

Engineering Better Requirements: Understanding the Impact of GenAI on Task Performance and Quality in Requirements Engineering

Benedikt Bluemelhuber
 Technical University of Munich
benedikt.bluemelhuber@tum.de

Sebastian Junker
 Technical University of Munich
sebastian.junker@wi.tum.de

Abstract

In order to deliver high-quality software systems, organizations utilize Requirements Engineering (RE) as a foundation for aligning development with stakeholder needs. With advances in Generative Artificial Intelligence (GenAI), potential emerges to augment RE processes to improve both task completion time and requirements quality. As GenAI-powered tools become more common in industry practice, our research examines under which circumstances GenAI supports RE tasks. Through our online experiment with 41 RE professionals from a manufacturing company, we demonstrate that GenAI assistance significantly reduces task completion time across different complexity levels and improves quality, particularly in simpler tasks. These results indicate that while GenAI effectively enhances RE efficiency in all contexts, its contribution to quality varies with task complexity. This suggests that organizations should strategically implement GenAI tools in RE workflows, recognizing both their productivity benefits and the continued importance of human expertise for more complex requirement scenarios.

Keywords: Generative Artificial Intelligence, Requirements Engineering, Online Experiment

1. Introduction

The software industry continues to expand, with increasingly complex systems demanding methods to ensure quality and alignment with stakeholder needs (Gurcan et al., 2022). Requirements Engineering (RE) is a critical, knowledge-intensive phase in software development, encompassing elicitation, analysis, specification, and validation of software requirements

(Pohl, 1996). Despite well-established frameworks and best practices, RE remains laborious and error-prone, often constrained by limited time (Bokhari & Siddiqui, 2011). Recent breakthroughs in Generative Artificial Intelligence (GenAI) offer promising benefits to assist employees in their daily work and drive firm performance (Fosso Wamba et al., 2024; Strobel et al., 2024). Large Language Models (LLMs) such as GPT-4 (OpenAI, 2024) represent a significant advancement in computational capabilities, demonstrating proficiency in processing and generating human-like text (Bruhin et al., 2024; Dwivedi et al., 2023). Within software development environments, these technologies have shown practical utility across numerous tasks, from code generation to testing code. Empirical evaluations of GenAI implementations in development workflows indicate measurable improvements in task completion and efficiency metrics (Mehler & Krautter, 2024; Peng et al., 2023). Given the promising outcomes demonstrated by GenAI in software development, investigating its applicability and efficacy within the context of RE warrants particular consideration (Cheng et al., 2024). While current research in this area has primarily focused on the feasibility of automated generation for requirements documentation (Barenkamp et al., 2020; Calegario et al., 2023), research lacks an understanding of how these technologies might impact the overall quality of requirements (Cheng et al., 2024). In this paper, we investigate whether GenAI can enhance RE activities in terms of both efficiency and quality. We address the following research questions: (1) To what extent does GenAI assistance reduce the time required to complete RE tasks? and (2) What effect does GenAI assistance have on the quality of outputs across different levels of task complexity? To address these questions, we conducted an online experiment

with 41 RE professionals from a global manufacturing company. Participants completed six RE tasks of varying complexity levels both with and without GenAI assistance while we measured completion times and assessed output quality through expert evaluations. Our findings contribute to the understanding of how GenAI tools can be effectively integrated into RE workflows.

2. Theoretical Background

2.1. Requirements Engineering: Processes and Metrics

RE is viewed as the initial phase of the software development life cycle (SDLC), where informal ideas are translated into formal specifications. During RE, both functional and non-functional requirements that the system must fulfill are gathered and documented in a requirements specification. RE comprises the elicitation, analysis, specification, and validation of software requirements (Pohl, 1996). This multi-disciplinary process involves stakeholders from both business and technical domains, each bringing unique priorities and perspectives (Bokhari & Siddiqui, 2011). By translating stakeholder needs into implementable software specifications, RE aims to ensure that the final product aligns with intended functionalities and constraints (Nuseibeh & Easterbrook, 2000).

RE tasks vary in their degree of complexity. Simpler tasks may involve enumerating functional requirements and verifying them against basic constraints. More complex activities require detecting ambiguities, resolving inconsistencies across stakeholder viewpoints, and integrating domain-specific knowledge (Bokhari & Siddiqui, 2011; Fantechi et al., 2023). The latter can be especially time-intensive and prone to errors, highlighting the importance of clear guidance and systematic methods throughout RE activities. To gauge the success of RE, organizations typically employ both quantitative and qualitative metrics. Quantitative measures include time-to-completion, requirements coverage, and defect rates to ensure that deadlines are met and important features are not overlooked (Bokhari & Siddiqui, 2011). Qualitative assessment involves evaluating whether the requirements adequately capture stakeholder needs and domain constraints (Costello & Liu, 1995; Marques et al., 2024). Early detection of ambiguous or hidden assumptions can prevent costly rework in later stages of software development (Fantechi et al., 2023). Recent work emphasizes the need for novel approaches that lighten the load on requirements engineers without compromising output quality (Cheng et al., 2024).

2.2. Generative AI in Software Engineering and its application in RE

GenAI refers to artificial intelligence techniques designed to produce novel outputs based on patterns observed in training data (Feuerriegel et al., 2024). The approach leverages LLMs, such as GPT-4 (OpenAI, 2024) or Gemini (Team et al., 2023), which generate coherent and contextually relevant text. Tools such as ChatGPT and Microsoft's Copilot have popularized the use of natural-language prompts, thus broadening the technology's reach to a wider pool of SE practitioners (Dell'Acqua et al., 2023; Gattupalli et al., 2023; Strobel et al., 2024).

Within the SDLC, early applications of GenAI have focused on coding tasks, test-case generation, and debugging routines (Nguyen-Duc et al., 2023). Such use cases demonstrate efficiency gains by automating repetitive work and improving consistency. Nonetheless, challenges remain concerning data privacy, hallucinations (where the AI produces incorrect yet plausible information), and intellectual property compliance (Ji et al., 2023; Smits & Borghuis, 2022). Governance mechanisms and ethical standards need to keep pace with the evolving capabilities of these models, especially when deployed in enterprise environments handling sensitive or proprietary codebases (Smits & Borghuis, 2022). As GenAI matures, it is increasingly applied to the language-intensive tasks of RE (Bruhin, 2024; Nguyen-Duc et al., 2023). For instance, LLM-based tools can aid in elicitation by brainstorming potential user needs or highlighting missing features, thereby expanding the initial scope of requirement statements (Ronanki et al., 2023). During analysis, researchers have found that AI can assist by detecting ambiguities and inconsistencies in requirement documents (Cheng et al., 2024; Murugesan & Cherukuri, 2023), potentially serving as a virtual reviewer.

The use of GenAI has demonstrated productivity benefits across software development contexts. Ziegler et al. (2024) found that GitHub Copilot users experienced productivity gains, while Unzicker et al. (2024) observed that GenAI-Assisted code reviews identified more issues in less time. These efficiency improvements in related software engineering contexts suggest similar benefits could be realized in RE processes, leading us to our first hypothesis:

H1: *Using GenAI for RE tasks decreases task completion time.*

However, challenges arise in ensuring the reliability and interpretability of GenAI-generated outputs in RE settings (Ji et al., 2023). Complex requirements may

demand nuanced domain expertise that exceeds the scope of generalized language models (Nguyen-Duc et al., 2023). Models may also produce factually incorrect or fictitious content (“hallucinations”), posing risks if overlooked during the specification process. Despite these concerns, several studies underscore GenAI’s promise for transforming informal or incomplete user inputs into consistent requirement formats (Calegario et al., 2023), uncovering hidden constraints (Fantechi et al., 2023), and streamlining ongoing requirements management (Gabriel, 2024). These quality enhancements suggest our second hypothesis:

H2: *Using GenAI for RE results in higher-quality requirement descriptions.*

3. Methodology

To investigate how GenAI affects both task completion time and output quality in RE tasks, we conducted an online experiment. This design enabled us to collect and analyze objective data on time metrics and quality ratings while comparing performance with and without GenAI assistance. In the field of IS research, this approach is commonly used in similar contexts (Mehler & Krautter, 2024; Turel et al., 2008).

3.1. Research Design and Experimental Setup

All participants completed six RE tasks, designed to test our hypotheses across three complexity levels: easy, medium, and complex (see Table 1). To ensure these tasks accurately represented authentic RE challenges, we developed them through an iterative collaborative process with two experienced practitioners (eight and nine years of industry experience). The collaboration involved multiple review sessions where the research team proposed initial task concepts, followed by expert feedback on their relevance, difficulty, and alignment with practical RE challenges.

Simple tasks required participants to formulate requirements for everyday products (e.g., forks, hammers), establishing baseline requirements formulation skills. Medium tasks presented flawed software system requirements containing ambiguities or contradictions typical in embedded systems, requiring identification and correction of specification errors. The complex tasks required detailed descriptions of the case study partner’s specific products, including their embedded software components, where participants identified and formalized implicit software and hardware requirements. Each task was presented on a dedicated survey page, featuring a visible timer and an open-text response field. To enhance time measurement

accuracy, participants manually entered the time spent on each task while a hidden script concurrently tracked page load and submission timestamps. This dual-recording method mitigated the effects of potential interruptions or user underestimation, yielding more reliable measures of actual task engagement.

To systematically evaluate response quality, we implemented a structured assessment applied consistently across all task types. The evaluation was conducted by two expert raters with more than ten years of experience who independently scored each response. The responses are rated on a 7-point Likert scale (Joshi et al., 2015). For all tasks, quality ratings were based on five standardized dimensions and defined with the manufacturing company based on ISO/IEC/IEEE 29148:2018 (*ISO/IEC/IEEE 29148:2018(E): Systems and software engineering — Life cycle processes — Requirements engineering*, 2018) following established RE best-practices. Based on our design, this included: consistency, active voice usage, atomicity (one requirement per sentence or work item), absence of negation, and testability. Table 1 provides a comprehensive summary of each task characteristic, its associated descriptions, and the evaluation criteria applied.

An exemplary screenshot of our online interface for the Simple Task without GenAI assistance is shown in Figure 1. At the top, participants encounter the outlined task with a similar example below to ensure that the task is clear. Right below, the text-entry field is located where participants’ put responses or revision. To continue with the next task, participants are prompted to record their completion time in seconds based on an on-screen timer at the top.

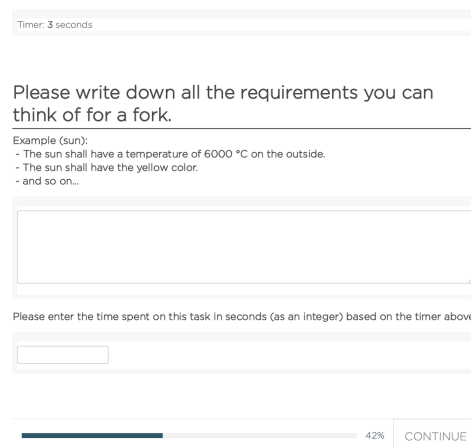


Figure 1. Illustration of the online interface for Task 1

After the pre-testing phase with two employees not

Table 1. Overview of Task Characteristics and Descriptions

Complexity	Task Activity	Application Description
EASY	Enumerating requirements for given tasks	Generate comprehensive requirements for common physical objects
MEDIUM	Reviewing flawed requirements for given tasks	Analyze embedded system control specifications (sound/ illumination indicators) for requirement engineering violations
COMPLEX	Detection of hidden constraints from textual description for given tasks	Perform requirements elicitation from consumer product system descriptions containing multiple subsystems, user interfaces, and technical constraints

being part of the experiment who provided feedback on task clarity and interface design, we implemented the online experiment using Unipark, a GDPR-compliant platform that fulfilled our requirements. Participants received information about data privacy policies and their GDPR rights before beginning the experiment. The experiment began with an introduction to the RE tasks, as shown in Figure 2, followed by random assignment to one of the two experimental sequences to mitigate potential learning effects, resulting in 20 participants beginning with Non-GenAI-Assisted tasks and 21 participants starting with GenAI-Assisted tasks.

Participants were then presented with their first set of tasks according to their assigned group. Throughout the experiment, tasks followed the same difficulty progression (simple, medium, complex) regardless of whether participants were in the Non-GenAI-Assisted or GenAI-Assisted condition.

Each task was presented on a dedicated page following the interface layout shown in Figure 1. For the GenAI-Assisted tasks, participants had access to ChatGPT through a separate browser window and were encouraged to use it to support task completion. We deliberately chose naturalistic interaction without standardized prompts to capture how RE professionals currently engage with GenAI tools in practice, recognizing this represents a trade-off between ecological validity and mechanistic understanding of prompt variation. Upon completion of all six tasks, participants were asked to provide demographic information, including age group, gender, highest level of education, years of experience in RE, and self-assessed skills with GenAI tools.

3.2. Data Collection and Analysis

We gathered data through a collaboration with a manufacturing company. We employed a purposive sampling approach, targeting employees whose job

descriptions explicitly included RE responsibilities. Participants were defined as eligible based on their formal roles in requirements-related activities such as elicitation, documentation, review, or approval of requirements specifications and be fluent in English.

This resulted in an initial pool of 127 potential participants who received email invitations containing the link to the experiment. The online experiment remained accessible for nine days. Of the participants who engaged with the experiment, 90% successfully passed attention checks embedded within the tasks, resulting in 41 complete and valid responses from participants across various company departments and locations. Participation was entirely voluntary and uncompensated. Completion times ranged from 18 to 43 minutes. Participants’ demographic characteristics are summarized in Table 2.

When asked about the frequency of GenAI tool usage, participants reported using these tools daily (12.2%), weekly (19.5%), monthly (9.76%), occasionally (34.15%), or rarely (24.39%). Nearly half of participants rated their GenAI tool proficiency as basic (48.78%), with about a quarter reporting intermediate skills (24.39%). Approximately 22% had no skills, and advanced or expert proficiency was rare at 2.44% each.

Table 2. Demographic Data (Age and Gender) of the Participants

Demography	Categories	Frequency
Gender	Female	10 (24.3%)
	Male	30 (73.2%)
	Prefer not to say	1 (2.4%)
Age Group	25 - 34	10 (24.4%)
	35 - 44	22 (53.6%)
	45 - 54	6 (14.6%)
	55 and above	3 (7.3%)

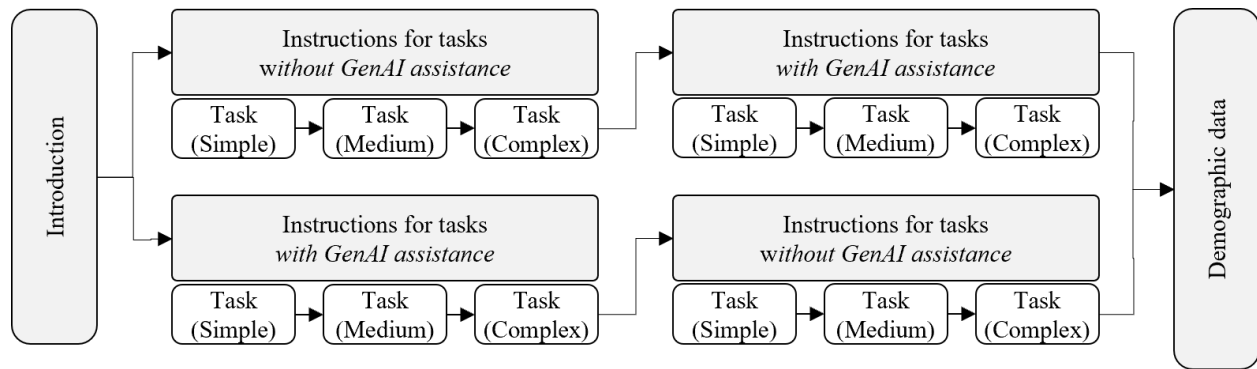


Figure 2. Flow for Experimental setup

4. Results

In this section, we present the findings from our online experiment on the impact of GenAI assistance on task completion time and output quality in RE activities. For the evaluation of quality, two expert raters who evaluated responses against predefined criteria. For the task completion time, we followed the metrics used in literature for evaluating task performance (Dell’Acqua et al., 2023; Noy & Zhang, 2023).

4.1. Inter-rater Reliability and Descriptive Overview

To assess the quality of participants’ responses to the open-ended RE tasks, two experienced RE experts independently evaluated each submission. Quality ratings were assigned using a 7-point Likert scale, where 1 represented the lowest quality and 7 represented the highest (Femmer, 2017; Joshi et al., 2015). Given the subjective judgment required to score open-ended responses, we computed the Intraclass Correlation Coefficient (ICC), specifically the ICC(3,1) model (Shrout & Fleiss, 1979), to evaluate inter-rater reliability. This analysis yielded an ICC(3,1) value of 0.699, with a 95% confidence interval of [0.629, 0.758] (see Table 3). According to established guidelines (Cicchetti, 1994; Koo & Li, 2016), this value indicates moderate to good agreement, supporting the robustness of our expert evaluations.

Table 3. Intraclass Correlation

Type	Point Est.	Lower 95% CI	Upper 95% CI
ICC(3,1)	0.699	0.629	0.758

Note. 246 points and 2 raters/measurements. ICC type as referenced by Shrout and Fleiss, 1979.

Prior to hypothesis testing, we examined the

distributions of our measures across all conditions. Table 4 presents descriptive statistics for task completion time and quality ratings across all complexity levels. The data demonstrate a consistent pattern where all task types show faster completion times under GenAI-Assisted compared to Non-GenAI-Assisted conditions. For quality ratings, the effect varies by task complexity, with the most pronounced improvement in simple tasks, while medium and complex tasks show modest differences.

4.2. Hypothesis Testing and Detailed Analysis

Before conducting our statistical tests, we first assessed whether our data met the assumptions for parametric tests. The Shapiro-Wilk test revealed significant deviations from normality ($p < .001$) in both time measurements and quality ratings across multiple task conditions. Given these violations and our repeated measures design, we employed Wilcoxon Signed-Rank Tests for comparing matched pairs of tasks under Manual RE and GenAI-Assisted conditions (Hollander et al., 2013; Wilcoxon, 1945). To quantify the magnitude of observed differences, we calculated the Rank-Biserial correlation coefficient (r_{rb}) as our effect size measure. Following established guidelines from (Cohen, 2013), we interpret effect sizes as small ($r_{rb} = 0.2$), medium ($r_{rb} = 0.5$), or large ($r_{rb} = 0.8$).

To systematically investigate our hypotheses, we conducted pairwise comparisons between Non-GenAI-Assisted and GenAI-Assisted conditions for each task complexity level. Table 5 summarizes the results with detailed statistics for each comparison.

Regarding our first hypothesis (H1), which proposed that GenAI assistance reduces task completion time in RE activities, we found consistent evidence across all complexity levels. For simple tasks, the completion time with GenAI assistance was significantly shorter than with manual RE ($p < .001$), with participants

Table 4. Descriptive Statistics for Task Completion Time and Quality Ratings

Task Type	Measure	Time (sec)		Quality (1–7)	
		Manual RE	GenAI-Assisted	Manual RE	GenAI-Assisted
Simple	Mean (SD)	265.3 (147.6)	184.7 (98.3)	4.2 (1.3)	4.9 (1.1)
	Median	217	156	4.0	5.0
Medium	Mean (SD)	357.8 (175.2)	246.9 (129.4)	3.7 (1.5)	3.7 (1.4)
	Median	312	203	3.5	3.5
Complex	Mean (SD)	428.6 (203.5)	318.2 (166.8)	3.2 (1.4)	3.6 (1.3)
	Median	394	275	3.0	3.5

completing tasks on average 80.6 seconds faster with GenAI support. This difference yielded a moderate to large effect size ($r_{rb} = 0.55$). For medium-complexity tasks, the time advantage of GenAI assistance was even more pronounced ($p < .001$), with an average reduction of 110.9 seconds and a perfect effect size ($r_{rb} = 1.00$). Similarly, for complex tasks, the time with GenAI assistance was significantly shorter ($p < .001$), with participants completing tasks 110.4 seconds faster on average and an effect size approaching perfection ($r_{rb} = 0.99$). The consistent and substantial time advantages across all complexity levels provide support for **H1**.

Concerning our second hypothesis (H2), which proposed that GenAI assistance leads to higher quality requirements, we observed complexity-dependent effects. For simple tasks, quality ratings were higher with GenAI assistance ($p = .012$), showing an average improvement of 0.7 points on the 7-point scale with a moderate effect size ($r_{rb} = 0.35$). This indicates that for straightforward requirements tasks, GenAI provides meaningful quality improvements. For medium-complexity tasks, however, no significant difference in quality was detected ($p = .990$), with identical mean ratings (3.7) for both conditions and a negligible effect size ($r_{rb} = 0.10$). This suggests that for tasks of medium complexity, GenAI assistance neither improved nor diminished the quality of outputs. For complex tasks, we observed a positive trend toward higher quality with GenAI assistance, with an average improvement of 0.4 points, but this difference did not reach statistical significance at the conventional threshold ($p = .090$). The effect size ($r_{rb} = 0.21$) falls in the small to moderate range, suggesting potential benefits that warrant further investigation with larger samples. The varying effects across complexity levels provide partial support for **H2**.

5. Discussion

RE plays a critical role in software development, providing the foundation for aligning system

capabilities with stakeholder needs (Pohl, 1996). As software complexity increases, so do the challenges of documenting, validating, and refining requirements. This study examined how GenAI affects performance across RE tasks of varying complexity and the quality of resulting outputs. Our online experiment found consistent evidence that participants using GenAI completed RE tasks in notably less time than those using manual methods alone. This strengthens the findings of related research tapping into GenAI usage along the SDLC (Mehler & Krautter, 2024; Ziegler et al., 2024). This acceleration was most apparent in medium and complex tasks, suggesting that AI-driven support yields high returns in domains where large texts need to be processed and worked through (Eloundou et al., 2024). This allows analysts to reduce workload on tasks, thus allowing more time for innovation (Alavi & Leidner, 2024; Benbya et al., 2024). Furthermore, our empirical results demonstrate that GenAI can enhance output quality under specific conditions. Consistent with existing AI literature (Noy & Zhang, 2023), simpler tasks showed pronounced improvement with GenAI assistance. Participants completed these tasks faster, and their requirement statements exhibited greater clarity and consistency. For complex tasks requiring deeper domain knowledge, however, quality improvements were less uniform. This aligns with observations that tasks within AI capabilities ("inside-the-frontier") see significant productivity gains, while tasks beyond current AI capabilities ("outside-the-frontier") may yield reduced performance (Dell'Acqua et al., 2023). The selected tasks for RE appear to straddle this technological frontier, with GenAI effectively streamlining routine aspects but showing limitations when deeper domain knowledge or implicit constraints are involved (Dell'Acqua et al., 2023). This discrepancy likely stems from the increased need for domain knowledge and critical reflection (Cheng et al., 2024). Even with GenAI assistance in drafting or refining requirements, analysts' domain expertise and ability to interpret context-specific

Table 5. Wilcoxon Signed-Rank Test Results for Hypothesis Testing

Hypothesis	Compared Conditions	Wilcoxon Test	Difference of means, Effect size r_{rb}	Outcome
H1: Using GenAI for Requirements Engineering tasks decreases task completion time.	Simple: Manual vs. GenAI	$W = 667,$ $z = 3.07,$ $p < .001$	80.6 seconds, $r_{rb} = 0.55$	H1 supported
	Medium: Manual vs. GenAI	$W = 861,$ $z = 5.58,$ $p < .001$	110.9 seconds, $r_{rb} = 1.00$	
	Complex: Manual vs. GenAI	$W = 859,$ $z = 5.55,$ $p < .001$	110.4 seconds, $r_{rb} = 0.99$	
H2: Using GenAI for Requirements Engineering results in higher-quality requirement descriptions.	Simple: Manual vs. GenAI	$W = 154.5,$ $z = -2.25,$ $p = .012$	-0.7 points, $r_{rb} = 0.35$	H2 partially supported
	Medium: Manual vs. GenAI	$W = 554,$ $z = 2.29,$ $p = .990$	0.0 points, $r_{rb} = 0.10$	
	Complex: Manual vs. GenAI	$W = 219,$ $z = -1.34,$ $p = .090$	-0.4 points, $r_{rb} = 0.21$	

constraints remain critical for complex tasks (Fügner et al., 2021). Our data indicates that complex RE tasks yielded no significant quality improvements when using GenAI, a pattern that corresponds with participant proficiency levels. Most surveyed RE professionals possessed substantial industry experience, yet nearly half reported only basic GenAI proficiency, with approximately one-fifth lacking any GenAI skills. This skill distribution, combined with predominantly infrequent usage patterns, suggests that the absence of quality gains in complex tasks may stem from insufficient prompting expertise rather than technological constraints. These findings underscore the critical role of skill development for GenAI in organizational processes. Organizations can counteract this with building dedicated initiatives aiming to upskill employees and building AI literacy (Brynjolfsson et al., 2023; Knoth et al., 2024).

6. Contributions

Our study provides several contributions to both theory and practice in the field of RE and GenAI adoption. From a theoretical perspective, we offer empirical evidence quantifying GenAI's impact on RE processes, aligning with existing research for efficiency gains in software development (Mehler & Krautter, 2024; Unzicker et al., 2024). Our results

demonstrate significant time reductions across all task complexity levels, contextualizing these benefits specifically within RE workflows. Thus, we align with the broader GenAI research landscape, as we establish that GenAI delivers task performance advantages in RE activities. Second, we extend understanding of the relationship between task complexity and GenAI efficacy in knowledge-intensive work. The quality improvements in simple tasks compared to modest gains in complex tasks suggest that GenAI capabilities intersect differently across the RE spectrum (Dell'Acqua et al., 2023). In addition for practitioners, our research offers actionable insights for implementing GenAI in RE processes. Organizations can achieve substantial time savings without compromising output quality, providing the possibility to enhance productivity in early software development phases. Our findings help prioritize where to deploy GenAI within RE workflows, suggesting organizations initially target simpler requirements tasks where both time savings and quality improvements are most pronounced. For complex tasks involving implicit constraints or domain-specific knowledge, GenAI remains valuable for time reduction but should be deployed with human oversight. Finally, our results suggest the importance of developing appropriate prompting techniques (Gattupalli et al., 2023). The limited quality improvements in complex tasks despite time savings

combined, suggesting organizations should invest in training requirements engineers both in tool operation and in developing critical evaluation skills needed to assess and refine GenAI outputs.

7. Limitations and Future Research

This study provides valuable insights into GenAI's impact on RE tasks, yet several limitations should be acknowledged to guide future research. While we designed our experiment to approximate real-world RE tasks, the use of Unipark as a tool to present tasks and collect responses may have influenced participants' performance. Although this approach ensured consistent and reliable data collection, it differs from typical RE environments where professionals use specialized requirements management tools and collaborate with stakeholders in real time. We acknowledge that an experimental setup, while necessary for controlled measurement, may not capture the iterative and collaborative nature of RE processes, and our within-subjects design may introduce learning or carryover effects despite counterbalancing efforts.

A further limitation is that the evaluation was conducted by only two experts. This limited number of evaluators may have introduced potential bias in quality assessment, as different experts might have varying perspectives on requirements quality and appropriateness. Additionally, our data collection was conducted as an online-only experiment, which may have introduced variability due to differences in participants' working environments, potential distractions. These factors could have influenced both time measurements and quality outcomes in ways difficult to quantify. The visibility of timing mechanisms during the experiment may also have created performance pressure not typically present in RE activities. Our sample size of 41 professionals, while statistically sufficient to detect significant effects in our experimental design, represents a limitation for broader generalizability. Furthermore, our experimental design employed simple randomization without blocking or stratification across task types or participant characteristics. The participating professionals from a single organization in the manufacturing sector constrains our ability to extend findings to larger populations and diverse organizational contexts. Different industries face unique regulatory landscapes, technical constraints, and organizational cultures that shape RE practices. Our findings may not transfer seamlessly to these varied contexts (Ramesh et al., 2010). Furthermore, our tasks may have simplified the complex reality of RE. RE activities

frequently span longer time frames and involve multiple stakeholders and interdependencies that could not be replicated in our online-experiment. This simplification may overestimate the benefits of GenAI for more complex, domain-specific requirements. The absence of standardized prompts or prompt logging, while preserving ecological validity, limits our ability to attribute performance variations to specific prompting strategies. We cannot determine whether participants who achieved better outcomes used more sophisticated prompting techniques or iterative refinement approaches. This represents an important trade-off: our results reflect realistic baseline performance in current industry practice but cannot identify optimal prompting patterns that might further enhance GenAI effectiveness in RE tasks.

As we have found that GenAI can reduce task completion time and improve output quality in certain RE contexts, we encourage future research to further evaluate the generalizability of our findings while also examining ways in which these tools integrate with professional RE practices (Weritz et al., 2024). Concerning the generalizability of our findings, future studies could evaluate the impact of GenAI on requirements engineers by conducting longitudinal experiments in real-world environments across industries and organizational contexts. This would help establish whether the observed benefits persist across varying domains with different regulatory requirements, technical constraints, and organizational cultures. The efficacy of individuals on the usage of GenAI tools influences associated outcomes (Weritz et al., 2024). Therefore, investigating how varying skill levels affect performance would provide deeper insights into the human factors that shape the effective integration of GenAI. Research could therefore explore how job requirements and skill profiles for professionals are transforming with GenAI. The required competencies likely shift from traditional documentation and elicitation skills toward prompt engineering and Human-AI collaboration.

References

- Alavi, M., & Leidner, D. E. (2024). Knowledge Management Perspective of Generative Artificial Intelligence. *Journal of the Association for Information Systems*, 25(1), 1–12.
- Barenkamp, M., Rebstadt, J., & Thomas, O. (2020). Applications of AI in classical software engineering. *AI Perspectives*, 2(1), 1.

- Benbya, H., Deakin University, Strich, F., Deakin University, Tamm, T., & Deakin University. (2024). Navigating Generative Artificial Intelligence Promises and Perils for Knowledge and Creative Work. *Journal of the Association for Information Systems*, 25(1), 23–36.
- Bokhari, M., & Siddiqui, S. (2011). Metrics for requirements engineering and automated requirements tools.
- Bruhin, O. (2024). Beyond Code: The Impact of Generative AI on Work Systems in Software Engineering. *ICIS 2024 Proceedings*, (15).
- Bruhin, O., Ebel, P., Mueller, L., & Li, M. (2024). GenAI and Software Engineering: Strategies for Shaping the Core of Tomorrow's Software Engineering Practice. *ICIS 2024 Proceedings*, (7).
- Brynjolfsson, E., Li, D., & Raymond, L. (2023). Generative AI at Work. <https://doi.org/10.48550/ARXIV.2304.11771>
- Calegario, F., Burégio, V., Erivaldo, F., Andrade, D. M. C., Felix, K., Barbosa, N., Lucena, P. L. d. S., & França, C. (2023). Exploring the intersection of Generative AI and Software Development. <https://doi.org/10.48550/arXiv.2312.14262>
- Cheng, H., Husen, J. H., Lu, Y., Racharak, T., Yoshioka, N., Ubayashi, N., & Washizaki, H. (2024). Generative AI for Requirements Engineering: A Systematic Literature Review. <https://doi.org/10.48550/ARXIV.2409.06741>
- Cicchetti, D. V. (1994). Guidelines, criteria, and rules of thumb for evaluating normed and standardized assessment instruments in psychology. *Psychological Assessment*, 6(4), 284–290.
- Cohen, J. (2013, May). *Statistical Power Analysis for the Behavioral Sciences* (0th ed.). Routledge.
- Costello, R. J., & Liu, D.-B. (1995). Metrics for requirements engineering. *Journal of Systems Software*, 29(1), 39–63.
- Dell'Acqua, F., McFowland, E., Mollick, E. R., Lifshitz-Assaf, H., Kellogg, K., Rajendran, S., Kraye, L., Candelon, F., & Lakhani, K. R. (2023). Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality. *SSRN Electronic Journal*.
- Dwivedi, Y. K., Kshetri, N., Hughes, L., Slade, E. L., Jeyaraj, A., Kar, A. K., Baabdullah, A. M., Koohang, A., Raghavan, V., Ahuja, M., et al. (2023). Opinion paper: “So what if ChatGPT wrote it?” Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. *International journal of information management*, 71, 102642.
- Eloundou, T., Manning, S., Mishkin, P., & Rock, D. (2024). GPTs are GPTs: Labor market impact potential of LLMs. *Science*, 384(6702), 1306–1308.
- Fantechi, A., Gnesi, S., Passaro, L., & Semini, L. (2023). Inconsistency Detection in Natural Language Requirements using ChatGPT: A Preliminary Evaluation. *2023 IEEE 31st International Requirements Engineering Conference (RE)*, 335–340.
- Femmer, H. (2017). *Requirements engineering artifact quality: Definition and control* [Doctoral dissertation]. Technische Universität München. <https://mediatum.ub.tum.de/1355823>
- Feuerriegel, S., Hartmann, J., Janiesch, C., & Zschech, P. (2024). Generative AI. *Business & Information Systems Engineering*, 66(1), 111–126.
- Fosso Wamba, S., Queiroz, M. M., Pappas, I. O., & Sullivan, Y. (2024). Artificial Intelligence Capability and Firm Performance: A Sustainable Development Perspective by the Mediating Role of Data-Driven Culture. *Information Systems Frontiers*, 26(6), 2189–2203.
- Fügener, A., Grahl, J., University of Cologne, Gupta, A., University of Minnesota, Ketter, W., University of Cologne, & Erasmus University. (2021). Will Humans-in-the-Loop Become Borgs? Merits and Pitfalls of Working with AI. *MIS Quarterly*, 45(3), 1527–1556.
- Gabriel, V. S. S. (2024). Generative AI: A Literature Review on Business Value. *AMCIS 2024 Proceedings*, (27).
- Gattupalli, S., Maloy, R., & Edwards, S. (2023). *Prompt Literacy: A Pivotal Educational Skill in the Age of AI* (tech. rep.). University of Massachusetts Amherst. <https://doi.org/10.7275/3498-WX48>
- Gurcan, F., Dalveren, G. G. M., Cagiltay, N. E., & Soylu, A. (2022). Detecting Latent Topics and Trends in Software Engineering Research Since 1980 Using Probabilistic Topic Modeling. *IEEE Access*, 10, 74638–74654.
- Hollander, M., Wolfe, D. A., & Chicken, E. (2013). *Nonparametric statistical methods*. John Wiley & Sons.
- ISO/IEC/IEEE 29148:2018(E): Systems and software engineering — Life cycle processes —*

- Requirements engineering* (Standard No. ISO/IEC/IEEE 29148:2018(E)). (2018). International Organization for Standardization; Institute of Electrical and Electronics Engineers.
- Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y. J., Madotto, A., & Fung, P. (2023). Survey of Hallucination in Natural Language Generation. *ACM Computing Surveys*, 55(12), 1–38.
- Joshi, A., Kale, S., Chandel, S., & Pal, D. (2015). Likert Scale: Explored and Explained. *British Journal of Applied Science & Technology*, 7(4), 396–403.
- Knonth, N., Tolzin, A., Janson, A., & Leimeister, J. M. (2024). Ai literacy and its implications for prompt engineering strategies. *Computers and Education: Artificial Intelligence*, 6, 100225.
- Koo, T. K., & Li, M. Y. (2016). A Guideline of Selecting and Reporting Intraclass Correlation Coefficients for Reliability Research. *Journal of Chiropractic Medicine*, 15(2), 155–163.
- Marques, N., Silva, R. R., & Bernardino, J. (2024). Using ChatGPT in Software Requirements Engineering: A Comprehensive Review. *Future Internet*, 16(6), 1–21.
- Mehler, M. F., & Krautter, K. (2024). Productivity vs. purpose: Generative AI Enhances Task Performance but Reduces Meaningfulness in Programming. *ECIS 2024 Proceedings*, (7).
- Murugesan, S., & Cherukuri, A. K. (2023). The Rise of Generative Artificial Intelligence and Its Impact on Education: The Promises and Perils. *Computer*, 56(5), 116–121.
- Nguyen-Duc, A., Cabrero-Daniel, B., Przybyłek, A., Arora, C., Khanna, D., Herda, T., Rafiq, U., Melegati, J., Guerra, E., Kemell, K.-K., et al. (2023). Generative Artificial Intelligence for Software Engineering—A Research Agenda. *arXiv Preprint arXiv:2310.18648*.
- Noy, S., & Zhang, W. (2023). Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654), 187–192.
- Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: A roadmap. *Proceedings of the Conference on The Future of Software Engineering*, 35–46.
- OpenAI. (2024). Introducing ChatGPT. <https://openai.com/blog/chatgpt>
- Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). The impact of AI on Developer Productivity: Evidence from GitHub Copilot. <https://arxiv.org/abs/2302.06590>
- Pohl, K. (1996). Requirement engineering: An overview. *Encyclopedia of Computer Science and Technology*, 36.
- Ramesh, B., Cao, L., & Baskerville, R. (2010). Agile requirements engineering practices and challenges: An empirical study. *Information Systems Journal*, 20(5), 449–480.
- Ronanki, K., Cabrero-Daniel, B., & Berger, C. (2023). ChatGPT as a tool for User Story Quality Evaluation: Trustworthy Out of the Box? <https://arxiv.org/abs/2306.12132>
- Shrout, P. E., & Fleiss, J. L. (1979). Intraclass correlations: Uses in assessing rater reliability. *Psychological Bulletin*, 86(2), 420–428.
- Smits, J., & Borghuis, T. (2022). Generative ai and intellectual property rights. In B. Custers & E. Fosch-Villaronga (Eds.), *Law and artificial intelligence: Regulating ai and applying ai in legal practice* (pp. 323–344). T.M.C. Asser Press.
- Strobel, G., Banh, L., Möller, F., & Schoormann, T. (2024). Exploring generative artificial intelligence: A taxonomy and types.
- Team, G., Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. (2023). Gemini: A family of highly capable multimodal models. *arXiv Preprint arXiv:2312.11805*.
- Turel, O., Yuan, Y., & Connelly, C. E. (2008). In Justice We Trust: Predicting User Acceptance of E-Customer Services. *Journal of Management Information Systems*, 24(4), 123–151.
- Unzicker, D., Mehler, M., Kammholz, L., Sturm, T., Jourdan, S., & Buxmann, P. (2024). All eyes on the reviewer: Understanding the impact of GenAI on mental workload and performance in code reviews. *ICIS 2024 Proceedings*, (4).
- Weritz, P., Wache, H., & Honigsberg, S. (2024). How Digital Readiness Relates to the Intention to Use Generative AI in Workplace Service Systems. *AMCIS 2024 Proceedings*, (5).
- Wilcoxon, F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6), 80–83. <https://doi.org/10.2307/3001968>
- Ziegler, A., Kalliamvakou, E., Li, X. A., Rice, A., Rifkin, D., Simister, S., Sittampalam, G., & Aftandilian, E. (2024). Measuring GitHub Copilot’s Impact on Productivity. *Communications of the ACM*, 67(3), 54–63.