real-time adaptation [6,7]. Its adaptability is critical in fast-paced industries like tech, where companies like Spotify and Google use Scrum to maintain competitiveness [8,49,50]. Scrum's emphasis on collaboration and transparency also supports modern, cross-functional teams, enhancing productivity and accountability [7,8,51].

Advantages of Scrum: Scrum empowers teams through self-organization, fostering creativity and innovation [8,14,15]. Its transparency, facilitated by artifacts like the product backlog, sprint backlog, and increment, ensures visibility and alignment with project goals [9,20,52]. Continuous improvement through retrospectives drives process refinement, leading to higher-quality deliverables and efficiency [21,22,53].

Limitations of Scrum: Scrum faces challenges such as scope creep, particularly in large-scale projects, which can lead to delays and budget overruns [23,24]. Dependency on team collaboration can hinder progress in large or distributed teams [8,26,27]. Resistance to change in organizations accustomed to traditional methodologies can also impede Scrum's adoption [28,29]. Additionally, daily meetings may feel redundant or overwhelming for some team members [30,31].

Scrum in the AI Era: AI integration enhances Scrum workflows through automation, data-driven insights, and improved collaboration. AI-powered analytics predict bottlenecks and optimize processes , while tools like Jira and Trello automate backlog prioritization and task management, reducing administrative burdens [3,35,32,33,55,60]. AI-powered collaboration platforms, such as Slack and Microsoft Teams, facilitate real-time translation, sentiment analysis, and meeting summaries, bridging cultural and linguistic gaps in distributed teams [56,57,61].

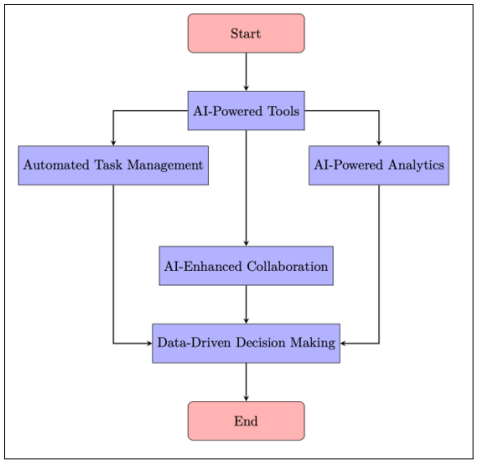

Figure 10: AI Integration in Scrum workflows

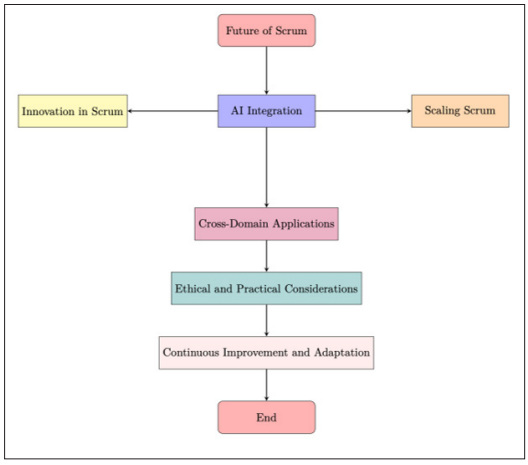

Figure 11: Future Roadmap for Scrum In the Ai Era

However, AI integration poses challenges, including ethical concerns like algorithmic bias and the need for human oversight to align decisions with organizational values [41,63]. Additionally, adopting AI tools requires teams to develop new skills and adapt processes, representing a cultural shift for organizations accustomed to traditional methodologies [58,59]. These advancements and challenges highlight Scrum's evolving role in the AI-driven software development landscape.

Synthesis of Findings: Scrum remains highly relevant in modern software development due to its adaptability, transparency, and focus on continuous improvement. While it offers significant advantages over traditional methodologies, challenges like scope creep and reliance on team collaboration must be addressed to maintain its effectiveness. The integration of AI into Scrum workflows presents opportunities to enhance efficiency, decision-making, and innovation. However, challenges such as ethical concerns, algorithmic bias, and the need for cultural adaptation require careful management to fully realize Scrum's potential in the AI era.

Future Directions

The future of Scrum hinges on its adaptability to emerging technologies, evolving market demands, and increasing project complexity. Key areas for evolution include:

AI Integration: AI can enhance Scrum workflows by automating tasks (e.g., backlog prioritization, task assignment), providing data-driven insights (e.g., predicting bottlenecks), and improving collaboration in distributed teams through real-time translation and sentiment analysis [3,6,8,37].

Scaling Scrum: Frameworks like SAFe, LeSS, and Nexus, along with advanced collaboration tools, can help scale Scrum for large, distributed teams, ensuring alignment and transparency [8,15,52].

Innovation in Practices: AI-enhanced retrospectives, gamification, and hybrid methodologies (e.g., Scrum-Kanban) can address emerging challenges and improve team engagement [2,68,69].

Cross-Domain Applications: Scrum's principles are increasingly applied in healthcare, education, and manufacturing, offering solutions for complex, iterative processes in these industries [70,71,25]. Ethical and Practical Considerations: Addressing AI bias, ensuring human oversight, and investing in skill development is critical for responsible AI integration [73-75].

Continuous Improvement: Emphasizing retrospectives, feedback loops, and community knowledge-sharing will ensure Scrum's ongoing relevance and effectiveness [76-78].

By focusing on these areas, Scrum can evolve to meet the demands of the AI era, maintaining its role as a vital framework for managing complex projects in a rapidly changing technological landscape.

Conclusions

Scrum has proven to be a resilient and adaptable framework in the ever-evolving landscape of software development. Its emphasis on iterative progress, collaboration, and continuous improvement has made it a cornerstone of Agile methodologies, enabling teams to deliver value incrementally and respond effectively to changing requirements. However, as the software industry enters the era of Artificial Intelligence (AI), Scrum faces new challenges and opportunities. The integration of AI into Scrum workflows offers transformative potential, from automating routine tasks to enhancing decision-making and collaboration. Yet, this integration also raises ethical and practical concerns, such as bias in AI algorithms and the need for human oversight.

To remain relevant, Scrum must evolve to address its limitations, such as scope creep, scalability issues, and resistance to change in traditional organizations. Scaling Scrum for large, distributed teams and fostering innovation through AI-powered tools and hybrid methodologies will be critical. Additionally, the application of Scrum beyond software development in industries like healthcare, education, and manufacturing highlights its versatility and potential for cross-domain adoption [80,81].

Ultimately, the future of Scrum lies in its ability to adapt to emerging technologies, embrace ethical considerations, and foster a culture of continuous improvement. By integrating AI, scaling effectively, and addressing its limitations, Scrum can continue to deliver value in the AI era and beyond.

References

1. Daraojimba EC, Nwasike CN, Adegbite AO, Ezeigweneme CA, Gidiagba JO (2024) Comprehensive review of agile methodologies in project management. Comput Sci & IT Res J 5: 190-218.
2. Ekechi C, Okeke C, Adama HE (2024) Enhancing agile product development with scrum methodologies: A detailed exploration of implementation practices and benefits. Eng Sci Technol J 5: 1542-1570.
3. Schwaber K, Sutherland J (2020) The Scrum Guide. Scrum.org. https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf.
4. Wikipedia (2024) Scrum (software development). Wikipedia.https://en.wikipedia.org/wiki/Scrum_(software_development).
5. Flora H (2014) A systematic study on agile software development methodologies and practices. Int J Comput Sci Inf Technol 5: 3626-3637.
6. Zikopi E (2019) A case study research on Scrum framework. KTH Royal Institute of Technology.
7. Bultmann P (2023) The history of Scrum. Agile 42. https://www.agile42.com/en/blog/scrum-history.
8. Scrum.org (2024) What is Scrum? Scrum.org. https://www.scrum.org/learning-series/what-is-scrum/.
9. Atlassian (2024) What is Scrum and how to get started. Atlassian. https://www.atlassian.com/agile/scrum.
10. Vicci H (2024) The impact of artificial intelligence on project managers and scrum masters: A review and evaluation study.
11. Cohn M (2010) Succeeding with Agile: Software Development Using Scrum. Addison-Wesley.
12. Pichler R (2010) Agile Product Management with Scrum. Addison-Wesley.
13. Aizaz F, Pervez Z, Mahmood S, Ali R, Khan SU, et al. (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.
14. ScrumAlliance (2024) The Scrum team roles and accountabilities. Scrum Alliance. https://resources.scrumalliance.org/Article/scrum-team
15. Blake S (2021) Scrum workflow: Roles, stages, and automation options. Easy Agile. https://www.easyagile.com/blog/scrum-workflow.
16. Vargas N, Lloria M, Roig-Dobón S (2016) Main drivers of human capital, learning, and performance. J Technol Transfer 41: 1-18.
17. Sutherland J (2021) The Scrum Papers: Nuts, Bolts, and Origins of an Agile Frameworkhttps://www.scruminc.com/scrumpapers.pdf.
18. Senarath US (2021) Waterfall methodology, prototyping and agile development.
19. Doshi H (2016) Three pillars of empiricism in Scrum. Scrum.org. https://www.scrum.org/resources/blog/three-pillars-empiricism-scrum.
20. Simplilearn (2024) Scrum project management: Advantages and disadvantages. Simplilearn. https://www.simplilearn.com/tutorials/agile-scrum-tutorial/what-is-scrum.
21. Schwaber K, Beedle M (2002) Agile Software Development with Scrum. Prentice Hall.
22. Mountain Goat Software (2024) Scrum team role and responsibility. Mountain Goat Software. https://www.mountaingoatsoftware.com/agile/scrum/roles/team.
23. Landau P (2022) The 7 Scrum artifacts: Definitions and examples. ProjectManager. https://www.projectmanager.com/blog/scrum-artifacts.
24. Visual Paradigm (2023) Understanding the dynamics: Product backlog vs sprint backlog. Visual Paradigm.
25. Zayat W, Senvar O (2020) Framework study for agile software development via Scrum and Kanban. Int J Innov Technol Manag 17: 2030002.
26. Rebel Scrum (2024) Agile Scrum Training Institute. Rebel

Scrum.
27. Demi S, Mikkonen T, Oivo M, Garbajosa J (2020) Mapping human values and Scrum roles: A study on students' preferences. In Proceedings of the IEEE/ACM International Conference on Software Engineering Workshops (ICSEW) 442-449.
28. Epic Flow (2021) The Scrum framework: Advantages, disadvantages, and ways to manage potential problems. Epic Flow.
29. Indeed (2022) A guide to Scrum team. Indeed.
30. Lobellova V (2020) The history of Scrum: How, when, and why. ScrumDesk.
31. Drumond C (2024) Members of the Scrum team. Atlassian.
32. Scrum.org (2024) What is a product owner? Scrum.org.
33. Scrum.org (2024) What is a developer on a Scrum team? Scrum.org.
34. Simpliaxis (2024) What are Scrum developer roles and responsibilities. Simpliaxis.
35. QRP (2020) Who is the Scrum developer: Role and responsibilities. QRP International.
36. Mdallal R, Alshurideh M, Al Kurdi B, Alzoubi H, Hamadneh S, et al. (2023) AI-powered conceptual model for Scrum framework. In Proceedings of the IEEE International Conference on Computer and Information Technology (ICCIT): 597-604.
37. Junco V (2024) The potential of AI to revolutionize organizational communication and teamwork. Rev Investig Multidisciplinaria Iberoamericana 3: 1-15.
38. Khasawneh N, Khasawneh S, Khasawneh M (2023) The potential of AI in facilitating cross-cultural communication through translation 107-130.
39. Karim EZ, Bagheri Khoulenjani A, Safaei M (2024) Integrating AI for agile project management: Innovations, challenges, and benefits.
40. Ramachandran KK, Rajasekaran V, Ramesh M, Balasubramanian S, Karthikeyan P, et al. (2024) Developing AI-powered training programs for employee upskilling and reskilling. J Inform Educ Res 4: 1187-1199.
41. Johnson M, Hoover R (2024) The role of human oversight in responsible AI systems.
42. MacFarlane A, Russell-Rose T, Shokraneh F (2022) Search strategy formulation for systematic reviews: Issues, challenges and opportunities. Intell Syst Appl 15: 200091.
43. Meline T (2006) Selecting studies for systematic review: Inclusion and exclusion criteria. Contemp Issues Commun Sci Disord 33: 21-27.
44. Nussbaumer-Streit B, Klerings I, Dobrescu AI, Persad E, Stevens A, et al. (2020) Excluding non-English publications from evidence syntheses did not change conclusions: A meta-epidemiological study. J Clin Epidemiol 118: 42-54
45. Amajuoyi P, Benjamin L, Adeusi K (2024) Agile methodologies: Adapting product management to rapidly changing market conditions. GSC Adv Res Rev 19: 249-267.
46. Zaheer M (2025) Comparative analysis of Scrumlity and alternative Agile frameworks for quality assurance.
47. Fitriani W, Rahayu P, Sensuse D (2016) Challenges in agile software development: A systematic literature review. Proc IEEE Int Conf Adv Comput Sci Inf Syst: 155-164.
48. Jin Z (2024) Integrating AI into agile workflows: Opportunities and challenges. Appl Comput Eng 100: 49-54.
49. Ahmad I, Ahmad S, Kazmi UI, Ali O, Khan MA, et al. (2024) Innovation management in tech start-ups: A review of strategies for growth and sustainability. Int J Sci Res Arch 11: 807-816.
50. Chukwurah E, Aderemi S (2024) Elevating team performance with Scrum: Insights from successful U.S. technology companies. Eng Sci Technol J 5: 1357-1371.
51. Celestin M, Vanitha N (2017) Remote project management: How to lead global teams from your living room. 240-246.
52. Carneiro LB, Silva ACCLM, Alencar LH (2018) Scrum agile project management methodology application for workflow management: A case study. In: Proceedings of the 2018 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM) 938-942
53. Sassa AC, Almeida IA, Pereira TNF, Oliveira MS (2023) Scrum: A systematic literature review. Int J Adv Comput Sci Appl 14: 173-181
54. Gemino A, Reich B, Serrador P (2020) Agile, traditional, and hybrid approaches to project success: Is hybrid a poor second choice? Proj Manag J: 52.
55. Nabeel M (2024) AI-enhanced project management systems for optimizing resource allocation and risk mitigation: Leveraging big data analysis to predict project outcomes and improve decision-making processes in complex projects. Asian J Multidiscip Res Rev 5: 53-91.
56. Agile For All (2017) Scrum Foundations: Increment and Done. YouTube. https://www.youtube.com/watch?v=0MTWysPmaoU.
57. GlobalLogic (2021) The Scrum Guide is Dead — Long Live the Scrum Guide! https://www.globallogic.com/insights/blogs/the-scrum-guide-is-dead-long-live-the-scrum-guide.
58. Let's Scrum It (2020) Sprint Goal in Scrum - Why Do We Need It? https://letsscrumit.com/blog/sprint-goal-why-do-we-need-it.
59. Miro (2024) What is a Scrum Workflow? https://miro.com/agile/what-is-a-scrum-workflow/.
60. Ogunbukola M (2024) The impact of artificial intelligence on project management: Enhancing efficiency, risk mitigation, and decision-making in complex projects.
61. Balla B (2024) The role of AI in streamlining Microsoft Teams and Slack integration. AI Journ. https://aijourn.com/the-role-of-ai-in-streamlining-microsoft-teams-and-slack-integration/.
62. Mohammad A, Chirchir B (2024) Challenges of integrating artificial intelligence in software project planning: A systematic literature review. Digital 4: 555-571.
63. Charkra S, Dey S, John A, Kanti S (2024) The human factor in AI decision-making: Mitigating bias and error.
64. Omonije A (2024) Agile methodology: A comprehensive impact on modern business operations. Int J Sci Res 13: 132-138.
65. Paasivaara M, Lassenius C, Heikkilä V (2012) Inter-team coordination in large-scale globally distributed Scrum: Do Scrum-of-Scrums work? Proc Int Symp Empirical Softw Eng Meas: 235-238.
66. Almeida F, Espinheira E (2021) Large-scale agile frameworks: A comparative review. J Appl Sci Manag Eng Technol 2: 16–29. DOI: 10.31284/j.jasmet.2021.v2i1.1832.
67. Joskowski A, Przybylek A, Marcinkowski B (2023) Scaling Scrum with a customized Nexus framework: A report from a joint industry-academia research project. Softw Pract Exper 53: 1525–1542. DOI: 10.1002/spe.3201.
68. Gelbard R, Ramon-Gonen R, Carmeli A, Bittmann RM, Talyansky R (2018) Sentiment analysis in organizational work: Towards an ontology of people analytics. Expert Syst 35: e12289.
69. Castro Lopes F, Fernandes S (2024) The use of gamification for learning SCRUM: Findings from a case study with information systems students. Trends High Educ 3: 235–246.

DOI: 10.3390/higheredu3020014.

70. Torrente G, De Souza TQ, Tonaki L, Cardoso AP, Manickchand Junior L, et al. (2021) Scrum framework and health solutions: Management and results. Stud Health Technol Inform 284: 290-294.
71. Kuz A (2021) Scrum: A new framework applied to education. Eduweb 15: 10-17. DOI: 10.46502/issn.1856-7576/2021.15.03.1.
72. Persson M, Daniels M, Cajander Å (2012) On the use of Scrum in project-driven higher education. In Proc Frontiers in Education Conference (FIE): 1-6.
73. Akter S, Dwivedi YK, Sajib S, Biswas K, Bandara RJ, et al. (2021) Addressing algorithmic bias in AI-driven customer management. J Glob Inf Manag 29: 1-27.
74. Rodgers W, Murray DJ, Stefanidis A (2023) An artificial intelligence algorithmic approach to ethical decision-making in human resource management processes. Hum Resour Manag Rev 33: 100925.
75. Naiburg E (2024) AI as a Scrum team member. Scrum.org. Available at: https://www.scrum.org/resources/blog/ai-scrum-team-member.
76. Sverrisdottir HS, Ingason HT, Jonasson HI (2014) The role of the product owner in Scrum: Comparison between theory and practices. In Proceedings of the 28th IPMA World Congress on Projects, Programmes and Portfolio Management: 257-267.
77. Rad K, Turley F (2018) Agile Scrum Foundation Courseware. Van Haren Publishing.
78. Scrum.org (2024) The Scrum Team. Available at: https://www.scrum.org/resources/scrum-team/
79. Scrum Alliance (2024) State of Scrum Report. Available at: https://www.scrumalliance.org/
80. VersionOne (2024) State of Agile Report. Available at: https://www.stateofagile.com/
81. Highsmith J (2002) Agile Software Development Ecosystems. Addison-Wesley Professional.