THE IMPACT OF AGENTIC AI ON SCRUM TEAM ROLES AND SOFTWARE DELIVERY PIPELINES:

Implications for SDG 8 and SDG 9

Solomon Simukoko (1052207)
solomonsimukoko@

Table of Contents

Introduction ***2***

Methodology ***3***

Research Design 3

Data Collection and Search Strategy 3

Inclusion and Exclusion Criteria 3

Data Analysis and Thematic Synthesis 4

The Economics of Agentic AI and the Rise of AI Tokenisation ***4***

Workforce Transformation and the Wider Implications of AI Tokenisation ***8***

Implications of Agentic AI for Sustainable Development Goals 8 and 9 ..... ***11***

Agentic AI and SDG 8: Decent Work and Economic Growth *11*

Agentic AI and SDG 9: Industry, Innovation and Infrastructure ***13***

Interdependence of the Sustainable Development Goals ***14***

Analysis and Discussion ***15***

Transformation of Scrum Team Roles *15*

Always-on Agents and the Continuous Software Development Lifecycle *16*

AI Infrastructure and Sustainable Software Engineering *19*

Conclusion ***21***

References ***24***

Introduction

The rapid advancement of agentic artificial intelligence (AI) represents one of the most transformative developments in contemporary software engineering. Unlike earlier generations of automation, which relied upon predefined rules and deterministic workflows, agentic AI systems possess the capability to reason, plan, utilise external tools and autonomously execute complex multi-stage tasks with limited human intervention (Anthropic, 2025). Consequently, AI is no longer functioning solely as an assistive technology but is increasingly becoming an active participant within software engineering teams (Scrum Teams), contributing directly to software design, implementation, testing, deployment and operational support. The emergence of these autonomous capabilities has the potential to fundamentally reshape both the composition of Agile Scrum teams and the structure of modern software delivery pipelines. (OpenAI, 2025).

The implications of these developments extend well beyond software engineering and align closely with the United Nations Sustainable Development Goals (SDGs), which provide an internationally recognised framework for evaluating the broader social, economic and environmental consequences of technological innovation (United Nations, 2015). Within the context of this study, two Sustainable Development Goals are particularly relevant. SDG 8 (Decent Work and Economic Growth) focuses upon productive employment, inclusive economic growth and workforce development. SDG 9 (Industry, Innovation and Infrastructure) promotes technological innovation, resilient infrastructure and sustainable industrialisation.

This paper critically reviews the current literature concerning the impact of agentic AI on Scrum team roles and software delivery pipelines. It begins by examining the changing economics of software development through the emergence of AI agents and token-based computational intelligence. It then explores how these developments are reshaping the composition and responsibilities of Agile Scrum teams before considering the wider workforce and environmental implications associated with increasing AI adoption. Finally, the chapter synthesises the literature providing a conceptual foundation for the empirical investigation presented in the subsequent chapters.

Methodology

Research Design

To ensure this study remains grounded in practical industry realities, the methodological design is situated within the professional disciplines of software engineering and business analysis. The research approaches the integration of agentic AI through a structured problem-solving lens, specifically analysing how automated cognitive capabilities disrupt traditional Agile delivery pipelines and economic models. In strict accordance with the study parameters, this research employs a secondary data analysis design, utilising a critical literature review and thematic synthesis to evaluate the operational and environmental impacts of agentic AI.

Data Collection and Search Strategy

The data collection process was designed to capture both the qualitative implications of AI integration and the quantitative economic and environmental realities of its deployment.

Inclusion and Exclusion Criteria

To maintain relevance and methodological rigor, strict inclusion and exclusion criteria were applied to the secondary data selection process:

- **Temporal Relevance:** Given the rapid advancement of large language models and generative AI, priority was given to literature and data sets published between 2021 and 2026. Older sources were excluded unless they provided foundational definitions of Agile methodologies or core design principles.
- **Thematic Relevance:** Sources were required to directly address either the economic restructuring of software delivery, the evolution of Agile/Scrum team dynamics, or the digital infrastructure requirements of AI deployment.
- **Data Type:** Only secondary qualitative and quantitative data were included. Case studies or papers relying strictly on the primary research of the authors were filtered to extract only their verified statistical findings or literature reviews.

Data Analysis and Thematic Synthesis

The collected secondary data were analysed using thematic synthesis. The extracted data was coded and categorised into three primary analytical themes: the economics of AI tokenisation, the transformation of multidisciplinary Scrum roles, and the environmental footprint of computational infrastructure.

Once the data was synthesised into these operational themes, a framework analysis was applied using the United Nations Sustainable Development Goals. The findings were mapped specifically against SDG 8 (Decent Work and Economic Growth) and SDG 9 (Industry, Innovation and Infrastructure) to evaluate both the positive advancements and negative consequences of agentic AI within contemporary software engineering practice.

The Economics of Agentic AI and the Rise of AI Tokenisation

The emergence of agentic artificial intelligence has fundamentally altered the economics of software development by transforming intelligence into a scalable computational resource. Historically, organisations increased productive capacity by recruiting additional employees, investing in staff development and expanding organisational structures. Productivity was therefore closely associated with workforce growth and labour costs. The introduction of generative AI has challenged this long-established relationship by enabling organisations to acquire computational intelligence on demand, allowing cognitive work to be performed through AI inference rather than exclusively through human expertise (Goldman Sachs, 2026). This shift represents one of the most significant economic transformations affecting knowledge-intensive industries since the widespread adoption of cloud computing.

Unlike traditional enterprise software, which is typically purchased through perpetual licences or subscription-based pricing models, modern large language models operate on a usage-based pricing structure. Organisations pay according to the number of tokens processed during inference, where a token represents the smallest unit of text interpreted or generated by an AI model (OpenAI, 2025). Every prompt submitted to an AI system and every response returned consumes computational

resources that are measured through token usage. Consequently, intelligence itself has become a measurable operational commodity, allowing organisations to purchase varying levels of computational capability according to business demand. This transition has given rise to what industry analysts describe as the tokenisation of intelligence, where the value of AI is increasingly determined by computational consumption rather than software ownership or employee capacity (Goldman Sachs, 2026).

The financial implications of this transition are illustrated by recent industry analyses comparing the operational costs of AI agents with comparable human labour. Goldman Sachs (2026) estimates that an AI coding agent costs approximately US$13.39 per day, compared with approximately US$300 per day for a software engineer performing equivalent coding activities. Similarly, AI-powered data-entry systems are estimated to operate at approximately US$59.68 per day, compared with approximately US$80 per day for equivalent human labour. However, this economic advantage is not universal. Voice-based AI call-centre agents currently remain slightly more expensive than comparable human workers because continuous speech recognition and synthesis require significantly greater computational resources during inference, resulting in an estimated operational cost of US$92.90 per day compared with US$90.00 per day for human labour (Goldman Sachs, 2026).

Table 1

Comparison of Estimated Daily Operating Costs Between Human Labour and AI Agents (Adapted from Goldman Sachs, 2026)

Occupation	Human Labour (US$/day)	AI Agent (US$/day)	Lower-Cost Option
Software Engineer	300.00	13.39	AI Agent
Call-centre Agent	90.00	92.90	Human Worker
Data-entry Worker	80.00	59.68	AI Agent

What this article interestingly illustrates is how jobs that we considered difficult to automate or replace just a few years ago are now being replaced more quickly by AI. For example, software engineering roles can increasingly be performed by AI models. However, call centre jobs, where human reasoning and interaction are important, can

be more difficult to replace, particularly when the cost of running the AI system is higher than simply employing a human. This suggests that jobs involving advanced analytical tasks may be easier for AI to replace, while jobs involving basic reasoning, interpersonal interaction and simpler decision-making may actually be more difficult or expensive to automate.

Furthermore, if we zoom into the Scrum team itself, the impact of AI becomes particularly interesting when we examine the different roles within the software development lifecycle. Consider software testing as an example. Traditionally, a software engineer in test (SDET), who develops automated testing scripts and frameworks, would generally be regarded as a more technically advanced and potentially more lucrative role than that of a manual tester. However, generative AI and AI agents are beginning to challenge this distinction because many of the activities associated with automated testing can increasingly be generated or executed by AI. Generative AI can produce test cases, generate automation scripts and assist with identifying defects, meaning that some of the work previously requiring a specialist automation tester can potentially be performed by an AI-assisted workflow (Sauvola et al., 2024; Pasuksmit et al., 2025).

This does not necessarily mean that the manual testing role disappears. In fact, it may create an interesting reversal in the relative value of different testing activities. Automated testing is highly structured: the tester can provide the AI with requirements, expected outcomes, test data and technical specifications, after which the AI can generate scripts and execute tests against the system. Research into AI-assisted software development demonstrates how quickly AI can perform structured technical tasks, while also highlighting the continued requirement for human oversight and evaluation of AI-generated outputs (Tosi, D. (2024), 2024; Pasuksmit et al., 2025). In this sense, the more a testing activity can be reduced to a clearly defined technical process, the greater the potential for AI to automate it.

Manual and exploratory testing is different. A human tester does not simply follow a predefined sequence of instructions; they interact with the system, interpret what they see, make judgements about whether the behaviour is appropriate and sometimes investigate unexpected behaviour that was not anticipated in the original requirements. This becomes particularly important when testing interfaces containing

complex geographical or spatial features. An AI model may be able to understand the underlying code, identify expected outputs and generate automated tests, but this is not necessarily equivalent to understanding how a real person experiences and interacts with the interface. Recent research on AI-assisted usability testing similarly suggests that LLM-based agents can complement human evaluators, rather than simply making human evaluation redundant (Geuenich et al., 2026).

There is therefore an important economic dimension to this argument. The question is not simply whether AI can perform a particular task, but whether it is economically and operationally worthwhile to replace the human performing it. If a relatively inexpensive manual tester can provide the necessary human judgement, while an AI agent requires substantial computational resources, configuration, monitoring and human oversight, there may be little immediate economic incentive to automate that particular activity. This creates a counterintuitive situation in which some lower-paid roles may initially be more resistant to automation than higher-paid technical roles. The determining factor is not necessarily the complexity or salary associated with the role, but the extent to which its underlying tasks can be codified and reliably reproduced by AI.

The same distinction can be applied to the role of the Business Analyst (BA). AI can already assist with many of the more structured outputs produced by a BA. For example, a set of requirements can potentially be transformed into user stories, acceptance criteria, process models, test scenarios or other forms of documentation. This means that some of the traditional production work associated with business analysis could become significantly faster or increasingly automated. Generative AI's ability to produce and transform structured information is one of the reasons why knowledge-intensive professional work is increasingly being reorganised around human-AI collaboration rather than simply manual production (Sauvola et al., 2024).

However, producing a user story is not the same as discovering the requirement in the first place. A significant part of the BA's value lies in interacting with stakeholders, asking questions, challenging assumptions, identifying contradictions and uncovering information that stakeholders may not explicitly articulate. This is particularly

important because organisational knowledge is not always formally documented. Some of the most valuable knowledge is tacit: it exists in people's experience, habits, assumptions and understanding of how a particular organisation actually operates. Recent research continues to examine the difficulty of integrating tacit knowledge into generative-AI systems, demonstrating that the problem is not simply one of transferring more information into an AI model (Xu et al., 2026).

For example, a stakeholder may tell a BA that a process works in a particular way, while an experienced BA may recognise from the conversation that this is only the formal process and that employees actually follow a different process in practice. The BA can ask follow-up questions, observe behaviour, recognise ambiguity and use contextual knowledge to uncover the underlying requirement. AI can assist with documenting what has been discovered, but the discovery itself depends heavily on interaction and context. This distinction is important because tacit knowledge is often difficult to express explicitly until someone with the appropriate experience asks the right question.

Consequently, AI may change the BA role in a similar way to how it is changing software development and testing. The production of artefacts may become increasingly automated, while the activities involving judgement, stakeholder engagement, contextual understanding and discovery may become relatively more important. The BA could therefore spend less time writing user stories and more time determining which user stories should exist in the first place.

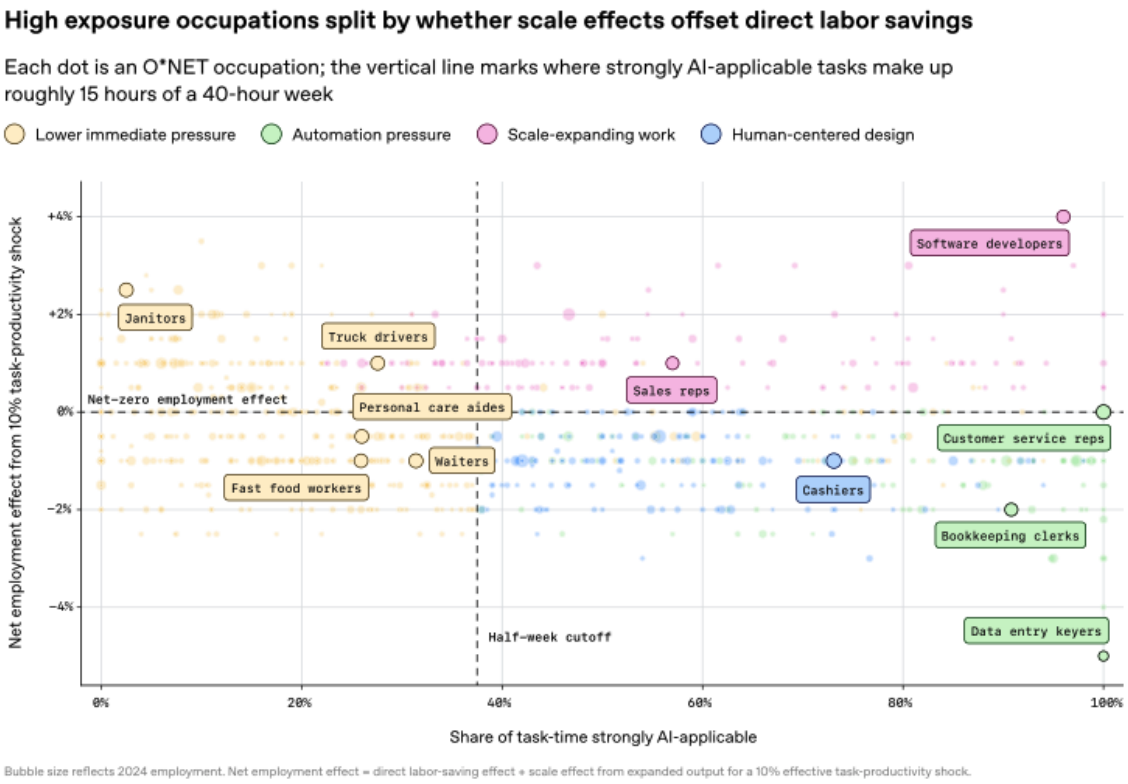
This suggests a broader principle about AI and employment: AI is likely to have the greatest immediate impact on tasks that are structured, repeatable and easily codified, while tasks requiring contextual judgement, interpersonal interaction and tacit knowledge may remain more resistant to automation.

Workforce Transformation and the Wider Implications of AI Tokenisation

f

The transformation of Scrum team roles reflects a broader shift occurring across knowledge-intensive industries. As organisations increasingly integrate agentic AI into everyday business operations, the relationship between labour, productivity and organisational value creation is being fundamentally redefined. Rather than replacing entire professions, current evidence suggests that AI is reshaping how work is organised by automating routine cognitive activities while increasing demand for skills that complement computational intelligence (Brynjolfsson et al., 2023). Consequently, the future workforce is likely to be characterised by collaboration between human expertise and AI systems rather than competition between them.

A distinguishing feature of generative AI is that it targets cognitive rather than physical labour.



Previous technological revolutions, such as mechanisation and industrial automation, primarily replaced repetitive manual tasks. Agentic AI, by contrast, is capable of performing activities traditionally associated with professional expertise, including analysing documents, generating reports, writing software, summarising information and supporting decision-making. This expansion into knowledge work has generated considerable debate regarding the future of employment and the changing nature of professional roles.

Current literature generally rejects the notion that AI will lead to the immediate disappearance of entire occupations. Instead, researchers argue that occupations consist of numerous individual tasks, some of which are more susceptible to automation than others (Brynjolfsson et al., 2023). Activities characterised by structured information processing, repetitive documentation and predictable workflows are increasingly being delegated to AI systems. In contrast, tasks requiring negotiation, creativity, ethical judgement, leadership, organisational awareness and emotional intelligence remain significantly more dependent upon human expertise. This distinction has important implications because it suggests that AI is changing the composition of professional work rather than eliminating professions altogether.

Within software engineering this task-based transformation is particularly evident. Developers increasingly spend less time producing routine code and more time reviewing AI-generated implementations. Business Analysts devote greater attention to stakeholder engagement and organisational strategy rather than documentation. Test engineers increasingly supervise automated validation processes instead of manually constructing extensive test suites. These changes indicate that professional expertise is moving upwards within the value chain, with humans concentrating on activities requiring judgement while AI performs routine technical execution.

The emergence of agentic AI also introduces entirely new categories of work. As organisations become increasingly dependent upon AI systems, demand is growing for professionals specialising in AI governance, model evaluation, prompt engineering, AI assurance, regulatory compliance and ethical oversight. These emerging disciplines illustrate an important characteristic of technological change: while some tasks disappear, new forms of employment simultaneously emerge. Historically, technological innovation has consistently created occupations that previously did not exist, and early evidence suggests that the AI economy is following a similar pattern (World Economic Forum, 2025).

Implications of Agentic AI for Sustainable Development Goals 8 and 9

The preceding literature demonstrates that agentic AI is more than a technological innovation; it represents a socio-technical transformation that influences organisational productivity, workforce structures, digital infrastructure and environmental sustainability simultaneously. Examining these developments through the United Nations Sustainable Development Goals (SDGs) provides a useful framework for understanding both the opportunities and challenges associated with AI adoption. Rather than considering economic growth, technological innovation and environmental sustainability as separate issues, the literature increasingly suggests that they are deeply interconnected. Consequently, the successful integration of agentic AI depends upon balancing the objectives of SDG 8 (Decent Work and Economic Growth) and SDG 9 (Industry, Innovation and Infrastructure)

Agentic AI and SDG 8: Decent Work and Economic Growth

SDG 8 promotes sustained, inclusive economic growth, productive employment and decent work for all (United Nations, 2015). The literature reviewed throughout this chapter indicates that agentic AI has significant potential to contribute towards these objectives by increasing productivity, reducing operational costs and enabling organisations to deliver greater economic value using existing resources. The tokenisation of intelligence allows organisations to purchase computational capability on demand, fundamentally changing how productive capacity is acquired. Instead of relying solely upon workforce expansion, organisations can increasingly scale operations through AI inference, improving efficiency while reducing the cost of routine knowledge-intensive activities (Goldman Sachs, 2026).

These productivity improvements are particularly evident within software engineering. AI-assisted coding, automated testing, intelligent documentation and deployment support enable Scrum teams to complete routine implementation activities more rapidly, allowing professionals to devote greater attention to higher-value activities such as architecture, stakeholder engagement and organisational strategy. Rather than diminishing the importance of human expertise, the literature suggests that AI reallocates professional effort towards tasks requiring judgement, creativity and contextual understanding (Brynjolfsson et al., 2023). From this

perspective, agentic AI aligns closely with the objectives of SDG 8 by enhancing organisational productivity while creating opportunities for higher-value employment.

However, productivity alone does not guarantee decent work. A recurring theme throughout the literature concerns the distribution of economic benefits generated through AI adoption. Acemoglu and Johnson (2023) argue that technological progress has historically favoured organisations controlling capital and technological infrastructure rather than automatically improving employment outcomes for the wider workforce. Similar concerns emerge within contemporary discussions surrounding agentic AI. Organisations possessing advanced computational infrastructure, proprietary datasets and financial resources may realise disproportionate productivity gains, while smaller organisations and lower-income economies struggle to access comparable capabilities.

The literature therefore highlights the importance of workforce adaptation. As routine cognitive activities become increasingly automated, demand is expected to shift towards skills that complement rather than compete with AI. Digital literacy, AI governance, interdisciplinary collaboration, ethical reasoning and critical evaluation of AI-generated outputs are increasingly recognised as essential competencies within the emerging digital economy (Organisation for Economic Co-operation and Development [OECD], 2024). Consequently, achieving SDG 8 requires more than technological innovation; it requires sustained investment in education, lifelong learning and organisational reskilling initiatives capable of preparing workers for AI-enabled workplaces.

Another important consideration concerns the quality of work itself. AI has the potential to reduce repetitive administrative activities, enabling employees to focus on more meaningful and intellectually engaging responsibilities. Conversely, excessive dependence upon AI may reduce professional autonomy if human workers become primarily responsible for supervising algorithmic systems rather than exercising independent judgement. Maintaining appropriate human oversight therefore becomes essential to ensuring that AI enhances rather than diminishes the quality of professional work. Human-centred governance frameworks that promote transparency, accountability and ethical decision-making are therefore fundamental to aligning AI adoption with the principles of decent work.

Agentic AI and SDG 9: Industry, Innovation and Infrastructure

The literature similarly demonstrates a strong relationship between agentic AI and SDG 9, which promotes resilient infrastructure, sustainable industrialisation and innovation (United Nations, 2015). Throughout this chapter, AI has been shown to influence software engineering at multiple stages of the software development lifecycle. Rather than functioning as an isolated development tool, agentic AI increasingly participates throughout software delivery pipelines by generating code, interpreting requirements, supporting testing, producing documentation and assisting deployment activities. Consequently, AI is becoming embedded within the digital infrastructure that underpins contemporary software engineering.

The integration of AI throughout software delivery pipelines enables organisations to increase development speed while maintaining iterative delivery practices consistent with Agile methodologies. Rapid prototyping, automated documentation and AI-assisted implementation reduce development cycle times and enable organisations to respond more effectively to changing customer requirements. These capabilities strengthen organisational innovation by reducing barriers between conceptual design and software deployment, thereby supporting the industrial innovation objectives underpinning SDG 9.

The literature also indicates that AI contributes towards the democratisation of software development. AI-assisted development tools increasingly enable small organisations, start-ups and public sector institutions to access sophisticated software engineering capabilities that were previously available only to organisations possessing extensive technical expertise and financial resources (McKinsey & Company, 2023). By lowering technical barriers to innovation, agentic AI has the potential to broaden participation in digital transformation and accelerate technological development across multiple sectors of the economy.

Nevertheless, this transformation introduces new infrastructure challenges. Modern AI systems depend heavily upon cloud computing, hyperscale data centres and specialised semiconductor technologies operated by a relatively small number of global technology providers. This concentration of computational capability raises important concerns regarding resilience, technological sovereignty and organisational dependency. Software delivery pipelines increasingly depend upon external AI services, creating potential vulnerabilities associated with infrastructure

failures, cybersecurity incidents and commercial restrictions. Consequently, resilient digital infrastructure requires organisations to balance AI adoption with strategies that minimise dependency upon individual providers while ensuring continuity of service.

Cybersecurity further illustrates the need for balanced governance. Although AI can strengthen software security through automated vulnerability detection and secure coding recommendations, it may simultaneously introduce insecure implementations, hallucinated outputs or hidden defects when human oversight is insufficient (Pearce et al., 2022). Sustainable innovation therefore requires governance structures capable of ensuring that AI-generated software remains secure, reliable and aligned with organisational objectives. The literature consistently suggests that technological innovation alone is insufficient; responsible governance is equally necessary if digital infrastructure is to remain resilient over the long term.

The environmental implications also establish a direct relationship between agentic AI and SDG 13. The computational infrastructure supporting token-based AI services requires substantial electricity generation, advanced cooling systems and continuous investment in specialised hardware (International Energy Agency [IEA], 2025). As organisations increasingly rely upon AI inference to support software engineering and broader business operations, demand for hyperscale computational infrastructure is expected to continue expanding. Consequently, the environmental cost of computational intelligence becomes inseparable from discussions surrounding AI-driven productivity and innovation.

Interdependence of the Sustainable Development Goals

A key finding emerging from the literature is that SDGs 8 and 9 cannot be considered independently when evaluating the impact of agentic AI. Increased productivity (SDG 8) depends upon expanding digital infrastructure and technological innovation (SDG 9. Progress towards one Sustainable Development Goal may therefore generate unintended consequences for another unless supported by appropriate governance frameworks.

This interdependence reinforces the importance of adopting a systems perspective towards AI implementation. Organisations should evaluate AI not solely according to financial return on investment or development speed but also according to

workforce development, digital resilience and environmental sustainability. Similarly, policymakers must ensure that strategies promoting AI innovation are accompanied by investment in education, renewable energy and digital infrastructure capable of supporting long-term sustainable development. Collectively, the literature suggests that the future success of the agentic economy depends upon achieving an appropriate balance between technological progress, economic prosperity and responsible environmental stewardship.

The synthesis presented throughout this chapter provides the theoretical foundation for the empirical investigation undertaken within this dissertation. Although existing research has explored individual aspects of AI adoption, comparatively limited attention has been given to understanding how agentic AI simultaneously transforms Scrum team roles, software delivery pipelines and sustainable development objectives. Addressing this gap provides the focus for the research presented in the subsequent chapters.

Analysis and Discussion

Transformation of Scrum Team Roles

OpenAI's analysis of more than 800,000 work-related ChatGPT found that 43.5% of occupation-specific AI interactions involved tasks traditionally associated with another occupation, a phenomenon described as task crossover (OpenAI, 2026). Rather than indicating occupational replacement, these findings demonstrate that AI enables professionals to perform activities that previously required specialist expertise. Within Scrum teams, this changes the traditional workflow in which Business Analysts gather and document requirements before handing work to developers, developers pass completed software to testers, and testers subsequently transfer validated software to DevOps engineers for deployment. Instead, AI increasingly enables each member of the Scrum team to perform elements of these activities independently, reducing dependence on sequential hand-offs while strengthening cross-functional collaboration (OpenAI, 2026).

Figure 1 here – OpenAI Task Crossover ("Who borrows work, and whose work gets borrowed").

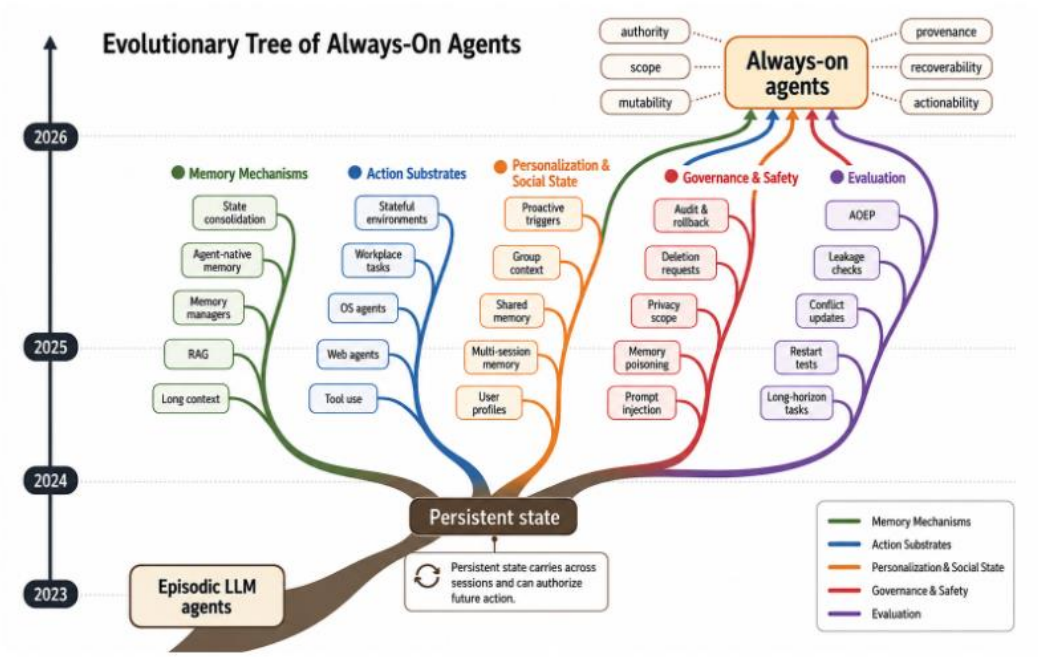
The task crossover evidence illustrates that software engineering roles are becoming increasingly integrated. Software Developers may now generate user stories, acceptance criteria and technical documentation that were traditionally produced by Business Analysts. Similarly, Business Analysts can employ AI to generate SQL queries, perform data analysis and develop automated testing scenarios, while Test Engineers increasingly use AI to generate code, recommend software fixes and create deployment scripts. Scrum Masters may also utilise AI to prepare sprint reports, identify delivery risks and analyse team performance before sprint reviews. These developments suggest that AI is reducing the functional separation between Scrum roles by allowing professionals to undertake a wider range of activities than has traditionally been possible. This redistribution of work has important implications for Agile software delivery. Traditional Scrum assumes that work progresses through clearly defined stages requiring collaboration between different specialists.

Always-on Agents and the Continuous Software Development Lifecycle

While the redistribution of work across Scrum teams represents one consequence of Agentic AI, the secondary data suggest that an equally significant transformation concerns the continuity of software engineering activities. Traditional Scrum and Agile methodologies are fundamentally designed around human working patterns, where software development progresses during normal working hours before pausing when team members finish work for the day. Although Continuous

Integration and Continuous Deployment (CI/CD) pipelines automate software builds, testing and deployments, many higher-level engineering activities such as requirements analysis, code generation, architectural design and defect investigation continue to depend upon human intervention. The emergence of always-on agents challenges this traditional operating model by enabling software engineering activities to continue independently of human availability (Ding et al., 2026).

Unlike conventional large language model (LLM) applications that process individual requests before resetting their context, always-on agents retain durable state across multiple interactions. This enables previous decisions, project knowledge, user preferences and software artefacts to influence future actions without requiring continual human prompting. Ding et al. (2026) argue that the defining characteristic of these systems is "not continuous execution but persistence" (p. 5). Rather than simply remembering previous conversations, persistent state enables AI agents to continue executing delegated activities over extended periods while maintaining awareness of project context. Consequently, AI agents are capable of resuming software engineering tasks where they previously stopped, allowing development activities to progress across multiple sessions without losing continuity.



Tree of Always-on Agents (Ding et al., 2026, p. 7).

The framework proposed by Ding et al. (2026) demonstrates that always-on agents extend beyond traditional memory systems by combining persistent memory, action

mechanisms, governance, personalisation and evaluation into a unified architecture capable of supporting long-term autonomous behaviour. When considered alongside the software development lifecycle, this model illustrates how AI agents may become active participants throughout software delivery rather than isolated coding assistants. An always-on Business Analyst could continue refining user stories as new stakeholder information becomes available, while an AI Developer continues generating source code based upon approved requirements. Similarly, AI Test Engineers may execute regression testing following repository updates, AI DevOps agents may monitor production environments for operational failures and AI documentation agents may continuously update technical documentation to reflect software changes. These activities no longer depend exclusively upon the working hours of human Scrum teams but may continue throughout the day and night as persistent agents maintain awareness of previous work.

However, the same characteristics that enable continuous software engineering also introduce new governance challenges. Persistent state enables AI systems to retain previous knowledge, but it also allows incorrect, outdated or malicious information to influence future decisions over extended periods. Ding et al. (2026) observe that "persistence converts transient, single-episode problems into long-lived, system-level ones" (p. 6), highlighting that errors are no longer confined to individual interactions but may continue affecting software delivery throughout multiple development cycles. For example, an incorrect architectural decision retained by an AI agent may continue influencing future code generation, testing activities and deployment recommendations long after the original mistake occurred. Similarly, outdated requirements, incorrect business rules or compromised persistent memory may continue shaping future software engineering decisions unless appropriate governance mechanisms exist to identify and correct them.

To address these challenges, Ding et al. (2026) propose six dimensions for governing persistent state: authority, scope, mutability, provenance, recoverability and actionability. Collectively, these dimensions ensure that retained information can be trusted before it influences future actions. Within software engineering these governance mechanisms become increasingly important because autonomous agents may simultaneously interact with source-code repositories, CI/CD pipelines, cloud infrastructure, testing environments and production systems. Consequently, the successful implementation of always-on agents depends not only upon improvements in AI capability but equally upon the development of governance frameworks capable of ensuring that persistent state remains accurate, auditable and

recoverable throughout the software development lifecycle. These findings suggest that the evolution towards continuous software engineering is not solely a technological challenge but also an organisational and governance challenge requiring new approaches to managing autonomous AI within modern Scrum teams.

AI Infrastructure and Sustainable Software Engineering

This relationship between persistent AI and computational infrastructure is becoming increasingly evident through the rapid expansion of AI data centres, often referred to as AI factories. These facilities provide the graphics processing units (GPUs), storage systems, networking infrastructure and specialised hardware necessary to support continuous AI inference at scale. Unlike traditional enterprise data centres, AI factories are designed to operate continuously, supporting thousands of concurrent AI agents that generate, evaluate and refine outputs simultaneously. Consequently, the computational requirements of Agentic AI are directly linked to the development of increasingly sophisticated digital infrastructure capable of supporting twenty-four-hour AI operations (International Energy Agency [IEA], 2025).

Figure 3 here – AI Factory/Renewable Energy Infrastructure image.

The International Energy Agency (2025) identifies AI as one of the fastest-growing contributors to electricity demand within digital infrastructure, with hyperscale AI data centres requiring substantial and continuous supplies of electrical power. The

evidence reviewed throughout this study suggests that this increase is not simply the result of larger AI models but also of the emergence of always-on agents capable of operating continuously across multiple software engineering activities. Unlike human Scrum teams, whose computational requirements reduce significantly outside normal working hours, persistent AI agents continue monitoring applications, analysing repositories, executing automated testing, responding to operational incidents and updating documentation regardless of time. Consequently, the transition towards continuous software engineering also represents a transition towards continuous computational demand.

The literature also demonstrates that the expansion of AI infrastructure has encouraged significant innovation in sustainable computing. To reduce environmental impacts, organisations are increasingly locating AI factories alongside renewable energy sources such as hydroelectric, wind, solar and nuclear power while adopting advanced liquid-cooling technologies capable of recovering waste heat generated by high-performance computing systems (IEA, 2025; Lund et al., 2025). Rather than allowing thermal energy to dissipate into the atmosphere, many modern facilities capture this heat and redistribute it through district heating systems that supply residential, commercial and industrial buildings. These developments indicate that AI factories are evolving beyond conventional data centres to become integrated components of broader sustainable energy systems.

The analysis undertaken throughout this chapter therefore demonstrates that Agentic AI is transforming software engineering at multiple levels simultaneously. The evidence indicates that AI is redistributing work across Scrum teams, enabling software engineering activities to continue beyond traditional working hours through always-on agents and increasing demand for resilient digital infrastructure capable of supporting continuous AI operations. Collectively, these findings suggest that the future of software engineering will be determined not only by improvements in AI capability but equally by the governance of autonomous systems and the sustainability of the infrastructure upon which they depend. This reinforces the central argument of this study that Agentic AI should be viewed not simply as a software development tool but as a technological transformation affecting professional practice, software delivery pipelines and the wider digital ecosystem.

Conclusion

The increasing integration of Agentic AI into software engineering has significant implications for Scrum teams and the software development lifecycle. While autonomous AI systems have the potential to improve productivity by generating code, analysing requirements, producing documentation and supporting testing activities, their deployment also introduces risks that require careful governance. For this reason, the UK Government's AI assurance guidance and the broader AI governance literature emphasise the importance of maintaining a human-in-the-loop (HITL) approach, particularly for decisions that have operational, legal or safety implications (Department for Science, Innovation and Technology [DSIT], 2025; National Institute of Standards and Technology [NIST], 2023). Human oversight remains essential because current large language models are probabilistic systems that predict likely outputs rather than verify objective truth.

One of the principal limitations of contemporary AI systems is that they generate responses through statistical pattern prediction rather than logical reasoning or factual verification. Large language models produce text by estimating the most probable sequence of tokens based on patterns learned during training rather than by consulting an authoritative knowledge base for every response (Bubeck et al., 2023; OpenAI, 2025). Consequently, an AI system may produce an output that appears plausible and syntactically correct while being factually inaccurate. This phenomenon, commonly referred to as AI hallucination, occurs because the model optimises for linguistic probability rather than factual certainty (Ji et al., 2023). In software engineering, this may result in AI generating code that compiles successfully but contains logical defects, security vulnerabilities or implementation errors that are not immediately obvious.

A second challenge arises from the optimisation objectives used during model training. Reinforcement Learning from Human Feedback (RLHF) encourages models to generate responses that users perceive as useful, relevant and helpful (Ouyang et al., 2022). While this significantly improves user experience, it may also increase the likelihood that an AI system attempts to provide an answer even when insufficient evidence exists. Instead of responding that it lacks sufficient information, the model may generate a plausible but incorrect response with a high degree of confidence. Recent research has shown that this tendency can contribute to overconfident or fabricated outputs, particularly in situations involving incomplete, ambiguous or conflicting information (Ji et al., 2023). Within a Scrum environment, this behaviour

may lead AI systems to generate incomplete business logic, invent software interfaces, fabricate API responses or create placeholder implementations that appear functional but fail during integration or production deployment.

A third limitation concerns the dependence of Agentic AI on external tools and information sources. Modern AI agents increasingly interact with APIs, databases, source-code repositories and retrieval systems to augment their knowledge beyond their training data (Lewis et al., 2020). However, when external services become unavailable or retrieval mechanisms fail, AI systems may continue generating responses using internal statistical knowledge rather than explicitly acknowledging uncertainty. As a result, software engineers may unknowingly receive outdated or fabricated information when an external dependency is unavailable. This becomes increasingly significant within always-on agents, where persistent state may preserve incorrect assumptions and allow them to influence future decisions across multiple development cycles (Ding et al., 2026).

These limitations have important implications for Scrum teams. Although Agentic AI can substantially reduce the effort required to produce code, user stories, test cases and technical documentation, it simultaneously shifts human effort towards validation, governance and quality assurance. Rather than replacing software engineers, business analysts or testers, AI changes the nature of their work from content creation to critical evaluation and decision-making. Developers must review AI-generated implementations, testers must verify automatically generated test suites, business analysts must validate AI-produced requirements against stakeholder needs, and DevOps engineers must ensure that AI-generated deployment configurations comply with organisational policies. Consequently, the productivity gains achieved through AI generation may be partially offset by the additional effort required to review, verify and govern AI-generated artefacts before they are incorporated into production systems (Ding et al., 2026; OpenAI, 2026).

The emergence of 'vibe coding', in which developers rely heavily on AI to generate substantial portions of an application through natural language prompts, further illustrates these challenges. When AI lacks sufficient contextual information to produce a complete implementation, it may generate placeholder methods, mock services or simulated responses that satisfy the requested prompt without representing production-ready functionality. While these outputs accelerate prototyping, they may also introduce technical debt if they are not identified before deployment. Consequently, organisations should avoid delegating large, complex

software implementations to a single AI interaction. Instead, software should be developed incrementally, allowing developers to review smaller units of AI-generated code before integration. This approach aligns with agile software engineering principles by reducing complexity, improving traceability and enabling earlier defect detection (Beck et al., 2001).

The findings of this analysis suggest that the most effective use of Agentic AI is not to replace human judgement but to augment it through structured governance. One practical approach is to employ multiple AI systems with complementary objectives. For example, one AI agent may generate source code while a second independent AI agent performs code review, security analysis or test generation using adversarial prompts designed to identify defects. This multi-agent verification process introduces an additional layer of quality assurance by reducing reliance on the outputs of a single model. Nevertheless, the final responsibility for software quality, safety and compliance should remain with human members of the Scrum team. Therefore, the future role of software professionals is likely to evolve from primarily producing software towards supervising, validating and governing increasingly autonomous AI systems operating throughout the software development lifecycle.

In conclusion, the evidence demonstrates that Agentic AI is not replacing Scrum teams but fundamentally redefining how they operate. Rather than functioning as isolated specialists working within fixed working hours, Scrum professionals increasingly supervise autonomous AI agents capable of supporting continuous software engineering throughout the software development lifecycle. This transformation requires new approaches to governance, infrastructure investment and workforce development if organisations are to realise the productivity benefits of Agentic AI while maintaining trust, accountability and sustainable innovation.

References

Acemoglu, D., & Johnson, S. (2023). *Power and progress: Our thousand-year struggle over technology and prosperity*. PublicAffairs.

Anthropic. (2025). *Building effective AI agents*. Anthropic.
Anthropic – Building effective agents

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., & Thomas, D. (2001). *Manifesto for Agile Software Development*.
Agile Manifesto

Bommasani, R., Hudson, D. A., Adeli, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv*. https://doi.org/10.48550/arXiv.2108.07258

Brynjolfsson, E., Li, D., & Raymond, L. R. (2023). *Generative AI at work*. National Bureau of Economic Research. https://doi.org/10.3386/w31161

Bubeck, S., Chandrasekaran, V., Eldan, R., et al. (2023). Sparks of artificial general intelligence: Early experiments with GPT-4. *arXiv*.
https://doi.org/10.48550/arXiv.2303.12712

Department for Science, Innovation and Technology. (2026). *Guidance for using the AI Management Essentials tool*. UK Government.

Ding, T., Nannapaneni, A., Liu, B., & Zhang, L. (2026). Always-on agents: A survey of persistent memory, state, and governance in LLM agents. *arXiv*.
https://doi.org/10.48550/arXiv.2606.30306

Fu, M., Pasuksmit, J., & Tantithamthavorn, C. (2025). AI for DevSecOps: A landscape and future opportunities. *ACM Transactions on Software Engineering and Methodology, 34*(4), Article 115. https://doi.org/10.1145/3712190

Geuenich, J., Joublin, F., Ceravola, A., Brandenburg, S., & Dodge, J. (2026). AI can see what you can't see: How LLM-agents complement human-based gender-inclusive usability testing. *ACM Transactions on Interactive Intelligent Systems, 16*(2), Article 15, 1–34.
https://doi.org/10.1145/3801978

Goldman Sachs. (2026). *Decoding the agentic economy: The coming inflection in AI usage and margins*. Goldman Sachs Global Investment Research.

International Energy Agency. (2025). *Energy and AI*. IEA.

International Institute of Business Analysis. (2023). *A guide to the business analysis body of knowledge (BABOK Guide)* (Version 3). IIBA.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y., Madotto, A., & Fung, P. (2023). Survey of hallucination in natural language generation. *ACM Computing Surveys, 55*(12), 1–38. https://doi.org/10.1145/3571730

Johnston, D., Holtz, D., Richmond, A. M., Ong, C., Tambe, P., & Chatterji, A. (2026). The shift to agentic AI: Evidence from Codex. *arXiv*. https://arxiv.org/abs/2606.26959

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W. T., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems, 33*, 9459–9474.

Li, P., Yang, J., Islam, M. A., & Ren, S. (2023). Making AI less thirsty: Uncovering and addressing the secret water footprint of AI models. *arXiv*. https://doi.org/10.48550/arXiv.2304.03271

Lund, H., Østergaard, P. A., Sorknæs, P., Nielsen, S., Skov, I. R., Yuan, M., Thellufsen, J. Z., Mathiesen, B. V., Benson, S. M., Jentsch, A., Zhang, X., Werner, S., Wiltshire, R., & Möller, B. (2025). District heating in clean energy systems. *Nature Reviews Clean Technology, 1*, 532–546. https://doi.org/10.1038/s44359-025-00076-8

McKinsey & Company. (2023). *The economic potential of generative AI: The next productivity frontier*. McKinsey Global Institute.

National Institute of Standards and Technology. (2023). *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. U.S. Department of Commerce. https://doi.org/10.6028/NIST.AI.100-1

Norman, D. A. (2013). *The design of everyday things* (Revised and expanded ed.). Basic Books.

Organisation for Economic Co-operation and Development. (2024). *OECD Employment Outlook 2024: The net-zero transition and the labour market*. OECD Publishing. https://doi.org/10.1787/ac8b3538-en

OpenAI. (2025). *GPT-4.1 system card*. OpenAI.

OpenAI. (2026). *How AI is expanding what people do at work*. OpenAI.

Ouyang, L., Wu, J., Jiang, X., et al. (2022). Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems, 35*, 27730–27744.

Pearce, H., Ahmad, B., Tan, B., Dolan-Gavitt, B., & Karri, R. (2022). Asleep at the keyboard? Assessing the security of GitHub Copilot's code contributions. In *2022 IEEE Symposium on Security and Privacy Workshops (SPW)*. IEEE.

Sauvola, J., Tarkoma, S., Klemettinen, M., Riekki, J., & Doermann, D. (2024). Future of software development with generative AI. *Automated Software Engineering, 31*, Article 26. https://doi.org/10.1007/s10515-024-00426-z

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide: The definitive guide to Scrum: The rules of the game*. Scrum.org.

Schwartz, R., Dodge, J., Smith, N. A., & Etzioni, O. (2020). Green AI. *Communications of the ACM, 63*(12), 54–63. https://doi.org/10.1145/3381831

Sorknæs, P., Nielsen, S., Lund, H., Mathiesen, B. V., Moreno, D., & Thellufsen, J. Z. (2022). The benefits of 4th generation district heating and energy efficient datacentres. *Energy, 260*, 125215. https://doi.org/10.1016/j.energy.2022.125215

Tosi, D. (2024). Studying the quality of source code generated by different AI generative engines: An empirical evaluation. *Future Internet, 16*(6), 188. https://doi.org/10.3390/fi16060188

United Nations. (2015). *Transforming our world: The 2030 Agenda for Sustainable Development*. United Nations.

World Economic Forum. (2025). *The Future of Jobs Report 2025*. World Economic Forum.

Xu, A. Z., Zhang, K., Bao, L., Wang, Y., & Wei, W. (2026). Beyond codification: Generative AI and the integration of tacit knowledge in organizations. *Academy of Management Proceedings, 2026*. https://doi.org/10.5465/AMPROC.2026.286bp