Informing AI Prompt Engineering for Industrial Engineering Training: A Human Factors Engineering Course Approach

Mr. EDWARD JAMES ISOGHIE, University of Louisville

Edward Isoghie is a PhD candidate with a research focus on human factors and engineering education leveraging emerging technologies such as AI, digital twin, and virtual reality. He obtained his bachelor's and master's degrees in Industrial and Production Engineering from the University of Ibadan, Nigeria, and a masters in Operations Management at the University of Alabama, Tuscaloosa.

Jason J Saleem, University of Louisville

Jason J. Saleem is an Associate Professor with the Department of Industrial Engineering at the J.B. Speed School of Engineering at the University of Louisville. He is also a Co-Director of the Center for Human Systems Engineering (CHSE). Dr. Saleem received his Ph.D. from the Department of Industrial and Systems Engineering at Virginia Tech in 2003, specializing in human factors engineering and ergonomics. Dr. Saleem's research interests focus on the integration of human factors engineering with the development of health information technology (HIT). His research also focuses on provider-patient interaction with respect to exam room computing, as well as virtual care tools and applications. Dr. Saleem also maintains an engineering education research portfolio and in 2024 was awarded a grant by the National Science Foundation (NSF) entitled, 'Introducing a Mixed-Methods Approach to Engineering Students through Human-Centered Design'.

Informing AI Prompt Engineering for Industrial Engineering Training: A Human Factors Engineering Course Approach

Abstract

The integration of artificial intelligence (AI) tools into engineering education presents new opportunities to enhance problem-solving, design reasoning, and reflective learning. This study explores the role of AI prompt engineering as a pedagogical tool within Industrial Engineering education. This research aims to design and investigate how AI structured prompts can guide students through user-centered design processes and usability evaluations and critical analysis in an Industrial Engineering course.

Students enrolled in this course participated in two instructional phases: a traditional usability evaluation exercise and an AI-assisted module employing a structured prompt. The prompt was designed to target specific usability engineering skills such as heuristic evaluation, task analysis, and interface redesign. Learning outcomes are assessed through evaluations of usability reports.

The findings demonstrate that AI-guided prompts can supplement human effort to identify and articulate design issues and promote reflective thinking about user experience. The study contributes to the growing discourse on human-AI collaboration in engineering education by offering a replicable model for integrating prompt engineering into learning environments. Overall, this work provides insights into designing effective AI-mediated learning strategies in human factors education and advances the future of Industrial Engineering education in the era of digital transformation.

Introduction

The advent of generative artificial intelligence (GenAI) is transforming industries and engineering education, reshaping teaching and learning, workflows and redefining the skills required for success in the modern workforce. Among these emerging competencies, prompt engineering has emerged as a critical skill for efficient communication with AI systems[1], [2]. It is imperative for industrial engineering students to translate domain-specific knowledge into effective design and use prompts for process design and improvement. In Industrial Engineering (IE), usability evaluation is a core method of human factors application. Prompt engineering may allow students to harness AI effectively as a partner for ideation, analysis, and evaluation. Usability evaluation is done to ensure that systems and processes are optimized for efficiency, safety, and user satisfaction. Traditionally, this evaluation has relied on manual methods, requiring significant time and expertise to observe complex interactions, recognize subtle usability problems, articulate underlying causes, and prioritize issues based on severity and impact. Novice evaluators often struggle with these demands, tending to focus on superficial interface features or to apply heuristics mechanically with limited understanding. Furthermore, trained evaluators find analysis of usability data to be time-consuming and cognitively demanding. However, the integration of AI tools, such as large language models (LLMs), offers unprecedented opportunities to streamline these evaluations [3]. By leveraging prompt engineering, students may design targeted queries that guide AI systems to analyze usability metrics, simulate scenarios, and generate actionable insights. This not only accelerates the evaluation process during usability analysis but also equips students with the skills to collaborate effectively with AI.

Emerging studies focusing on user experience (UX) have utilized AI for usability evaluation in various forms. LLMs generate follow-up questions during unmoderated usability testing, eliciting richer explanations from participants and helping evaluators probe user reasoning more deeply. Additionally, LLM has been used to synthesize verbal, behavioral, and design data into traceable findings, reducing time-to-insight while preserving links to underlying evidence [4], [5] [6]. Despite these advances, existing research leaves several critical questions unanswered from an industrial engineering education perspective. Most studies implemented AI-assisted usability tools to carry out usability evaluation. There is limited research on how AI could support students to develop stronger usability reasoning skills, improving their ability to justify and prioritize issues. These gaps highlight the need for study that explicitly situates AI-assisted usability evaluation within an educational framework. The purpose of this study is to address this gap by examining the use of AI to support usability evaluation for students in industrial engineering. Grounded in prior work on AI-assisted UX analysis and broader research on AI in engineering education, this study conceptualizes AI as an instructional scaffold. This study aims to frames AI-supported usability evaluation relevant to engineering education, investigates how AI assistance influences students' ability to identify, justify, and prioritize usability issues, and derives pedagogical implications for integrating AI into Industrial Engineering curricula to support learning.

Literature Review

Prompt engineering is increasingly recognized as a foundational and transformative skill for interacting with LLMs such as ChatGPT [1],[7]. It is defined as the art of designing effective inputs to guide AI systems in producing relevant, high-quality outputs. The practice has gained prominence due to the rapid adoption of generative AI tools in industry. Having prompt patterns offers reusable, composable strategies for controlling output, improving reasoning quality, managing interaction flow, and mitigating errors. Previous works have highlighted the importance of structured methodologies for designing, testing, and refining prompts to overcome issues such as prompt brittleness and ambiguity [3],[8]. This enhances effective human-AI interactions, output quality and advance prompt engineering as a transferable skill. Qian [9] categorized prompt engineering into technique-based and process-based approaches, with the former targeting specific learning goals and the latter supporting cognitive engagement and collaborative thinking. Several advanced techniques such as chain-of-thought prompting, self-refinement prompting, decomposed prompting and role prompting have been shown to improve reasoning and problem-solving capabilities and effective methods for improving the accuracy and relevance of AI-generated responses [8],[10]. These methods encourage step-by-step reasoning, iterative improvement, and modular task decomposition, enabling AI systems to produce more accurate and interpretable outputs. Despite the effectiveness of these prompting strategies, the strategies fall short of producing accurate AI output and increase user cognitive load, which has led to experimentation with hybrid prompting that combines two or more prompt types. This limitation has led to several prompt design frameworks. Lo [11] provided criteria for crafting effective prompts, emphasizing qualities and proposes the CLEAR framework (Concise, Logical, Explicit, Adaptive, and Reflective) as a structured method to improve the quality, relevance, and evaluability of AI-generated content. This work emphasizes not only grammatical clarity but also reflective iteration, enabling learners to assess AI outputs, recognize limitations, and refine prompts systematically. Building on this foundation, Park and Choo [12] extended prompt engineering into applied pedagogical practice by translating abstract principles into educator-centered strategies. Their IDEA framework integrates CLEAR with actionable components PARTS (Persona, Aim, Recipients, Theme, Structure) and REFINE (Rephrase key words, Experiment with context and examples, Feedback loop, Inquiry questions, Navigate by iterations, and Evaluate and verify outputs) to support iterative design, contextualization, and ethical application of generative AI in classrooms. Unlike more technically oriented frameworks, their approach foregrounds instructional alignment, learner diversity, and educator accountability. Furthermore, Ratnayake and Wang [13] introduce the PERFECT framework, a modular prompting structure designed to enhance LLM performance across NLP tasks through explicit purpose definition, role specification, formatting, constraints, and temporal context. Empirical evaluations demonstrated that structured prompts improve accuracy and reasoning depth. The potential of prompt engineering in engineering education, particularly in industrial engineering, is barely explored. This study examines the use of AI prompt engineering for usability evaluation in industrial engineering education.

Prompt Engineering in Usability Studies

According to Schulhoff et al. [14], 58 text-based prompt engineering techniques have been shown to enhance the performance of generative AI models across diverse applications. Several studies have employed various prompting strategies with large language models (LLMs) to address challenges in usability evaluation to identify usability. Shin et. al [15] explored multiple strategies, including Role Prompting, Chain of Thought (CoT) Prompting, Self-Refining Prompting, Least-to-Most Prompting, and hybrid approaches combining Role Prompting and CoT Prompting in multimodal UX evaluation. Role Prompting involved assigning specific personas to LLMs, such as user personas or domain experts, to enhance contextual analysis, which helped identify usability issues like confusing terminology and navigation errors. CoT Prompting guided LLMs through step-by-step reasoning, enabling systematic identification of complex, multi-step usability issues. Self-Refining Prompting allowed iterative refinement of responses, improving the depth of analysis, while Least-to-Most Prompting broke down complex problems into smaller sub-problems, enhancing logical coherence and accuracy. The hybrid approach, combining Role Prompting and CoT Prompting, proved most effective, identifying the highest number of UX issues with the highest severity scores. Likewise, Alsaleh et al. [16] used Zero-Shot Prompting and Fine-Tuning to analyze negative app reviews and generate usability recommendations. Zero-Shot Prompting relied on the LLM's pre-trained knowledge to perform tasks without examples, identifying basic issues but often producing general responses. Fine-Tuning involved retraining LLMs on annotated datasets to improve domain-specific accuracy, with fine-tuned GPT-4 achieving the highest BLEU, ROUGE, and METEOR scores, outperforming Gemini and BLOOM in generating actionable recommendations. Zhong et al. [17] shows that instruction-based, reasoning-oriented prompts allow LLMs to simulate expert heuristic evaluation from screenshots. The study shown that AI-driven evaluations identify a substantial proportion of usability issues often exceeding individual human evaluators. Across all studies, the authors demonstrated the transformative potential of LLMs, particularly GPT-4, in improving usability evaluations. However, challenges such as hyper-accuracy distortions, difficulty in understanding app conventions, and limited ability to aggregate information across multiple screens were noted. These findings emphasize the importance of human oversight and further optimization of LLMs to address their limitations and ensure optimal results. Prompt engineering has been applied in some areas such as medical education [18], data analysis and programming skill acquisition [19] and flight training [20] .Structured prompts aligned with Bloom's Taxonomy and Depth of Knowledge (DOK) have been used to guide students through tasks requiring analysis, evaluation, and creation [21]. The integration of AI and prompt engineering into education faces several challenges. Elwakil and Emami [22] highlight the varying levels of AI literacy among students, which necessitates tailored instruction and foundational scaffolding. Therefore, the application of prompt engineering in industrial engineering education has become imperative to translate this domain-specific knowledge into effective prompts for tasks such as usability evaluation.

Methodology

This study used a qualitative, inspection-based methodology to compare a traditional heuristic evaluation with an AI-assisted heuristic evaluation across two mobile applications (a housing app and a fitness app) as shown in Figure 1 and Figure 2 to students in an Industrial Engineering course. The evaluation relied exclusively on provided static screenshots and predefined task scenarios rather than interaction with live systems. For the housing app, seven (7) screenshots covered scenarios such as searching for rental listings, viewing recently sold properties, and checking houses for sale; for the fitness app, six (6) screenshots depicting scenarios such as planning training, starting activities (e.g., walking, running, mountain biking), and tracking progress.

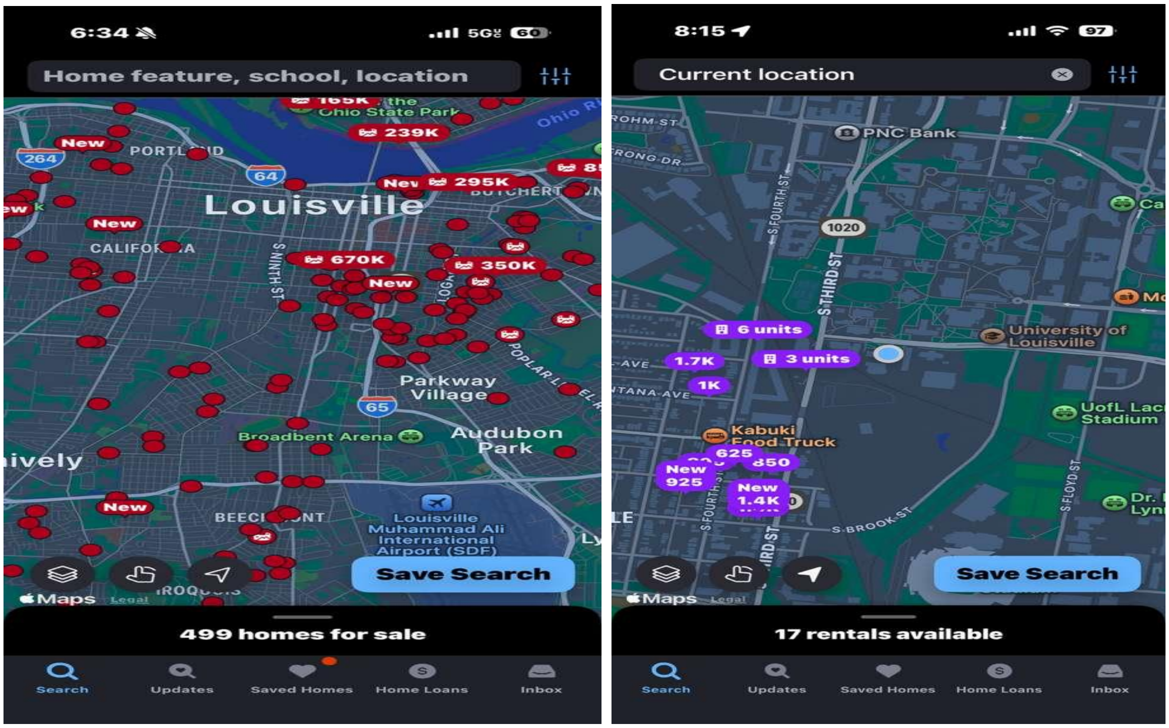

Figure 1: Examples of two screens in the housing app

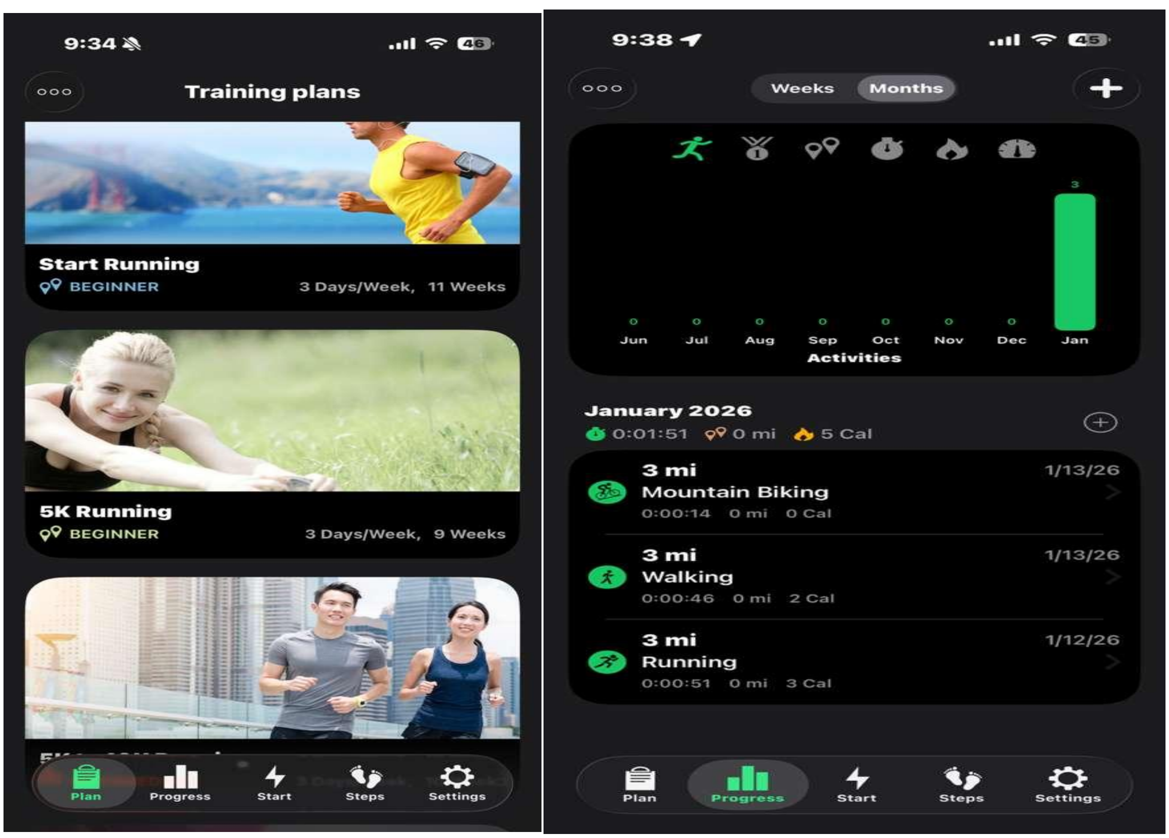

Figure 2: Examples of two screens in the fitness app

Nielsen's ten usability heuristics [23] served as the analytic framework, and the interfaces were examined along the implied task flow to ensure that issues were identified not only within single screens but also across transitions. The study applied a within-subject design with two conditions: (1) heuristic evaluation conducted independently without AI assistance, and (2) an AI-supported heuristic evaluation using a structured prompt to generate usability findings. In both conditions, each heuristic was systematically assessed for each application, with the students documenting observed violations. All identified issues were recorded with a screenshot reference, a brief explanation linking the issue to the relevant heuristic, a severity rating on a 0-4 scale (0= no usability problem at all, 1 = cosmetic problem, 2 = minor usability problem, 3 = major usability problem, to 4 = usability catastrophe), and a short rationale grounded in anticipated user impact within the task context. Because the analysis used static screenshots, limited assumptions about system behavior (e.g., feedback timing, navigation depth, missing states) were made only when necessary and were explicitly stated. Furthermore, six (6) Nielsen's heuristic violations were expected to be identified in both conditions. The analysis of the usability evaluation results focused on comparing heuristic violations identification, severity assessment, and evaluative consistency across the two conditions: heuristic evaluation without AI assistance and with AI assistance. Findings were analyzed at three levels: (1) coverage of Nielsen's heuristics violated (2) convergence and divergence across without and with AI assistance and (3) severity distribution.

Prompt Design

The prompt was deliberately designed using the PERFECT framework [13] to ensure it produces clear, systematic, and high-quality usability evaluation output. First, the Purpose is stated explicitly: the evaluator must conduct a heuristic evaluation, identify violated heuristics, assign severity ratings, and produce design recommendations, which removes ambiguity and defines the expected outcomes. Next, the Elements are specified to guide what the response must include: use of Nielsen's 10 heuristics, selection of six violated heuristics, screenshot-based locations, problem descriptions, severity ratings, and severity justifications. This ensures the analysis is complete but still focused. The Role is defined by instructing the respondent to act "as a usability expert," which sets expectations for professional judgment and appropriate terminology. The Format is constrained by requiring a table for violations and severity ratings, followed by a summarized justification and design recommendations, ensuring the output is structured and comparable across students. While explicit Examples are not necessary for this task, the prompt requires concrete evidence by referencing specific screenshots and transitions, which function as the operational equivalent of examples in artifact-based usability inspection. The Conditions are made explicit by limiting the evaluation to the provided screenshots and task description and requiring interaction-level analysis across screens, preventing unsupported assumptions and encouraging task-flow reasoning. Finally, the Timeframe is embedded in the sequential nature of the screenshots and the instruction to consider transitions, capturing the temporal dimension of user interaction during task completion rather than historical or predictive time. These PERFECT-aligned design choices make the prompt rigorous, reproducible, and well-suited for systematic comparison of usability findings. The designed prompt is shown below:

As a Usability expert perform a heuristic evaluation using all 10 of Nielsen's usability heuristics using only the provided screenshots and task description. Identify six heuristics violated, specify place of occurrence by referencing screenshots, and give problem description. Also, assign a severity rating from 0 (not a usability problem) to 4 (usability catastrophe), and justify each severity rating in a table. Lastly, give summarized justification and design recommendations. Your evaluation should consider the interaction across screens, not individual screens in isolation.

Data Analysis

Data analysis combined qualitative and descriptive analytical procedures to examine differences between heuristic evaluation without AI assistance and AI-assisted heuristic evaluations. Two data sources were analyzed: (1) usability evaluation outputs produced by participants (identified issues, heuristic violated, and severity ratings), and (2) participants' justification describing their evaluation experience under each condition. Usability evaluation outputs were analyzed descriptively to assess heuristic violated and severity ratings across participants. For each application and condition, identified usability issues were grouped by Nielsen's heuristics and aggregated across the participants. The frequency with which specific heuristics were violated was calculated to identify dominant usability problem areas. Severity ratings were analyzed by computing distributions across the 0-4 scale, enabling comparison of how often minor versus major problems were identified under each condition. Finally, the summarized justification and design recommendations of the usability reports were analyzed qualitatively.

Results and Discussion

This study examined the use of structured AI prompts on students' usability evaluation and critical analysis in an Industrial Engineering course. Eighteen students participated in two instructional phases: a traditional usability evaluation activity conducted without AI support and an AI-assisted activity guided by a structured prompt. The human-only evaluation identified 154 usability issues (78 in the housing app and 76 in the fitness app), whereas the AI-assisted evaluation identified 132 issues (66 in the housing app and 66 in the fitness app). The analysis further indicates that, without AI assistance, evaluators more frequently mapped multiple usability issues to a single heuristic violation. Specifically, six heuristics were associated with multiple issues in the human-only condition, a pattern that was not observed in the AI-assisted condition.

The analysis presented in Table 1 focuses on eleven submissions which consented to the study to provide comparison of heuristic identification patterns across the two conditions (without AI vs.

with AI) for the Housing and Fitness apps. For each heuristic, the numerator shows how many times that heuristic was identified as violated under a given condition, while the denominator is the total number of times that heuristic was identified across both conditions; the corresponding percentages therefore indicate the share of identifications attributable to each condition, and difference in percentage between without and with using AI to show convergence and divergence across both conditions. Also, the distribution of heuristic violations reveals differences between evaluations conducted without AI assistance and those conducted with AI support across both the housing and fitness applications. Overall, several heuristics show relatively balanced identification rates between conditions. For instance, *Visibility of system status* and *User control and freedom* display near parity across both apps, with marginal percentage differences. Similarly, *Match between system and the real world* and *Consistency and standards* exhibit moderate variation but no consistent directional shift across both applications. In contrast, clear differences emerge in specific heuristics. For the housing app, the largest reductions in violations identified with AI (relative to without AI) occur in: Flexibility and efficiency of use (90% without AI vs. 10% with AI; 80% difference), Help users recognize, diagnose, and recover from errors (100% vs. 0%; 100% difference), Error prevention (78% vs. 22%; 56% difference), Help and documentation (75% vs. 25%; 50% difference). A similar pattern appears in the fitness app, particularly for: Flexibility and efficiency of use (83% vs. 17%; 67% difference), Help and documentation (83% vs. 17%; 67% difference), Aesthetic and minimalist design (62% vs. 38%; 23% difference). Conversely, AI-assisted evaluations identified relatively more violations in Recognition rather than recall for both apps (Housing: 38% vs. 63%; Fitness: 40% vs. 60%).

Based on the results, substantial divergence is observed context-dependent heuristics, particularly those involving efficiency, flexibility, documentation, and error recovery. In these categories, human-only evaluations consistently identified more violations than AI-assisted evaluations. This may indicate that nuanced workflow inefficiencies, recovery pathways, and documentation gaps require contextual reasoning that is less effectively surfaced through AI-supported processes. In contrast, AI-assisted evaluations showed a relative increase in violations associated with Recognition rather than recall. This pattern suggests that AI may systematically flag violations that are more structurally identifiable through rule-based reasoning. Furthermore, figure 3 and 4 shows the severity distribution on how usability issues were classified across evaluation conditions for both app. For the **housing app**, without AI assistance, violations were predominantly rated at lower and moderate severity levels. Specifically, Severity 1 (0.80) and Severity 2 (0.58) show higher proportions compared to the AI-assisted condition (0.20 and 0.42, respectively). In contrast, with AI assistance, a higher proportion of issues were classified as Severity 3 (0.56 vs. 0.44 without AI). Notably, Severity 4 issues were identified only in the without-AI condition (1.00) and were absent in the AI-assisted evaluation (0.00). For the **fitness app**, a similar pattern emerges. Without AI, Severity 1 (0.82) and Severity 2 (0.59) dominate relative to the AI condition (0.18 and 0.41, respectively). However, AI-assisted evaluation shows a higher proportion of Severity 3 issues (0.66 vs. 0.34 without AI) and a slightly lower but still substantial proportion of Severity 4 issues (0.43 vs. 0.57 without AI). Across both applications, the general trend suggests that AI-assisted evaluation reduces the relative proportion of low-severity (Severity 1) classifications while increasing the proportion of mid-level (Severity 3) severity ratings. Overall, human-only evaluations tend to classify a greater proportion of issues at lower severity levels (Severity 1 and 2), whereas AI-assisted evaluations shift a larger share of issues into moderate severity (Severity 3). The absence of Severity 4 issues in the housing app under AI-assisted evaluation is particularly notable. This may indicate a conservative bias in AI-supported classification for critical usability issues. In contrast, for the fitness app, Severity 4 issues remain present under both conditions, suggesting that context or interface characteristics may influence how critical usability issues are detected.

Table 1: Heuristic Violated for both apps (Housing and Fitness)

Heuristic Violated	App	Without AI	With AI	% Difference (without AI -with AI)
Visibility of system status	Housing	48% (10/21)	52%(11/21)	-5%
	Fitness	48% (10/21)	52%(11/21)	-5%
Match between system and the real world	Housing	55%(11/20)	45%(9/20)	10%
	Fitness	50%(10/20)	50%(10/20)	0%
User control and freedom	Housing	47%(9/19)	53%(10/19)	-5%
	Fitness	53%(10/19)	47%(9/19)	5%
Consistency and standards	Housing	48%(11/23)	52%(12/23)	-4%
	Fitness	54%(13/24)	46%(11/24)	8%
Error prevention	Housing	78%(7/9)	22%(2/9)	56%
	Fitness	50%(7/14)	50%(7/14)	0%
Recognition rather than recall	Housing	38% (6/16)	63% (10/16)	-25%
	Fitness	40%(6/15)	60%(9/15)	-20%
Flexibility and efficiency of use	Housing	90% (9/10)	10% (1/10)	80%
	Fitness	83%(5/6)	17%(1/6)	67%
Aesthetic and minimalist design	Housing	47%(8/17)	53%(9/17)	-6%
	Fitness	62%(8/13)	38%(5/13)	23%
Help users recognize, diagnose, and recover from errors	Housing	100% (1/1)	0	100%
	Fitness	50%(2/4)	50%(2/4)	0%
Help and documentation	Housing	75% (6/8)	25% (2/8)	50%
	Fitness	83%(5/6)	17% (1/6)	67%

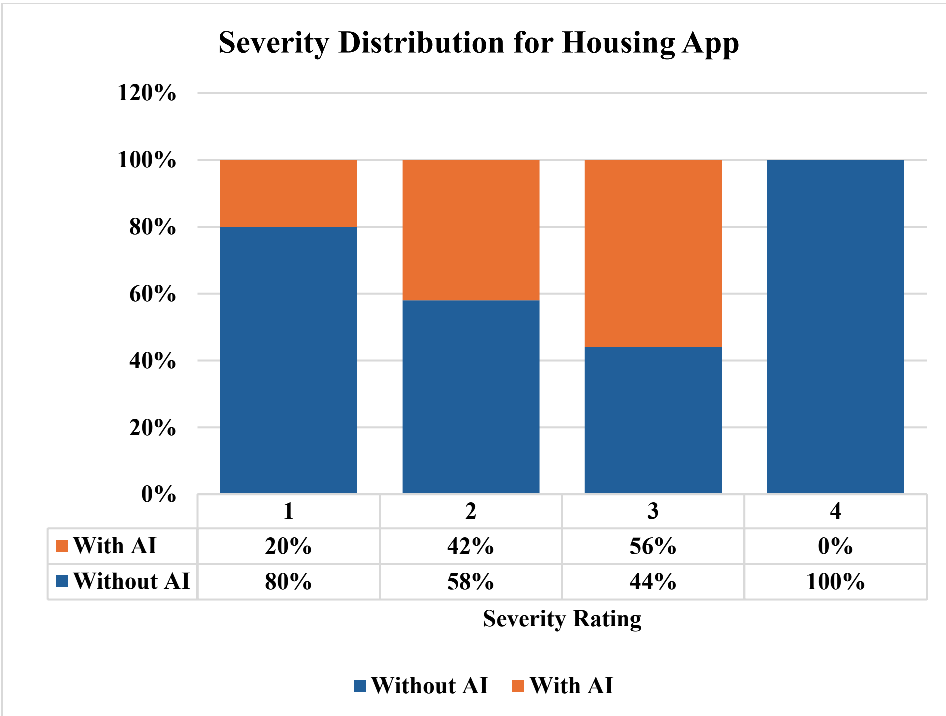

Figure 3: Severity rating distribution for housing app

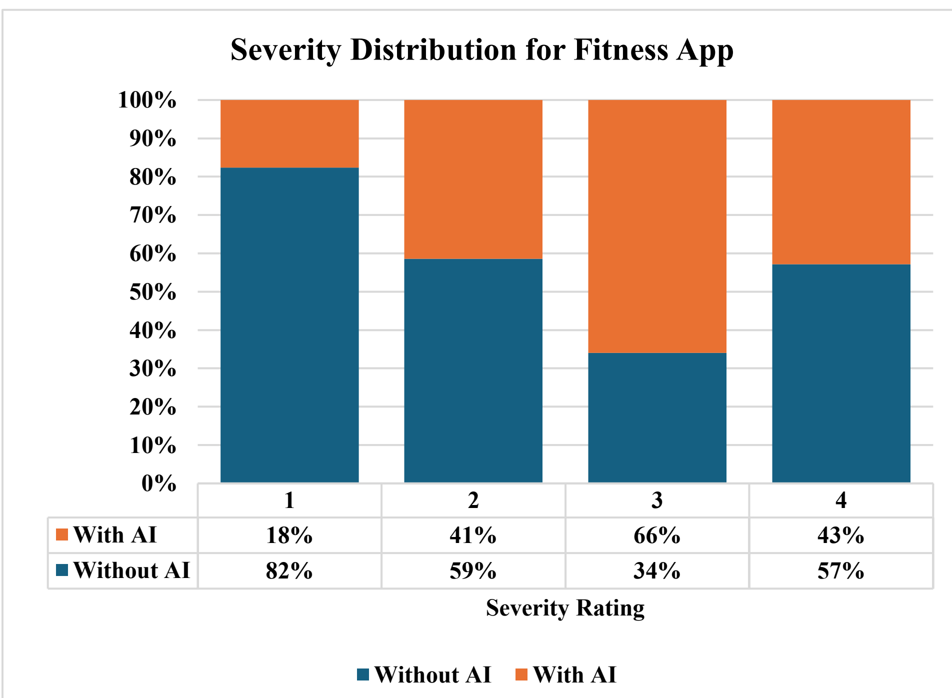

Figure 4: Severity rating distribution for fitness app

Qualitative Result

To compliment the quantitative results, we qualitatively analyzed the eleven (11) submissions for each application under the two conditions focusing on the summary and justification, and design recommendations in the usability evaluation report. The analysis focused exclusively on the textual explanations where participants described usability problems, justified heuristic violations, and proposed design improvements. First, the relevant text was reviewed line-by-line and analyzed to identify codes. Second, conceptually similar codes were consolidated into broader categories to reduce redundancy and increase analytic clarity. Third, codes were elevated into themes when they appeared across multiple participants and represented a recurring usability construct. Theme refinement was performed iteratively by re-reviewing all submissions to ensure each theme had internal coherence and captured patterns present across the dataset.

For the Housing app, data were coded using seven predefined themes derived from the finalized Housing codebook: visual clutter, status visibility, visual inconsistency, user control limitations, recall burden, terminology mismatch, and missing help/documentation. For the Fitness app, data were coded using seven corresponding themes from the Fitness codebook: data inconsistency, missing tracking feedback, ambiguous control/error prevention issues, navigation lockout, recall burden, layout density, and terminology ambiguity. Using the codebook, across participants, AI-based evaluations largely converged with manual findings for both apps. This study demonstrates how students can leverage AI prompt engineering to become engineers who can design safer, more intuitive, and more resilient technologies, skills that are increasingly essential in a world where complex socio-technical systems and AI-enabled tools are ubiquitous.

Limitations

This study was conducted without time constraints and with a small sample size. Future work should increase the number of participants and incorporate time constraints to better reflect realistic evaluation conditions. In addition, the prompt limited the AI to identifying violations of only six heuristics, which may have influenced the findings. Subsequent studies should avoid restricting the number of heuristics the AI can detect to enable a more comprehensive assessment of usability issues.

Conclusion

This paper examines the use of structured AI prompt engineering as an instructional approach for usability evaluation education. Finding across both applications reveals that AI assisted evaluation shifted toward specific heuristic categories and shifted severity classifications toward moderate levels, while human-only evaluations more frequently identified context-sensitive issues related to flexibility, documentation, error recovery, and efficiency. Reliance solely on AI-assisted heuristic evaluation may bias the distribution of identified usability issues. Hybrid approaches that combine human expertise with AI support may therefore provide a more balanced and comprehensive assessment. This study has direct implications for engineering education, particularly in usability engineering, human-centered design, and human-AI collaboration. First, the findings demonstrate that AI-assisted heuristic evaluation can expand the breadth of problem identification, especially for less visually salient issues such as feedback visibility, documentation gaps, cognitive load, and error prevention. Also, by utilizing prompts as pedagogical scaffolds, the key contribution of this study is the use of prompt engineering as a viable and effective tool for industrial engineering educators. Finally, this study contributes to broader discussions about responsible AI integration in engineering education. Rather than replacing student judgment, AI acted as a structured support tool that enhanced heuristic consistency. This supports a model of human-AI collaboration in education, where AI augments analytical rigor while students retain interpretive authority.

References

[1] D. Federiakin, D. Molerov, O. Zlatkin-Troitschanskaia, and A. Maur, "Prompt engineering as a new 21st century skill," *Front. Educ. (Lausanne).*, vol. 9, 2024, doi: 10.3389/feduc.2024.1366434.

[2] M. Elwakil and T. Emami, "Teaching Prompt Engineering Across Disciplines: Challenges, Outcomes, and Best Practices," 2025.

[3] M. Z. Naser, "A Review of Prompt Engineering Techniques for Large Language Models," *Int. J. Hum. Comput. Interact.*, pp. 1–35, Dec. 2025, doi: 10.1080/10447318.2025.2607553.

[4] E. Kuric, P. Demcak, and M. Krajcovic, "Unmoderated Usability Studies Evolved: Can GPT Ask Useful Follow-up Questions?," *Int. J. Hum. Comput. Interact.*, vol. 41, no. 15, pp. 9752–9769, 2025, doi: 10.1080/10447318.2024.2427978.

[5] K. Turbeville *et al.*, "LLM-powered Multimodal Insight Summarization for UX Testing," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Nov. 2024, pp. 4–11. doi: 10.1145/3678957.3685701.

[6] E. J. Isoghie, "The Use of Generative AI for the Rapid Development of Qualitative Interview Transcripts for a Human-Centered Design Problem."

[7] A. Vu and J. Oppenlaender, "Prompt Engineer: Analyzing Skill Requirements in the AI Job Market," May 2025, [Online]. Available: http://arxiv.org/abs/2506.00058

[8] J. White *et al.*, "A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT," Feb. 2023, [Online]. Available: http://arxiv.org/abs/2302.11382

[9] Y. Qian, "Prompt Engineering in Education: A Systematic Review of Approaches and Educational Applications," Dec. 01, 2025, *SAGE Publications Inc.* doi: 10.1177/07356331251365189.

[10] J. Wei *et al.*, “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models,” Jan. 2023, [Online]. Available: http://arxiv.org/abs/2201.11903

[11] L. S. Lo, “The CLEAR path: A framework for enhancing information literacy through prompt engineering,” ., vol. 49, no. 4, Jul. 2023, doi: 10.1016/j.acalib.2023.102720.

[12] J. Park and S. Choo, “Generative AI Prompt Engineering for Educators: Practical Strategies,” *Journal of Special Education Technology*, Sep. 2024, doi: 10.1177/01626434241298954.

[13] H. Ratnayake and C. Wang, “A Prompting Framework to Enhance Language Model Output,” in *AI 2023: Advances in Artificial Intelligence*, T. Liu, G. Webb, L. Yue, and D. Wang, Eds., Singapore: Springer Nature Singapore, 2024, pp. 66–81.

[14] S. Schulhoff *et al.*, “The Prompt Report: A Systematic Survey of Prompt Engineering Techniques,” Feb. 2025, [Online]. Available: http://arxiv.org/abs/2406.06608

[15] S. Shin, J. Oh, and S. Lee, “Can LLMs See What I See? A Study on Five Prompt Engineering Techniques for Evaluating UX on a Shopping Site,” in *Conference on Human Factors in Computing Systems - Proceedings* , Association for Computing Machinery, Apr. 2025. doi: 10.1145/3706599.3720079.

[16] N. Alsaleh, R. Alnanih, and N. Alowidi, “Enhancing Software Usability Through LLMs: A Prompting and Fine-Tuning Framework for Analyzing Negative User Feedback,” *Computers*, vol. 14, no. 9, Sep. 2025, doi: 10.3390/computers14090363.

[17] R. Zhong, D. W. Mcdonald, and G. Hsieh, “Synthetic Heuristic Evaluation: A Comparison between AI- and Human-Powered Usability Evaluation,” 2025. doi: XXXXXXX.XXXXXXX.

[18] T. Heston and C. Khun, “Prompt Engineering in Medical Education,” *International Medical Education*, vol. 2, no. 3, pp. 198–205, Aug. 2023, doi: 10.3390/ime2030019.

[19] A. Garg, K. Nisumba Soodhani, and R. Rajendran, “Enhancing data analysis and programming skills through structured prompt training: The impact of generative AI in engineering education,” *Computers and Education: Artificial Intelligence*, vol. 8, Jun. 2025, doi: 10.1016/j.caeai.2025.100380.

[20] F. Duruaku, N. A. Sonnenfeld, B. Nguyen, and F. G. Jentsch, “Informing AI Prompt Engineering for Flightcrew Training Design Through Human Factors Research,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 69, no. 1, pp. 1223–1229, Sep. 2025, doi: 10.1177/10711813251369798.

[21] S. L. Spessard, “The Prompt as Pedagogy: Activating Bloom’s and Advancing Depth of Knowledge Through AI Prompting,” 2025.

[22] M. Elwakil and T. Emami, “Teaching Prompt Engineering Across Disciplines: Challenges, Outcomes, and Best Practices,” 2025.

[23] J. Nielsen and R. Molich, “Heuristic evaluation of user interfaces,” in *Conference on Human Factors in Computing Systems - Proceedings*, Association for Computing Machinery, Mar. 1990, pp. 249–256. doi: 10.1145/97243.97281.

Appendix

Usability Heuristic Evaluation Assignment
Assignment Type: Individual
Overview

In this assignment, you will conduct a heuristic usability evaluation of two mobile applications:

1. A housing application
2. A fitness application

You will complete the evaluation, using two different approaches:

- Task 1: Heuristic evaluation without using AI prompt
- Task 2: Heuristic evaluation with structured AI prompt

You will use Nielsen’s 10 usability heuristics as the evaluation framework. The goal of this assignment is to develop your ability to systematically inspect user interfaces, identify usability problems, assess their severity, and propose human-centered design improvements.

This is an individual assignment. Collaboration with other students is not permitted.

Evaluation Material

You are provided with the following materials on Blackboard:

- Screenshots of a housing app:
- Screenshots of a fitness app
- Task scenarios for each app
  - Tasks for the housing app include searching for houses available for rent, viewing recently sold properties, and checking houses listed for sale.
  - Tasks for the fitness app: plan training, start training such as walking, running, mountain biking, and track progress.

Important:
You are not required to explore the live applications. Your analysis should be based only on the provided screenshots and task descriptions.

Task

Using the provided screenshots and task scenarios, you will independently conduct a usability evaluation of both applications.

For each app, you are expected to:

- Identify six (6) heuristic violations in these apps.
- Specify place of occurrence
- Provide problem description by referencing screenshots
- Assign a severity rating and justify it
- Give a summarized justification and design recommendations

Since you are working with static screenshots, you may make reasonable assumptions (e.g., missing system feedback, navigation depth, expected user behavior). All assumptions must be clearly stated and justified in your report.

Task 1: Heuristic Evaluation without AI: demonstrate your ability to independently apply Nielsen's 10 usability heuristics without AI assistance. Note: you are only required to identify six violations in both apps.

Task 2: Heuristic Evaluation with AI (Structured Prompt): Use AI to perform usability evaluation using the AI prompt provided below.

Structured AI Prompt

As a Usability expert perform a heuristic evaluation using all 10 of Nielsen's usability heuristics using only the provided screenshots and task description. Identify six heuristics violated, specify place of occurrence by referencing screenshots, and give problem description. Also, assign a severity rating from 0 (not a usability problem) to 4 (usability catastrophe), and justify each severity rating in a table. Lastly, give summarized justification and design recommendations. Your evaluation should consider the interaction across screens, not individual screens in isolation.

Deliverables

You are required to submit one heuristic evaluation report consisting of the following two main components:

1. **Heuristic Evaluation Table**

A structured table covering both applications, including:

- Heuristic violated
- Place of occurrence (Screenshot reference)
- Problem description
- Severity rating

  Assign a **severity rating from 0-4, where:**

  0 = Not a usability problem
  1 = Cosmetic issue
  2 = Minor usability problem
  3 = Major usability problem
  4 = Usability catastrophe
- Brief rationale for severity rating

The table should clearly demonstrate systematic coverage of all heuristics for both apps.

2. **Summary and Justification**

A written summary that includes key usability issues identified in each app, design recommendations explicitly linked to Nielsen's heuristics and prioritized by severity, explaining why certain issues are more critical. Finally, compare AI-generated outcome (task 2) with task 1 results.

Submission Instructions

You are required to upload your completed Usability Analysis Report to Blackboard as a single document (PDF or Word). The report should be clearly organized.

Grading Criteria

Your submission will be evaluated on:

- Correct application of Nielsen's 10 heuristics
- Justification of severity ratings
- Summarized justification and design recommendations