The Use of Generative AI Tools by Requirements Engineers: An Interview with Industry Professionals

Josué V. Ferreira[1], **Carlos S. Portela**[2], **Sandro R. B. Oliveira**[1]

[1]Programa de Pós-Graduação em Computação (PPGCOMP)
Universidade Federal do Pará (UFPA) - Belém - PA

[2]Programa de Pós-Graduação em Computação Aplicada (PPCA)
Universidade Federal do Pará (UFPA) - Tucuruí - PA

josue.ferreira@icen.ufpa.br, {csp, srbo}@ufpa.br

Abstract. *This paper investigates how Generative Artificial Intelligence (GenAI) grounded in Large Language Models (LLMs) are being employed to support activities within Requirements Engineering (RE). Based on interviews conducted with software engineering professionals, the findings indicate that GenAI tools are particularly valuable during requirements elicitation and specification, contributing to improved productivity and reduced operational effort, especially when addressing Functional Requirements (FRs). Despite these benefits, participants also reported important limitations, including sensitivity to prompt formulation, generation of overly generic outputs, and limited capability to adequately support Non-Functional Requirements (NFRs).*

1. Introduction

Requirements Engineering (RE) is a fundamental component of software development, as the quality of the requirements produced directly influences the system's reliability, its maintainability, and its alignment with stakeholder expectations. Nevertheless, conducting RE activities, including elicitation, analysis, specification, prioritization, validation, and requirements management, remains a demanding process that requires substantial domain knowledge, stakeholder interaction, and continuous refinement of software artifacts. In response to these challenges, several approaches and computational tools have been proposed to support RE practices. Existing studies report solutions targeting activities such as requirements elicitation [Ronanki et al. 2023, Ren et al. 2024], requirements analysis [Mahbub et al. 2024], prioritization [Felfernig et al. 2018, Vijayakumar e Nethravathi 2021], and quality assessment of requirements artifacts [Parra et al. 2015].

More recently, advances in Artificial Intelligence (AI) have introduced new opportunities for automating and enhancing software engineering activities, particularly with the emergence of Generative Artificial Intelligence (GenAI) systems. Large Language Models (LLMs), including models such as GPT-4, have demonstrated the ability to process and generate textual content that is close to human-level quality, enabling new forms of support for Requirements Engineering tasks [Zhao et al. 2021, Ronanki et al. 2023]. Supported by Natural Language Processing (NLP) techniques, these models can assist in activities involving content generation, refinement, interpretation, and organization of software artifacts [Yu et al. 2024].

Although the use of GenAI in software engineering has rapidly expanded, important limitations and open questions remain regarding its effective application throughout the RE lifecycle [Zhao et al. 2021]. In particular, there is still limited understanding of how these technologies support Functional Requirements (FRs) and Non-Functional Requirements (NFRs) in practical contexts. Questions persist regarding which RE activities benefit most from LLM-based support, how practitioners formulate prompts to obtain reliable outputs, and how these tools contribute to properties such as consistency, completeness, traceability, and quality of requirements specifications.

In this scenario, the present study investigates the adoption of GenAI in RE from the perspective of software engineering professionals. Using an exploratory qualitative approach, the research examines how GenAI assistants are being incorporated into RE activities, with particular attention to their perceived impact on both FRs and NFRs.

2. Background

2.1. Requirements Engineering

According to [Sommerville 2018], RE activities are responsible for detailing all the functionalities a system must provide, as well as specifying the services and constraints associated with its operation. For [Pressman e Maxim 2021], the goal of RE is to provide all project stakeholders with a shared understanding of the problem. Thus, RE activities can be organized into six phases: Elicitation, Analysis, Specification, Prioritization, Validation, and Management, as outlined below:

- **Elicitation:** Involves gathering and identifying stakeholders' needs;
- **Analysis**: Focuses on analyzing and refining the elicited requirements, resolving potential conflicts, inconsistencies, or ambiguities;
- **Specification:** Requirements are formalized and documented clearly and precisely, using structured text, diagrams, or formal models;
- **Prioritization:** Ranks requirements based on their value, impact, and urgency, to guide implementation order and allocate resources to the most critical functionalities;
- **Validation:** Ensures that requirements are complete, consistent, and aligned with stakeholders' expectations; and
- **Management:** Involves tracking and controlling requirements throughout the software lifecycle.

2.2. Generative Artificial Intelligences

Large Language Models (LLMs) are advanced Natural Language Processing models capable of understanding and generating text in a manner similar to human language [Min et al. 2023]. These models enable the creation and refinement of various artifacts in software development, such as code in multiple programming languages and diagrams, based on the context provided by the user. To achieve this, the user interacts with the LLM through specific queries by embedding the question's context into prompts. The Generative AI then provides an output in natural language [Reynolds e McDonell 2021], facilitating the analysis and interpretation of the results.

With recent technological advances, LLMs have emerged as support tools for addressing the challenges posed by Requirements Engineering, optimizing processes that were once highly manual and error-prone. Illustrating this potential, the study by [Luitel et al. 2023] applied a model based on Bidirectional Encoder Representations from Transformers (BERT) to expand specific terms in a dataset. The results demonstrate that LLMs can contribute to the semantic expansion of terms, reducing ambiguities and increasing the completeness and consistency of requirements specifications.

3. Methodology

This study is characterized by a qualitative and exploratory approach, aiming to understand software engineers' perceptions regarding the impact of GenAI tools in supporting tasks related to FRs and NFRs in Requirements Engineering. The qualitative approach enables the investigation of subjective experiences and interpretations, allowing for an in-depth exploration of the nuances in participants' perceptions [Denzin e Lincoln 2011]. The exploratory nature of the study is justified by the emerging nature of the topic, making it possible to identify practices, challenges, and opportunities associated with the use of LLM-based tools.

The study begins with problem definition (Stage 1), identifying gaps in the use of GenAI in RE, with a focus on analyzing these tools' support for FRs and NFRs. To guide the investigation, in Stage 2, three central research questions were formulated:

- **(RQ1)** How do Software Engineering professionals evaluate the impact of generative AI in supporting functional and non-functional requirements across the different phases of Requirements Engineering?
- **(RQ2)** Which types of functional and non-functional requirements are perceived by professionals to benefit the most from the use of AI assistants?
- **(RQ3)** What challenges and improvement opportunities are identified in the use of generative AI tools to support Requirements Engineering processes?

These questions guide the investigation into the application of generative AI in handling FRs and NFRs throughout the various RE phases (elicitation, analysis, specification, prioritization, validation, and management) where the tools are perceived as most useful (RQ1). Additionally, the research explores aspects related to user requirement generation, interfaces, and quality attributes such as security, performance, usability, and other essential properties for software success (RQ2) [Malkawi 2013]. Finally, it seeks to identify challenges and improvement opportunities in the use of such tools to maximize their contribution to essential RE processes (RQ3).

In Stage 3, related to data collection planning, semi-structured interviews were conducted. The recruitment process involved sending approximately 30 invitations through posts and direct messages on LinkedIn, as well as by email, resulting in five effective participants, corresponding to a response rate of approximately 16%. The non-participation of the other invitees was mainly attributed to unavailability due to time constraints and professional demands. The interviews addressed a variety of topics aligned with the study's scope, including: the most commonly used generative AI tools; perceived applicability in each RE phase; types of requirements most benefited; recommended prompts for improved accuracy; challenges encountered; and suggestions for improvement. The questionnaire is available at `https://forms.gle/hBDKfAFuajvR9iVX6`.

The interviews were conducted following a structured protocol (Stage 4), with an estimated duration of 30 to 40 minutes. Sessions were held remotely using the Google Meet platform. With the participants' consent, interviews were recorded for documentation and later analysis, ensuring data anonymization and privacy protection.

Subsequently, in Stage 5, data analysis was performed through an interpretative approach, considering the research questions and the study's objectives. Initially, the participants' responses were organized according to the main topics addressed during the interviews, such as applicability in the ER phases, support for RFs and RNFs, challenges, and opportunities for improvement. Subsequently, convergent perceptions and recurring themes in the participants' accounts were identified and interpreted in light of the study's objectives and the adopted theoretical framework.

Finally, in Stage 6, the presentation of results synthesized the main findings concerning the relevance of generative AI tools in RE activities.

4. Results

4.1. Participant Profiles

Table 1 presents the profiles of the five professionals interviewed, highlighting that three of them are at the senior level and have eight years of experience in software development. To ensure anonymity, the identifier "P" (for Professional) is used to reference each interviewee. Four participants work as Requirements Analysts, while one holds the position of Product Owner with prior experience as a Requirements Analyst.

Table 1. Interviewee Profile.

#	Gender	Age	Experience	Level	AI Tool	Duration
P1	F	35	8 years	Senior	ChatGPT	33:45
P2	M	21	1 year	Junior	ChatGPT	16:46
P3	M	26	2 years	Junior	ChatGPT, Claude.ai	37:57
P4	F	33	8 years	Senior	ChatGPT/GPT4All	39:02
P5	M	29	8 years	Senior	ChatGPT	25:43

Although the sample size is small, the exploratory nature of the study aims to identify trends and generate initial insights, without requiring a large number of participants, as noted by [Rego et al. 2018]. It is also worth noting that the presence of senior professionals enriches the discussions, contributing years of experience in overcoming various challenges in Requirements Engineering.

Table 1 also shows that the average age of the participants is 28.8 years, and the average professional experience is 5.4 years, suggesting a blend of perspectives from both junior and senior professionals. Interview durations ranged from 16 minutes and 46 seconds to 39 minutes and 2 seconds, with an average of 30 minutes and 39 seconds, falling within the expected timeframe. Regarding the use of generative AIs, 100% of the interviewees confirmed their use in activities related to Requirements Engineering. Among the available options, ChatGPT is used by all participants. Additionally, one participant reported using Claude.ai. In terms of frequency, 40% (2 professionals) use AI daily for RE-related tasks, while the remaining 60% (3 professionals) use it on a weekly, monthly, or occasional basis.

4.2. RQ1. How do Software Engineering professionals evaluate the impact of generative AI in supporting functional and non-functional requirements across the different phases of Requirements Engineering?

Most interviewees (80%) considered generative AI most useful during the specification phase. Elicitation and validation were mentioned by 40% of the participants. Only one respondent (20%) cited the analysis phase, while prioritization and management were not mentioned by any of them. The perception that specification is the phase of Requirements Engineering most benefited by Generative AI can be explained by its predominantly textual and structured nature, which favors the generation, refinement, and organization of content by LLMs. In contrast, the limited mention of the analysis phase (20%) and the absence of prioritization and management suggest that these processes require more critical and contextual judgment, which may limit AI applicability.

The positive impact of generative AI on Requirements Engineering is evidenced by the fact that four interviewees rated its influence as '*Positive*' on a scale ranging from '*Very Negative*', '*Negative*', '*Neutral*', '*Positive*' and '*Very Positive*', while only one participant rated it as 'Neutral'. The use of AI was highlighted, for instance, for its ability to accelerate work during the requirements elicitation phase. However, professionals also emphasized that these tools still require improvements, as noted by P2: "*AI has helped me a lot in boosting productivity, but it needs improvements, such as greater accuracy in understanding the information*". P5 reinforces this view by stating: "*Generative AIs have undeniable potential, but there is still much to be done in terms of output quality and, above all, in terms of the sensitivity/dependence of these tools on the input prompts*".

Another point emphasized by professionals is the AI's ability to provide rapid responses, enabling requirements engineers to analyze a greater number of scenarios. P2 noted that their main activities with AI include elicitation, specification, and the creation of test scenarios in the validation phase. P2 also reported a specific use case in which they used ChatGPT to support the elicitation phase:"*ChatGPT is a useful tool for the requirements elicitation phase, allowing documents to be uploaded and requirements to be extracted for the system efficiently, contributing significantly to this activity*". P4 used AI to optimize requirement specification and summarize documents during a project: "*During the planning phase of a project involving system development, I used ChatGPT to summarize the submitted documents and later to review the specified requirements*".

On the other hand, some limitations were noted. P5 pointed out that, in the elicitation/discovery phase for complex systems, AI may not fully grasp the project's needs, resulting in insufficient or biased suggestions: "*In complex systems, such as distributed systems, AI may not fully understand the needs of a large-scale project, even if the prompt includes detailed scenarios and business rules. Using AI may become unfeasible, as a major constraint is how much it can suggest in response to the demand, which may result in outputs that do not match the actual needs. This leads to the generation of insufficient or biased data*". P2 reinforced this concern by stating that tools should be integrated with RE artifacts to provide more complete and project-grounded responses: "*I believe that integration with artifacts collected during*

Requirements Engineering would give AI a better understanding, enabling it to deliver more complete and effective responses".

4.3. RQ2. Which types of functional and non-functional requirements are perceived by professionals to benefit the most from the use of AI assistants?

The interviewees indicated that FRs are the most benefited by the use of generative AI, particularly due to its strong application in elicitation, specification, and validation tasks. AI was used to map functional requirements, usability flows, and database structures by P5 in a system under development. They reported this practical case using Claude.ai: "*Using the Claude Sonnet tool, we performed a mapping of a general view of functional requirements for a system under development. In this scenario, we used the tool to define the interface scope, button actions, usability flow, and database structure*".

Figure 1 shows the FRs most supported by GenAI.

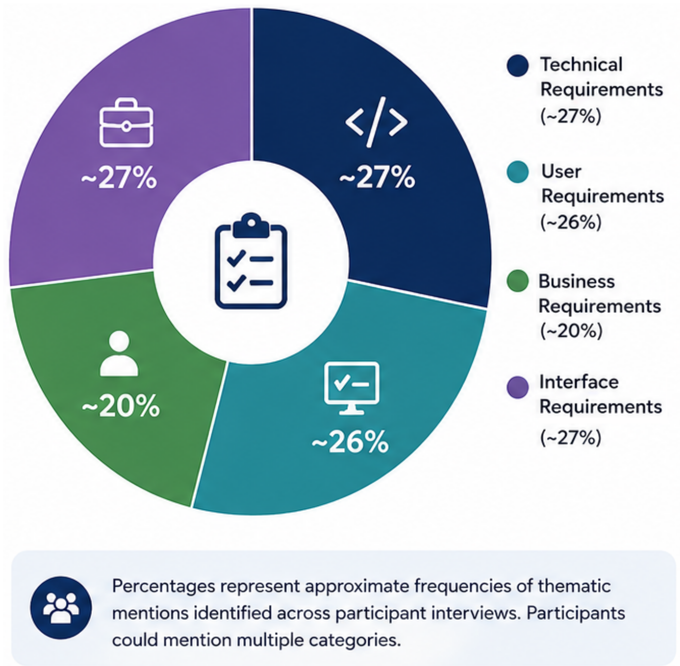

Figure 1. Perceived Support of Generative AI for FRs.

The predominance of FRs as the most benefited may be linked to the fact that engineers who already have a well-defined vision of the project tend to achieve better results when crafting more specific prompts. The data reveal that GenAI assist in addressing all types of functional requirements, with particular emphasis on Technical, User, and Interface Requirements, which represent approximately 27%, 26%, and 27%, respectively. The predominant recommendation highlights ChatGPT, used in 100% of the cases, as the preferred tool for supporting FRs, complemented by Claude.ai. The choice of ChatGPT reflects its ability to aid in defining technical requirements that meet client needs, as well as in synthesizing documents, reviewing specifications, and defining interface functionalities, aligning with the importance of technical, interface, and user requirements.

As for NFRs, AIs face more significant challenges, according to the participants' perceptions, as shown in Figure 2.

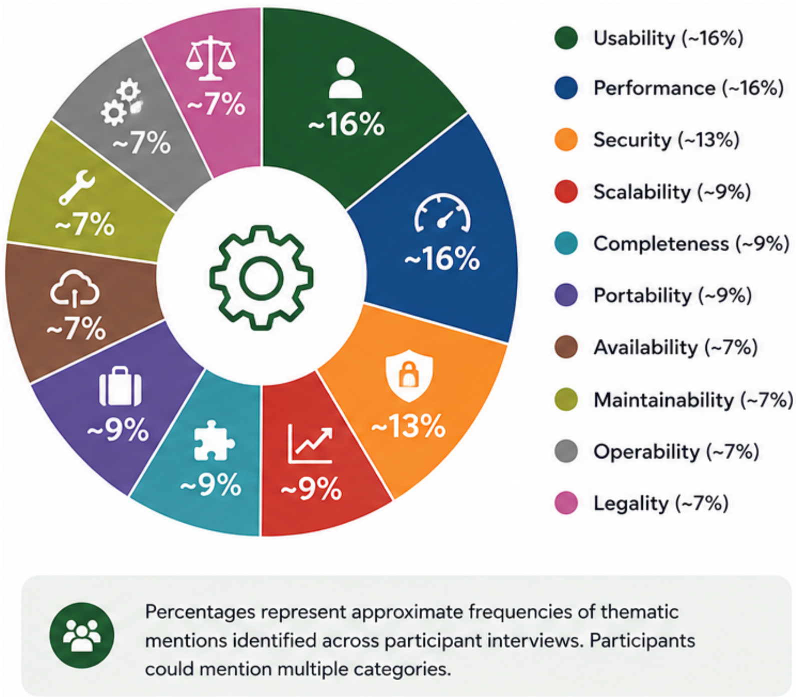

Figure 2. Perceived Support of Generative AI for NFRs.

The NFRs most positively impacted by generative AI were Performance and Usability (both with 16%). Security (13%) also stood out, reinforcing the concern for data protection. Meanwhile, Completeness, Portability, and Scalability (each with 9%) were cited as relevant for ensuring coverage, adaptability to different environments, and the ability to handle increasing demands. Maintainability, Operability, Compliance, and Availability requirements (7%) received less emphasis, despite being essential for the reliability and sustainability of the software.

As previously mentioned, limitations were emphasized regarding the ability of GenAI to assist with more complex NFRs, such as security, scalability, and availability. P5 suggested that AI should include reflective questioning before providing answers in order to avoid overly generic responses: "*AI could implement questions, for example, if the user says: 'I want to create a monolith', the AI should ask in what context that statement applies, returning a question such as: 'What is the context for this monolith creation?'*"".

A relevant point observed during the interviews is that professionals tend to assign less importance to non-functional requirements compared to functional ones. Reflecting this, P4 mentioned using generative AIs solely for performance and security requirements. This indication of a potential neglect of NFRs in software projects is a concern already noted in the specialized literature. [Ramos et al. 2019] and [Oliveira et al. 2024] emphasize that the lack of consideration for NFRs during the requirements analysis phase can lead to significant project failures.

4.4. RQ3. What challenges and improvement opportunities are identified in the use of generative AI tools to support Requirements Engineering processes?

The main challenges identified include:

- **Generic responses and the need for manual adjustments:** P1 noted that AI often generates shallow responses, requiring extensive rework: "*It provides generic information, which increases analysis time and, in some cases, demands complete revisions*". P2 reinforced that AI does not always interpret problems accurately, necessitating constant refinements: "*It's usually necessary to adjust the responses several times to get an acceptable result*".
- **Limitations in specialized domains:** P4, who works in the banking credit domain, pointed out the AI's difficulty in handling specialized knowledge: "*The responses are average because the system lacks deep knowledge on specific topics*".
- **Data security and integrity:** P3 raised concerns about privacy and legal compliance when using AI: "*Any AI tool must be approved by management before use, also considering information integrity*".

Despite these challenges, the professionals identified several opportunities for improvement:

- **Response precision and adequacy:** Improvements in technical writing, template generation, and support for requirements traceability (P1).
- **Integration with RE artifacts:** Linking to documents and requirements tools for greater contextual understanding (P2).
- **Information security:** Implementation of secure corporate sessions and greater control over data usage (P3).
- **User interaction optimization:** Providing examples of effective prompts (zero-shot, one-shot, and few-shot) to facilitate the generation of accurate responses (P4).
- **Adaptation to specific project contexts:** Integration with project management tools, wireframe design environments, and interactive support for validating non-functional requirements such as availability and scalability (P5).

P5 also suggested an enhancement to AI interaction by proposing that the system ask questions before delivering recommendations on NFRs: "*AI could include questions to better understand the context before responding, reducing the need for domain-specific expert systems for each area of software development*".

5. Discussion

The statements provided by the interviewees demonstrate that Generative AI tools are already being incorporated into practical RE activities. One participant described the use of ChatGPT to summarize project documentation and support requirements revision during planning activities, while another reported employing Claude.ai to assist in defining interface behavior, usability workflows, and database-related structures. These examples indicate that the integration of AI assistants with software artifacts and engineering support environments may improve the efficiency of RE activities and reduce the effort required to formulate highly detailed prompts.

The findings also indicate distinct levels of effectiveness between FRs and NFRs. According to the participants' perceptions, generative AI systems currently provide stronger support for FR-related activities, especially in tasks involving textual generation and organization. In contrast, the treatment of NFRs still presents considerable limitations, particularly when contextual adaptation and project-specific constraints are required. Requirements associated with security, scalability, maintainability, and availability involve factors that extend beyond textual interpretation, including architectural decisions, infrastructure characteristics, organizational restrictions, and specialized technical expertise accumulated throughout software development. Consequently, AI-generated suggestions for NFRs are frequently perceived as generic or insufficiently aligned with the realities of complex systems.

Although the participants recognized important productivity gains associated with GenAI adoption, they also highlighted several operational and technical challenges. One of the most recurrent concerns involves the dependence on carefully elaborated prompts to obtain satisfactory responses. Professionals reported the need for multiple refinements before achieving outputs considered useful, which may introduce additional rework into the RE process. Furthermore, limitations become more evident in specialized domains that involve sensitive information, strict regulations, or highly specific business rules, where AI systems still demonstrate restricted contextual understanding.

To mitigate these limitations, participants suggested improvements focused on increasing contextual awareness and interaction quality. Among the proposed enhancements are the inclusion of prompt guidance examples, tighter integration with project management and RE support tools, and mechanisms capable of generating more context-sensitive outputs. Another relevant suggestion concerns the incorporation of reflective interactions in AI systems, enabling them to ask clarifying questions before producing recommendations or requirements suggestions.

This perspective is aligned with the study conducted by [Hasso et al. 2024], which explored the customization of prompts according to project context. In that approach, prompt structures combined fixed instructions related to general guidance with variable sections containing project-specific information, such as operational scenarios and requirements descriptions. The results demonstrated that contextualized prompting strategies can improve the AI's understanding of software projects and contribute to more precise and relevant outputs during RE activities.

6. Limitations

The main limitation of this study is the small number of participants, with only five interviewees, a consequence of the difficulty in recruiting Software/Requirements Engineers, despite widespread outreach. Although the exploratory nature of the research justifies a smaller sample size to identify trends and initial hypotheses, this limitation may reduce the diversity of perspectives and fail to capture all challenges and practices related to the application of generative AI.

Despite the small sample size, recurring patterns and convergent perceptions emerged throughout the interviews, especially regarding the benefits, limitations, and challenges of using GenAI in RE activities. The presence of three senior professionals adds robustness to the findings, as they tend to identify more complex challenges, while junior professionals focus on immediate applicability. This thematic convergence, consistent with the criterion of theoretical saturation widely adopted in exploratory qualitative research, provided sufficient interpretive support for the analysis and discussion of the results presented in this study.

Furthermore, the analysis based on subjective testimonials may introduce individual biases related to each interviewee's experience and context, potentially impacting the validity of the insights. As such, the results should be interpreted as indicative and preliminary. Future studies with larger samples and mixed methodological approaches could validate and consolidate the identified trends, thereby reducing threats to validity and strengthening the robustness of the conclusions.

7. Conclusions and Future Work

The findings of this study suggest that Generative AI tools, particularly ChatGPT, are increasingly contributing to RE activities, especially in tasks associated with FRs. According to the interviewed professionals, these technologies are mainly beneficial during elicitation and specification activities, where they help accelerate documentation processes and reduce repetitive manual effort. The frequent adoption of such tools in practical scenarios reinforces their perceived usefulness in improving productivity and supporting everyday RE activities, although their contribution to phases such as prioritization and requirements management remains limited.

At the same time, the results reveal important limitations that still affect the effective use of GenAI in RE contexts. Participants emphasized challenges related to the dependence on highly detailed prompts and the generation of responses that are often too generic for complex software projects. These issues indicate the necessity of improving contextual understanding, response precision, and integration between AI systems and software engineering support environments. In this sense, more intuitive interaction mechanisms and tighter integration with RE artifacts may help professionals obtain outputs that are more consistent with project-specific needs.

Although this investigation was conducted with a limited number of participants, the study provides initial evidence regarding the practical adoption of Generative AI in Requirements Engineering and highlights relevant directions for future research. Subsequent studies should involve broader and more diverse participant groups, explore stronger integration between AI assistants and RE tools, and investigate strategies for improving prompt engineering practices in software engineering environments.

Future work may also examine how contextual adaptation mechanisms, interactive questioning strategies, and domain-aware AI support can improve the treatment of NFRs. Advancing these aspects may enable GenAI systems to assume a more strategic role within RE processes, contributing not only to task automation but also to improvements in software quality, consistency, and alignment with stakeholder expectations.

Acknowledgments

We thank OpenIA for supporting the Generative AI GPT-4o (Plus version) in grammatically reviewing the final text of this paper. The authors critically reviewed all AI-generated content and assume full responsibility for the final version of the paper, including its accuracy, originality, and the conclusions presented.

References

Denzin, N. K. e Lincoln, Y. S. (2011). *The Sage Handbook of Qualitative Research*. Sage.

Felfernig, A., Stettinger, M., Atas, M., Samer, R., Nerlich, J., Scholz, S., e Raatikainen, M. (2018). Towards utility-based prioritization of requirements in open source environments. In *2018 IEEE 26th International Requirements Engineering Conference (RE)*, pages 406–411. IEEE.

Hasso, H., Fischer-Starcke, B., e Geppert, H. (2024). Quest-re question generation and exploration strategy for requirements engineering. In *2024 IEEE 32nd International Requirements Engineering Conference Workshops (REW)*, pages 1–9. IEEE.

Luitel, D., Hassani, S., e Sabetzadeh, M. (2023). Using language models for enhancing the completeness of natural-language requirements. In *International Workshop on Requirements Engineering Foundation for Software Quality (REFSQ)*, pages 87–104.

Mahbub, T., Dghaym, D., Shankarnarayanan, A., Syed, T., Shapsough, S., e Zualkernan, I. (2024). Can gpt-4 aid in detecting ambiguities, inconsistencies, and incompleteness in requirements analysis? a comprehensive case study. *IEEE Access*, 12:171972–171992.

Malkawi (2013). The art of software systems development: Reliability, availability, maintainability, performance (ramp). *Human-Centric Computing and Information Sciences*, 3:1–17.

Min, B., Ross, H., Sulem, E., Veyseh, A. P. B., Nguyen, T. H., Sainz, O., Agirre, E., Heintz, I., e Roth, D. (2023). Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Computing Surveys (CSUR)*, 56(2).

Oliveira, A., Correia, J., Assunção, W. K. G., Pereira, J. A., de Mello, R., Coutinho, D., Barbosa, C., Libório, P., e Garcia, A. (2024). Understanding developers' discussions and perceptions on non-functional requirements: The case of the spring ecosystem. *Proceedings of the ACM on Software Engineering*, 1(FSE):1–22.

Parra, E., Dimou, C., Llorens, J., Moreno, V., e Fraga, A. (2015). A methodology for the classification of quality of requirements using machine learning techniques. *Information and Software Technology*, 67:180–195.

Pressman, R. S. e Maxim, B. R. (2021). *Software Engineering: A Practitioner's Approach*. McGraw Hill Brasil, 9th edition.

Ramos, F. B. A., Pedro, A., Cesar, M., Costa, A. A. M., Perkusich, M. B., de Almeida, H. O., e Perkusich, A. (2019). Evaluating software developers' acceptance of a tool for supporting agile non-functional requirement elicitation. In *Proceedings of the International Conference on Software Engineering and Knowledge Engineering (SEKE)*, pages 26–42. KSI Research.

Rego, A., Pina, M., e Jr., V. M. (2018). Quantos participantes são necessários para um estudo qualitativo? linhas práticas de orientação. *Revista de Gestão dos Países de Língua Portuguesa*, 17(2):43–57.

Ren, S., Nakagawa, H., e Tsuchiya, T. (2024). Combining prompts with examples to enhance llm-based requirement elicitation. In *48th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1376–1381. IEEE.

Reynolds, L. e McDonell, K. (2021). Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (CHI EA '21)*. Association for Computing Machinery.

Ronanki, K., Berger, C., e Horkoff, J. (2023). Investigating chatgpt's potential to assist in requirements elicitation processes. In *49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 354–361. IEEE.

Sommerville, I. (2018). *Software Engineering*. Pearson, 10th edition.

Vijayakumar e Nethravathi, P. S. (2021). Use of natural language processing in software requirements prioritization – a systematic literature review. *International Journal of Applied Engineering and Management Letters (IJAEML)*, 5(2):152–174.

Yu, X., Liu, L., Hu, X., Keung, J., Xia, X., e Lo, D. (2024). Practitioners' expectations on automated test generation. In *Proceedings of the 33rd ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA 2024)*, pages 1618–1630.

Zhao, L., Alhoshan, W., Ferrari, A., Letsholo, K. J., Ajagbe, M. A., Chioasca, E.-V., e Batista-Navarro, R. T. (2021). Natural language processing for requirements engineering: A systematic mapping study. *ACM Computing Surveys (CSUR)*, 54(3):1–41.