Synthetic user requirements: Sense making at early stages of product development

Valeria Resendez[1], Andrew Hornback[2], Harinishree Sathu[2], J. Ben Tamo[2], Yining Yuan[2], Nese Baz[1], Funda Yildirim[1], Russell Chan[1], May D. Wang[2], Maria Fernanda Cabrera[3], Simone Borsci[1]

[1]University of Twente, Enschede, Overijssel, Netherlands
[2]Georgia Institute of Technology, Atlanta, GA, USA
[3]Universidad Politécnica de Madrid, Madrid, Spain

ABSTRACT

Designing digital systems that meet stakeholder needs depends on the effectiveness of requirements elicitation, a process that is essential but often time-consuming and resource-intensive. To support early-stage requirements engineering, we tested whether adding external knowledge to large language models (LLMs) can emulate requirements generated by human experts. We tested three large language models (GPT-5.2-instant, Claude-Sonnet-4.5, and Gemini-3-Pro) under different conditions. Each model was given different levels of background information (i.e., knowledge) and asked to take on different expert roles. All model–knowledge–expert combinations were tested on two tasks (i.e., prompts), representing two levels of specificity. Each combination was repeated 20 times, producing a total of 960 requirement sets. The LLM-generated requirements were evaluated across two dimensions: output quantity and alignment with human expert requirements. Requirement quantity varied by model and contextual knowledge, while overlap with human expert requirements remained low. Our findings suggest that LLMs can serve primarily as generative scaffolding for human experts.

1 INTRODUCTION

The future of precision medicine is in the development of digital applications that enable access to standardized data [7, 22]. Such development requires mapping the interdisciplinary requirements of stakeholders to ensure the successful implementation of the technology [5, 6, 16]. Requirements guide the design of new technologies by making explicit what the system should do and the limitations under which it must function [25]. Determining requirements involves a set of activities including discovery, prioritization, negotiation, and collaboration with all relevant stakeholders [15, 29]. Yet, generating requirements is a challenging process, as it is often time-consuming and resource-intensive [13, 27]. These challenges have motivated research into tools and methods that can support parts of the requirements engineering process.

Large language models (LLMs) have emerged as promising tools. Recent research has demonstrated their potential for generating, categorizing, and assessing requirements, although their results are typically less varied and still demand human supervision and domain knowledge to adequately meet user needs [1, 2, 18]. Beyond text generation, LLMs have also been explored as emulators of human behavior. For instance, prior studies show that LLMs can simulate demographic and social characteristics or act as negotiation agents through predefined roles and interaction rules [12, 14]. However, such emulation often fails to capture the contextual richness and variability of real human behavior, and their performance across tasks remains uncertain [8, 14]. More broadly, these limitations reflect well-known challenges of LLMs, including hallucinations and methodological concerns related to training data, benchmarking practices, and replicability [8, 17]. We therefore ask, *to what extent can context-based LLMs emulate requirements generated by human experts?* This question is particularly relevant in the context of healthcare, where requirements are shaped by contextual information such as clinical and technical expertise, workflows, and regulations [28]. In such settings, eliciting stakeholder requirements depends on access to task-specific and contextual information that LLMs lack. Retrieval-augmented generation (RAG) addresses this gap by integrating information retrieval into the generation process: rather than relying solely on learned representations, RAG retrieves relevant knowledge from external sources and incorporates it at inference time [8]. In engineering requirements, potential stakeholder needs, constraints, and domain knowledge are essential elements to depict the context in which the product will be used. Such context is often gathered and recorded in heterogeneous documents (e.g., reports, grey and scientific literature, project or design plans, etc.). The RAG can therefore supply the context that would otherwise be unavailable to the LLM's model. However, its effectiveness depends on retrieval quality; for example, poorly indexed or irrelevant sources may introduce noise rather than clarity [11].

2 AIMS OF THIS WORK

We propose that LLMs can generate early-stage requirements from textual information, such as topic-specific literature or project proposals. Some requirements produced by LLMs may be partially or fully validated by human experts, potentially accelerating the process of requirements elicitation and negotiation. To our knowledge, no prior work has examined the use of LLMs, either with or without RAG, to generate project requirements. Here, we investigate whether LLMs can produce requirements comparable to those generated by humans, introducing a framework that combines RAG, role-based prompting, and a multi-model (i.e., LLM) comparison.

3 METHODOLOGY

We conducted a 3 × 2 × 4 simulation study to compare LLM-based requirements under different levels of contextual knowledge and expert-role assumptions. Across three LLMs (GPT-5.2-instant, Claude-Sonnet-4.5, and Gemini-3-Pro), we tested two knowledge conditions: (a) no external knowledge and (b) summary project-proposal knowledge. In addition, we evaluated the effect of instructing the LLM to assume three expert roles: no explicit expert, clinical expert, legal expert, and technical expert. All 24 model–knowledge–expert combinations were tested across two tasks defined in the prompts, each representing two levels of specificity (i.e., simple vs. detailed). This results in 48 conditions. Model–knowledge–expert combinations were set up in Poe AI and prompted through the API to generate requirements. All decoding parameters (e.g., sampling temperature and top-p) were kept at the default settings of the respective models. To account for variability in the generation of requirements, we iterated through each combination 20 times, resulting in 960 generated requirement sets. After completing the simulations, we aggregated the results across conditions and reported the mean and standard deviations.

3.1 Models

Three LLMs (i.e., GPT-5.2-instant, Claude-Sonnet-4.5, and Gemini-3-Pro) were selected to compare models that differ in their approaches to reasoning, tool use, and multimodal processing. Claude-Sonnet-4.5 emphasizes instruction following and context-heavy workflows; GPT-5.2-instant is a general-purpose model designed for complex and agentic tasks; and Gemini-3-Pro focuses on multimodal reasoning and large-scale input processing. This selection allowed comparative assessments across distinct modeling capabilities [4, 24].

3.2 Role-Based Experts

We generated roles using an automated pipeline, translating our set of real-world experts into a single group of experts. The pipeline consists of three stages. First, we explicitly defined an expert group as a set of real-world experts who share a common domain focus. Second, we retrieved all publicly available information about selected experts from the web, including professional biographies and relevant publications. Third, we aggregated this material and synthesize the evidence into a description that reflects the group's collective expertise, priorities, and predefined focus. Using this process, we constructed three expert roles: (1) Clinical experts, represented by pediatricians, who specialize in the medical care of infants, children, and adolescents [26]. (2) Technical experts, responsible for providing expertise on software design constraints, prototype development, and technical feasibility [23]. (3) Legal experts, who contribute to the development and oversight of governance frameworks that ensure the lawful and ethical use of sensitive data [21].

3.3 Tasks

Models were prompted to generate requirements with two levels of specificity. The first tasks explicitly asked models to generate functional and non-functional requirements for a federated infrastructure that enables the analysis and reuse of health and genomic data. In contrast, the second task requested a general list of requirements for a federated infrastructure for health and genomic data management, expanding the expected capabilities to include data discovery, access, analysis, and reuse, and highlighting additional system qualities such as technical robustness and innovation. While the first task provided more structure by describing the requirement categories, we considered the second as more detailed since it introduces a broader scope of system functions and attributes. All models were prompted using a standardized template maintained across conditions. Contextual knowledge was injected as a structured prefix. No minimum or maximum length requirement was specified; the model determined statement granularity autonomously. An example of the prompt is provided in the OSF link: https://osf.io/umeqv.

3.4 Evaluation against human-generated requirements

Outputs were compared to a human reference list of 366 requirements developed within a European research project over a period of 18 months. This list was derived from proposal reviews, literature analysis, and expert input (∼ 60 participants), collected through interviews, surveys, workshops, and focus groups.

3.5 Data Analysis

We evaluated the generated requirements across two aspects: (1) quantity of the requirements and (2) semantic alignment to human-generated requirements:

1. Frequency of requirements. This measure evaluated whether adding contextual knowledge and expert roles increased the number of generated requirements. Since the outcome variable is a frequency number and it varied across conditions, we used a negative binomial regression to model differences in requirement frequency. The binomial model included the LLM type (GPT-5.2-instant, Claude-Sonnet-4.5, Gemini-3-Pro), knowledge condition (no knowledge and proposal), and expert role (no expert, clinical, legal, and technical expert). Results are reported as Incidence Rate Ratios (IRRs), which indicate the relative change in the expected number of generated requirements compared to a reference condition [10]. An IRR greater than 1.0 reflects a higher expected requirement count, whereas an IRR below 1.0 reflects a lower expected count relative to the reference.

2. Alignment with human expert requirements. This measure assessed the extent to which LLM-generated requirements overlapped with requirements identified by human experts. For each generated requirement, we check whether it is semantically similar to any human requirement. A threshold of ≥ 0.75 was applied to classify a generated requirement as matching a human-generated one, consistent with the mid-to-upper range of cosine similarity values associated with high semantic overlap in sentence embedding models [3, 19, 20]. If any human requirement exceeded this threshold, the generated requirement was counted as a match. Based on these matches, we computed the proportion of generated requirements exceeding the cosine similarity threshold. Second, we computed the average cosine similarity between each generated requirement and its closest human counterpart, capturing the overall semantic proximity between generated and expert requirements

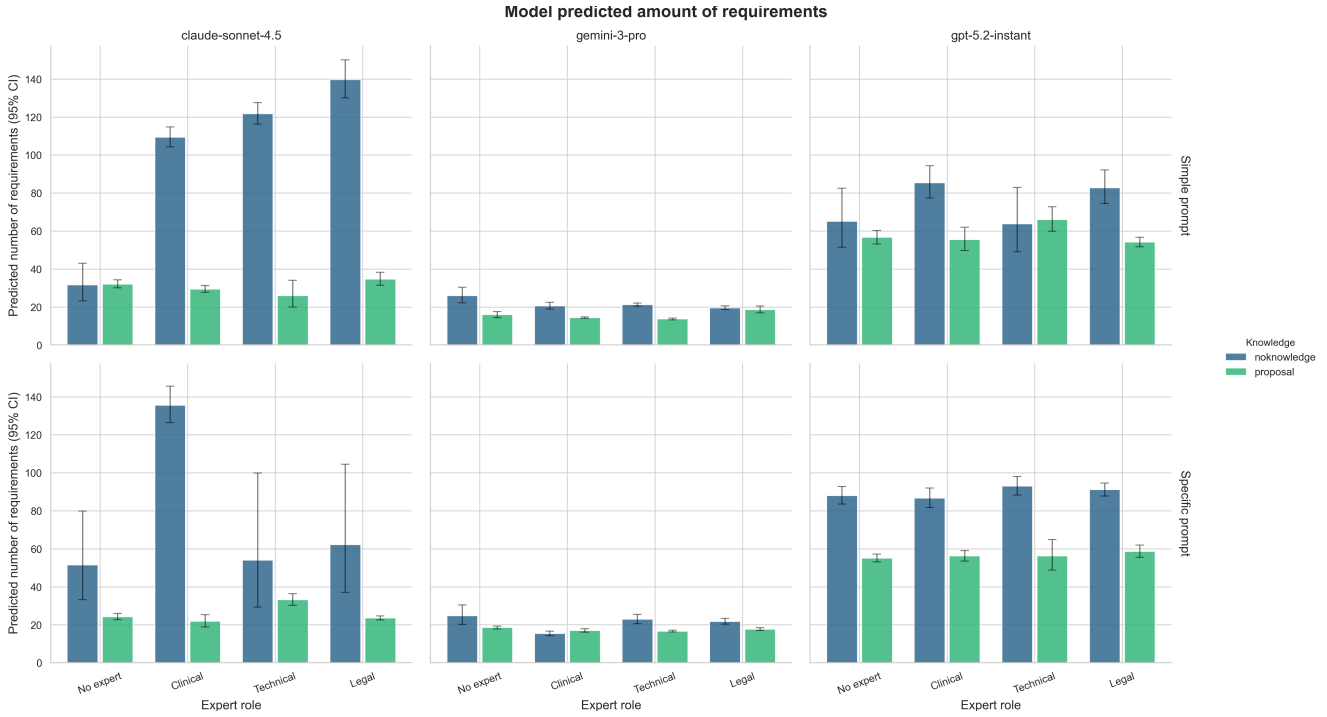

Figure 1: Estimated mean frequency of requirements per condition.

even when no direct match exists. Both metrics were calculated for each iteration and aggregated across experimental conditions (prompt × model × role × knowledge). Results are reported as the mean (M) and standard deviation (SD) across iterations.

4 RESULTS

4.1 Frequency of Requirements

To examine whether contextual knowledge and expert roles influenced the number of generated requirements, we fitted a negative binomial generalized linear model ($N = 960$). The model showed sufficient explanatory power (pseudo-$R^2 = 0.332$, log-likelihood = 4512.2), indicating that requirement frequency varied across conditions. To provide a conservative assessment, p-values were adjusted using a Bonferroni correction ($m = 19$ comparisons; $\alpha_{\text{adj}} = .0026$). Overall, the number of generated requirements differed across conditions. Including the proposal in the RAG system reduced the number of generated requirements significantly in comparison to the no-knowledge condition (IRR = 0.70, Bonferroni-adjusted $p < .001$). This result suggests that contextual knowledge could constrain the generation of requirements. Additionally, differences in LLM models were also observed. Particularly, Gemini-3-Pro generated fewer requirements than GPT-5.2-instant (IRR = 0.31, adjusted $p < .001$). In contrast, prompt specificity and other interaction effects (e.g., knowledge and export role) were not significant after the correction. Figure 1 summarizes the estimated marginal means from the negative binomial generalized linear model across all 48 experimental conditions. Details of the outputs and frequency analyses are available on OSF (https://osf.io/umeqv).

4.2 Average of generated requirements aligned with at least one human expert requirement

Overall, the overlap between synthetic and human-generated requirements was low across conditions when applying a cosine similarity threshold of .75. On average, models reproduced between 0 and 1.7% of the human requirements per condition. This result suggests that only a small proportion of synthetically generated requirements closely matched the human-generated requirements. The highest overlap with the human-expert requirement set was observed for Gemini-3-Pro under the clinical role with the proposal and the more specific prompt ($M = 2.49\%$, $SD = 2.83\%$) of generated requirements exceeding the cosine similarity threshold of .75; ($M_{cos} = 0.48$, $SD_{cos} = 0.02$). Other conditions showing a relatively similar overlap included Claude-Sonnet-4.5 under the legal role with proposal knowledge and the simple prompt ($M = 1.35\%$, $SD = 1.83\%$, cosine similarity $> .75$; $M_{cos} = 0.49$, $SD_{cos} = 0.02$), and Claude-Sonnet-4.5 under the technical role with proposal knowledge and the simple prompt ($M = 1.34\%$, $SD = 2.90\%$, cosine similarity $> .75$; $M_{cos} = 0.48$, $SD_{cos} = 0.02$).

Although exact matches were rare, generated requirements showed moderate semantic similarity to the human-generated requirements. Across conditions, the average cosine similarity of the synthetically generated requirements and their closest human counterpart ranged from approximately $M = 0.30$ to $M = 0.52$ (typically $SD \approx 0.01$–0.05 per condition), indicating that many generated requirements captured conceptually related ideas, although they did not correspond to the specific requirement.

5 DISCUSSION

This study evaluated the extent to which context-augmented LLMs can emulate requirements generated by human experts in a healthcare-genomics domain. The key finding is that increases in requirements were not aligned with more semantic alignment to human-generated requirements. In general, the no-knowledge condition generated more requirements than the proposal-based contextual knowledge condition. Across models, GPT-5.2-instant produced the highest number of requirements, followed by Claude-Sonnet-4.5, while Gemini-3-Pro generated fewer. However, the higher output did not correspond to more alignment with human-expert requirements. For practitioners, this suggests that LLMs without contextual knowledge can serve as broad-coverage tools, surfacing candidate requirements at negligible cost. However, engineers using LLMs for an initial set of requirements would need thoughtful filtering to manage the noise.

The injection of the proposal into the knowledge of the LLM (RAG) did not improve alignment with human-generated requirements. Although the generated requirements were fewer, the alignment to human-generated requirements was marginal. In this regard, adding contextual knowledge to LLMs may not generate outputs that are more closely related to those of human experts. This result could partly be due to the similarity metric, which, while capturing broad semantic overlap, is insensitive to the pragmatic and structural qualities that distinguish expert requirements from plausible alternatives. Moreover, human-expert requirements were generated through an 18-month iterative process involving workshops, surveys, and negotiation, a process that synthesizes tacit knowledge, stakeholder priorities, and considerations that no document fully captures. LLMs, even when augmented, lack access to this interpretive layer. These results align with prior work demonstrating that contextual guidance can constrain LLM outputs without necessarily improving the quality [17, 18]. The implication is that RAG, as currently implemented in this work, may function more as a focusing mechanism than as a quality-enhancement tool in requirements elicitation.

Assigning expert roles similarly constrained the requirement output volume without producing more aligned requirements. While some role-knowledge combinations achieved higher similarity scores (e.g., Gemini-3-Pro with a clinician role), no systematic advantage emerged for role-prompted conditions. This may reflect the limitations of persona-based prompting: although LLMs can adopt surface-level characteristics of a specified role, they cannot replicate the decision-making heuristics, professional priorities, or tacit knowledge that shape how real experts formulate requirements [9, 12, 26]. Taken together, rather than substituting for expert judgment, LLM-generated requirements could serve as conversational scaffolds, starting points that reduce the cognitive burden on stakeholders during early requirement elicitation. The value of LLM-generated requirements must be in priming discussions, surfacing overlooked considerations, or accelerating the identification of areas requiring deeper negotiation. The 18-month human process that produced the ground-truth requirements in this study represents a substantial investment. If LLMs can compress even the initial divergent-thinking phase, the efficiency gains may be considerable.

Alongside these findings, two limitations deserve attention. First, a low alignment rate does not necessarily indicate that non-matching requirements are incorrect or useless. Some LLM-generated requirements may represent valid system needs absent from the reference set, either because they were deprioritized during expert workshops or because they reflect considerations that emerge more readily from automated analysis than from participatory elicitation. A qualitative evaluation of non-aligned requirements would therefore help distinguish genuine misalignments from potentially valuable additions. Second, the use of POE AI as the access to the models introduces a further constraint worth acknowledging: provider-specific filters may have varied in ways that were not controlled. Addressing this through direct API access under the same conditions could help further determine whether the alignment patterns observed here hold across deployment environments.

Nonetheless, what the present findings make clear is that no current model-knowledge-expert combination produces requirements that substitute for human experts. The relevant question is therefore not which combination performs better in isolation, but which most effectively reduces the effort of the human experts who will ultimately own the requirements.

6 CONCLUSION

LLMs can support early-stage requirements elicitation, but their contributions should be interpreted with care. Adding contextual knowledge (e.g., through RAG) may constrain outputs without consistently improving alignment with expert-defined requirements. Similarly, role prompting appears to have a limited impact, and generating a larger number of requirements does not necessarily translate into higher quality. Rather than replacing human expertise, the practical value of LLM-generated requirements lies in reducing the initial effort needed to start participatory processes. In this sense, LLMs can provide a low-cost starting point that helps structure early discussions among stakeholders. Integrating LLMs into requirements engineering, therefore, requires recalibrating expectations about their role. A potential workflow could be structured as follows. First, generate a broad and unconstrained set of candidate requirements without using RAG. This initial set can serve to stimulate discussion and reveal blind spots during early stakeholder engagement. Next, introduce RAG using relevant project documentation, domain literature, and regulatory frameworks to produce a more focused and context-aware set of requirements. Finally, involve an additional group of human experts to review both sets and determine which requirements should be carried forward into negotiation and prioritization. In this approach, RAG-based and non-RAG-based generation are not competing alternatives but complementary strategies. In both cases, human-led elicitation remains essential to ensure that the resulting requirements reflect user needs.

7 ACKNOWLEDGMENTS

This project received funding from the European Union's Horizon Europe research and innovation program under grant agreement No. 101137423. We thank the members of the PROTECT-CHILD for their contributions and collaboration.

REFERENCES

[1] Waad Alhoshan, Alessio Ferrari, and Liping Zhao. 2025. How effective are generative large language models in performing requirements classification? (2025).

[2] Chetan Arora, John Grundy, and Mohamed Abdelrazek. 2023. Advancing requirements engineering through generative AI: Assessing the role of LLMs. (2023).

[3] Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic textual Similarity - Multilingual and cross-lingual focused evaluation.

[4] Mathieu Chartier, Nabil Dakkoune, Guillaume Bourgeois, and Stéphane Jean. 2025. HiBenchLLM: Historical Inquiry Benchmarking for Large Language Models. *Data & Knowledge Engineering* 156 (March 2025), 102383.

[5] Elizabeth Cutting, Meghan Banchero, Amber L. Beitelshees, et al. 2016. User-centered design of multi-gene sequencing panel reports for clinicians. *Journal of Biomedical Informatics* 63 (October 2016), 1–10.

[6] Keith Danahey, Brittany A. Borden, Brian Furner, et al. 2017. Simplifying the use of pharmacogenomics in clinical practice: Building the genomic prescribing system. *Journal of Biomedical Informatics* 75 (November 2017), 110–121.

[7] Diptavo Dutta and Nilanjan Chatterjee. 2025. Expanding scope of genetic studies in the era of biobanks. *Human Molecular Genetics* (May 2025), ddaf054.

[8] Mohamed Abo El-Enen, Sally Saad, and Taymoor Nazmy. 2025. A survey on retrieval-augmentation generation (RAG) models for healthcare applications. *Neural Computing and Applications* 37, 33 (November 2025), 28191–28267.

[9] Rubén Fuentes-Fernández, Jorge J. Gómez-Sanz, and Juan Pavón. 2010. Understanding the human context in requirements elicitation. *Requirements Engineering* 15, 3 (September 2010), 267–283.

[10] Joseph M. Hilbe. 2011. *Negative Binomial Regression* (2 ed.). Cambridge University Press, Cambridge, UK.

[11] Andrew Hornback, Harinishree Sathu, Kyungbeom Kim, Yifei Wang, Yuanda Zhu, Monica Isgut, Pavithra Avula, Asma Khimani, and May D. Wang. 2026. Large language models in healthcare and biomedical informatics: A comprehensive review. *Innovation and Emerging Technologies* 13 (2026), 2630001.

[12] Tiancheng Hu and Nigel Collier. 2024. Quantifying the persona effect in LLM simulations. (2024).

[13] Devi Karolita, John Grundy, Tanjila Kanij, Humphrey Obie, and Jennifer McIntosh. 2024. CRAFTER: A persona generation tool for requirements engineering. In *Proceedings of the 19th International Conference on Evaluation of Novel Approaches to Software Engineering*. SCITEPRESS, Angers, France, 674–683.

[14] Ahmad Mouri Zadeh Khaki, Ahyoung Choi, and Laleh Seyyed-Kalantari. 2025. Simulating social behavior of LLM-based autonomous negotiator agents. *International Journal of Human–Computer Interaction* 41, 23 (December 2025), 15169–15178.

[15] Edward Meinert, Madison Milne-Ives, Svitlana Surodina, and Ching Lam. 2020. Agile requirements engineering and software planning for a digital health platform. *JMIR Public Health and Surveillance* 6, 2 (May 2020), e19297.

[16] Jens Meyer, Stefan Ostrzinski, Daniel Fredrich, et al. 2012. Efficient data management in a large-scale epidemiology research project. *Computer Methods and Programs in Biomedicine* 107, 3 (September 2012), 425–435.

[17] Johannes J. Norheim, Eric Rebentisch, Dekai Xiao, et al. 2024. Challenges in applying large language models to requirements engineering tasks. *Design Science* 10 (2024), e16.

[18] Giovanni Quattrocchi, Liliana Pasquale, Paola Spoletini, and Luciano Baresi. 2025. Can LLMs generate user stories and assess their quality? (2025).

[19] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese BERT-networks. (2019).

[20] Navid Rekabsaz, Bhaskar Mitra, Mihai Lupu, and Allan Hanbury. 2017. Toward incorporation of relevant documents in word2vec. *arXiv preprint arXiv:1707.06598* (2017).

[21] Nola M. Ries. 2021. Conceptualizing Interprofessional Working – When a Lawyer Joins the Healthcare Mix. *Journal of Interprofessional Care* 35, 6 (Nov. 2021), 953–962.

[22] Silvia Rodríguez-Mejías, Sara Degli-Esposti, Sara González-García, and Carlos Luis Parra-Calderón. 2024. Toward the European Health Data Space. *Journal of Biomedical Informatics* 156 (August 2024), 104670.

[23] H. Saiedian and R. Dale. 2000. Requirements engineering: making the connection between the software developer and customer. *Information and Software Technology* 42, 6 (April 2000), 419–428.

[24] Andrei Sobo, Awes Mubarak, Almas Baimagambetov, and Nikolaos Polatidis. 2025. Evaluating LLMs for Code Generation in HRI. *Applied Artificial Intelligence* 39, 1 (December 2025), 2439610.

[25] Ian Sommerville. 2016. *Software Engineering* (10 ed.). Pearson Education, Boston, MA, USA.

[26] Julie Uchitel, Errol Alden, Zulfiqar A. Bhutta, et al. 2022. Role of pediatricians, pediatric associations, and academic departments in ensuring optimal early childhood development globally. *Journal of Developmental and Behavioral Pediatrics* 43, 8 (October 2022), e546–e558.

[27] K. Joeri Van Der Velde, Gurnoor Singh, Rajaram Kaliyaperumal, et al. 2022. FAIR Genomes metadata schema promoting Next Generation Sequencing data reuse. *Scientific Data* 9, 1 (April 2022), 169.

[28] Lex Van Velsen, Jobke Wentzel, and Julia EWC Van Gemert-Pijnen. 2013. Designing eHealth that matters via a multidisciplinary requirements development approach. *JMIR Research Protocols* 2, 1 (June 2013), e21.

[29] Didar Zowghi and Chad Coulin. 2005. Requirements elicitation: A survey of techniques, approaches, and tools. In *Engineering and Managing Software Requirements*, Aybüke Aurum and Claes Wohlin (Eds.). Springer, Berlin/Heidelberg, 19–46.