Enhancing User-centered Design with Large Language Models

Bruno Gadelha
Institute of Computing
Federal University of Amazonas - UFAM
Manaus-AM, Brazil
bruno@icomp.ufam.edu.br

Thiago Queiroz
Institute of Computing
Federal University of Amazonas - UFAM
Manaus-AM, Brazil
thiago.queiroz@icomp.ufam.edu.br

Cleidson R. B. de Souza
Faculty of Computing
Federal University of Para - UFPA
Belém-PA, Brazil
cleidson.desouza@acm.org

Tayana Conte
Institute of Computing
Federal University of Amazonas - UFAM
Manaus, Brazil
tayana@icomp.ufam.edu.br

Igor Steinmacher
School of Informatics, Computing and Cyber Systems
Northern Arizona University - NAU
Flagstaff-AZ, USA
igor.steinmacher@nau.edu

***Abstract*—User-centered design (UCD) is a foundational approach in software development that prioritizes users' needs, behaviors, and expectations throughout the design process. For instance, Design Sprints effectively integrate UCD into fast-paced software development, enabling teams to fail fast, learn quickly, and refine solutions based on real user feedback. This paper explores the potential of Large Language Models (LLMs) as an integrated component of UCD methodologies, specifically within the Design Sprint framework. Our goal is to analyze how LLMs can assist designers throughout different phases of the Design Sprint, improving efficiency and aiding in developing user-centered solutions. To achieve this goal, we conducted a study with Computer Science, Software Engineering, and Design students during a course on innovative software product engineering. The study was structured into three phases: (i) theoretical instruction, (ii) practical application of the design sprint methodology with LLM integration, and (iii) presentation and discussion of the developed products. We conducted focus groups to gather insights on the participants' experiences using an LLM as a support tool for Design Sprint activities. Our results provide insights into how LLM can be leveraged to assist in rapid product design while preserving UCD principles.**

***Index Terms*—User-centered design, Design Sprint, LLM**

I. Introduction

User-centered design (UCD) is a foundational approach in software development that prioritizes users' needs, behaviors, and expectations throughout the design process [15]. It involves understanding who the users are, their motivations and goals, and the context in which they will interact with the system. This perspective leads designers to address questions such as: Who will use this technology? What do they aim to accomplish? How much effort are they willing to invest in learning the system? Will they operate it individually or collaboratively? [2]. By considering these aspects early in the development cycle, UCD ensures that human capabilities and limitations are factored into design decisions, resulting in more intuitive, effective, and inclusive solutions [5].

Among the methodologies that operationalize UCD, Design Thinking stands out as a human-centered, iterative framework structured around five stages (Empathize, Define, Ideate, Prototype, and Test) that foster creativity, collaboration, and experimentation [14]. Despite its flexibility in tackling complex problems, its open-ended nature can extend timelines, making it less suitable for fast-paced environments. To address this, Design Sprint [9], developed by Google Ventures, offers a structured, time-boxed variation composed of five phases (Understand, Sketch, Decide, Prototype, and Test) that condense months of design work into a single week [15]. This accelerated process enables teams to ideate, prototype, and validate solutions rapidly, simulating user interactions, testing assumptions, and collecting real feedback without building full-scale products [8], thereby supporting evidence-based decisions while reducing the risks and costs of traditional development cycles [9].

A persistent challenge in applying UCD methodologies lies in their heavy reliance on the continuous involvement of diverse stakeholders like end users, business analysts, designers, engineers, and domain experts, each contributing unique insights to align products with user expectations and business goals [1], [4], [16]. While such interdisciplinary collaboration enriches design outcomes, it also increases financial and human costs, extends development timelines, and adds coordination complexity. Gathering diverse inputs, conducting usability research, iterating on designs, and validating solutions can lead to lengthy cycles. Even agile approaches like Design Sprint, intended to streamline development, still require expert facilitation, structured problem-solving, and multiple validation steps, making them resource-intensive and difficult to scale [9].

In this context, there is a growing need for tools that help designers and stakeholders apply UCD principles more efficiently while preserving the quality of user experiences. This paper explores the potential of generative Artificial Intelligence (genAI), particularly Large Language Models (LLMs), to complement UCD practices by supporting key activities such as ideation, prototyping, and user research [6]. We focus on the Design Sprint as a representative UCD framework and examine how LLMs can assist designers across its phases to increase efficiency and maintain user-centered outcomes. To this end, we conducted a study with 14 students from Computer Science, Software Engineering, and Design undergrad courses who, during a four-week course on innovative software product engineering, completed a final project using an adapted version of the Design Sprint. By analyzing their experiences and project outcomes, we identify points in the methodology where LLMs can provide meaningful support, ultimately enhancing the effectiveness of user-centered design.

II. Research Method

To investigate the use of LLMs in Design Sprint, we conducted a course on Advanced Topics in Software Engineering. The course, which lasted four weeks with weekly online meetings, was attended by 14 students from Software Engineering, Computer Science, and Design undergrad courses. Most of the students had no prior experience with user-centered design approaches or the Design Sprint methodology, although a few were familiar with the concept of Design Thinking from previous coursework. All participants signed an informed consent form authorizing the collection and analysis of materials generated throughout the course for research purposes.

The course was organized into three phases: (i) theoretical instruction, covering fundamental concepts of UCD, the Design Sprint methodology, and prior studies on automating Software Engineering activities; (ii) practical execution of an adapted Design Sprint, in which students developed a software product by following structured assignments for each stage of the methodology. To accommodate time constraints and availability, the standard framework was adjusted, and instead of working on a predefined problem, each team selected its own project topic, fostering engagement, ownership, and more authentic interactions with the tools and techniques. This phase also integrated LLM use into key activities, such as ideation and prototyping; and (iii) presentations and discussion, where teams showcased their prototypes, shared experiences, highlighted the benefits and limitations of using LLMs, and reflected on how these tools influenced their decision-making, creativity, and overall design process.

A. Design Sprint Adaptation and Task Execution

Each team executed a complete Design Sprint to conceptualize and prototype a technology. There were five teams, four with three students and one with two, totaling 14 participants. The process was adapted to integrate LLMs at key stages while accommodating the short course duration. On the first day, teams freely selected a project topic based on a problem, need, or technological concept of interest, identified potential users, and mapped their main pain points. They then used *ChatGPT* to produce a *solution map* [9], a structured description of user–system interactions from the initial state to successful task completion. This artifact served as a high-level abstraction of the user journey, helping teams break down the problem into logical steps, anticipate user needs and system behaviors, and establish a shared mental model of the solution space.

On the second day, guided by the solution map, teams applied the Crazy 8 technique [9] to stimulate divergent thinking and rapidly generate alternatives. Prompts to *ChatGPT* requested eight distinct solution ideas, and in some cases also produced visual mockups or sketches through image generation capabilities. This structure encouraged creativity, variety, and consideration of multiple approaches before narrowing the focus. The third day was dedicated to idea selection and justification. Teams manually evaluated the AI-generated ideas based on feasibility, relevance, and potential impact, selecting the most promising to prototype. Justifications considered user needs, technical viability, alignment with the identified problem, and the clarity of the AI-generated content, reinforcing human judgment and design ownership.

On the fourth day, teams used *ChatGPT* to generate front-end code, typically in HTML, CSS, and JavaScript, aligned with their desired interface and interaction models. Some students iteratively refined their prompts to improve the quality, structure, and coherence of the generated code. The resulting low-fidelity prototypes were then reviewed, adapted, and refined to better match the team's vision, usability principles, and intended user experience. Throughout the sprint, students used the free version of *ChatGPT* (GPT-4o mini), a multimodal model capable of processing text, images, and audio, which reflected realistic conditions for typical users without subscription-based features.

B. Final Presentations and Focus Group Discussion

At the end of the course, each team presented its prototype, explaining key decisions made during the Design Sprint and reflecting on how LLMs influenced their workflow—from problem framing to prototyping—and shaped their design choices and outcomes. After the presentations, a focus group was conducted to gather insights on participants' experiences with *LLMs as a support tool for Design Sprint activities*, addressing perceived benefits and limitations, the impact of AI-generated content on creativity and problem-solving, the degree of trust and integration into decision-making, and potential improvements for future AI-assisted sprints.

C. Deliverables and Data Collection

To analyze the effectiveness and limitations of LLMs in supporting Design Sprint activities, teams were required to submit the following materials:

- **Final Report:** A document detailing each stage of the Design Sprint, including scenario description, AI-generated solution maps, the ideation process, and prototype evaluation;
- **Chat Transcripts:** A complete record of all interactions with *ChatGPT* (in PDF format);

- **Prototype Code:** The unmodified AI-generated prototype code and the final version incorporating manual modifications (both provided in ZIP format);
- **Screenshots of the Prototype:** Visual documentation of the generated interface; and
- **Focus Group Responses:** A summary of participants' qualitative feedback from the focus group discussion.

D. Data Analysis

A qualitative analysis of the teams' final reports, ChatGPT interaction logs, and submitted prototype artifacts was conducted. The goal of this analysis was to identify how LLMs were adopted by students throughout the Design Sprint process, specifically in tasks such as idea generation, scenario structuring, and prototype development, as well as in auxiliary activities like writing project documentation.

To assess the *effectiveness* of LLM usage, we considered how well the AI-supported outputs contributed to the quality, coherence, and completeness of the final deliverables. Effectiveness was not measured by academic grades alone, but rather through a qualitative evaluation of each team's process and outcomes. This included reviewing the creativity and originality of the proposed solutions, the clarity and structure of the generated artifacts, and the appropriateness of AI integration across sprint phases. These evaluations were conducted independently by the authors, prior to the focus group and without reference to any final course grades, ensuring that the analysis reflected the perceived design value and process quality rather than academic performance per se.

No formal usability evaluation of the prototypes was conducted, and code-level inspection was limited to reviewing the fidelity of generated interfaces with respect to each team's stated objectives. The purpose of the analysis was not to judge technical correctness but to understand how LLMs supported, or constrained, the students' ability to carry out user-centered design tasks within a time-constrained, sprint-based workflow.

III. Results

The study revealed distinct patterns in how teams used *ChatGPT*, ranging from extensive integration in ideation, problem framing, and prototyping to more selective application for documentation or concept validation. These variations, shaped by prompt quality, prior knowledge, and team dynamics, directly influenced the depth, creativity, and relevance of outputs. To capture these nuances, we examined deliverables, prompts, prototypes, and focus group feedback, with results discussed in detail in the following sections.

A. Use of ChatGPT by Each Team

Our qualitative analysis revealed different strategies for AI integration. Some teams used ChatGPT extensively for ideation, allowing the AI to generate multiple alternative solutions later refined by human input. Others focused on using ChatGPT for prototyping, generating code snippets in HTML, CSS, and JavaScript, or other web development technologies. A few teams leveraged the AI mainly for content generation, particularly in structuring problem statements and drafting user stories. Table I highlights how each team used ChatGPT during their work.

One of the most noticeable patterns was the varying degrees of reliance on AI across teams. While some groups treated ChatGPT as a collaborative assistant, iteratively refining its suggestions with critical human input, others adopted a more passive approach, using the model's outputs with minimal adaptation. For example, Team 1 generated a list of solution alternatives with ChatGPT and selected one directly for implementation, without significant refinement or iteration. This highlights a central factor in LLM effectiveness: its contribution is amplified when designers remain actively engaged in questioning, reworking, and contextualizing the generated content, whereas accepting outputs almost verbatim can reduce creative ownership and limit contextual tailoring.

> *"We asked ChatGPT to generate eight possible solutions and chose the best one among them, which we then implemented as suggested."* - Team 1

In contrast, another group (Team 5) reported using ChatGPT responses as a starting point, but enriching them with insights from team discussions and external visual references. They manually edited the generated code and reorganized interface components to better align with their design vision. This hybrid approach reflects a more intentional and critical use of the AI tool, reinforcing the idea that effective AI-supported design requires active human curation.

> "Although ChatGPT offered us several interface examples, we decided to combine its suggestions with elements from previous user feedback and paper sketches we had developed earlier." - Team 5

These contrasting strategies suggest that overreliance on LLMs can lead to generic outcomes, while active engagement allows teams to preserve creativity and align outputs with user-centered goals.

B. Insights from the Focus Group Discussion

To complement the analysis of project deliverables, a focus group was conducted at the end of the course to gather students' perspectives on the use of ChatGPT in the Design Sprint process. The discussion provided deeper insights into how AI influenced their workflow, its perceived advantages, and the challenges faced during the sprint.

One of the most prominent themes that emerged was the role of ChatGPT as a time accelerator. Several students noted that the traditional Design Sprint process typically spans five days to allow for ideation, discussion, and refinement. However, with AI assistance, many teams completed their sprints in significantly less time, in some cases even in a single day. The ability to rapidly generate structured outputs enabled students to focus more on analyzing ideas rather than manually creating them. As one participant stated:

> *"Instead of taking five days, we finished in one day. Although the methodology suggests following the five-day process, it was much more efficient than I expected."*

TABLE I
COMPARISON OF TYPES OF CHATGPT PROMPTS USED BY TEAMS

Team	Scenario Definition	Ideation (Crazy 8)	Prototyping	Refinement and Documentation
Team 1	AI-generated user personas and user scenarios	AI-generated variations of product solutions	Limited AI usage in prototyping	Used AI to structure report sections and improve wording
Team 2	Prompted AI to suggest key features based on user pain points	AI-generated sketches and idea comparisons	AI-generated front-end code (HTML/CSS)	Minor AI-assisted edits for final report
Team 3	AI-generated market analysis and competitor benchmarking	Basic brainstorming with minimal AI input	No AI involvement in prototyping, manual development	AI-assisted writing for clarity and coherence
Team 4	Created AI-generated step-by-step user journey maps	Combined AI-generated sketches with human iteration	Prompted ChatGPT for front-end code snippets	Limited AI use in final documentation
Team 5	AI-assisted validation of initial product hypothesis	Used AI for feature suggestions rather than full idea generation	No AI involvement in prototyping, manual development	AI consulted for grammar and structure improvements

While the efficiency gains were widely appreciated, some students raised concerns about the potential loss of creativity and depth in idea generation. Traditional brainstorming often involves extensive discussions, allowing spontaneous and innovative ideas to emerge. In contrast, AI-generated solutions tend to follow predefined patterns based on existing data, which may limit the diversity of generated ideas. As one student pointed out:

> *"Usually, we come up with great ideas through discussions. ChatGPT only produces results based on the parameters we provide."*

Another recurring topic was the role of critical thinking in AI-assisted design. While ChatGPT provided structured suggestions, it could not evaluate whether an idea was truly innovative or aligned with real-world constraints. Participants emphasized the need for human oversight to critically assess AI-generated content before accepting it as final. One student clearly articulated this concern:

> *"If you don't have a strong foundation, you won't even know whether what it generated is correct or incorrect."*

The focus group discussion also explored whether ChatGPT would be equally effective in a multidisciplinary team setting. Some students believed that AI could serve as a valuable support tool, while others argued that the value of human discussion and debate should not be underestimated. The consensus was that ChatGPT works best when used as a *complementary* tool rather than a replacement for human interaction. The following quotes from students expressed this perspective by stating:

> *"We can never replace human decision-making with a tool. The tool should be used as support, not as a substitute."*

> *"The idea came from us, but the rest was done with the help of ChatGPT."*

Finally, some students noted that ChatGPT's effectiveness depended on the quality of the prompts provided. Teams that structured their questions well obtained more useful and relevant responses, whereas teams that provided vague prompts received more generic AI-generated outputs. This highlights the importance of prompt engineering skills when using AI in creative and technical workflows. This is illustrated in the next section.

IV. ANALYSIS OF CHATGPT PROMPTS IN THE DESIGN SPRINT PROCESS

As shown earlier, integrating LLMs into the Design Sprint provided valuable support for brainstorming, refining ideas, and accelerating prototyping, but its effectiveness was strongly influenced by the quality of the prompts used. At the start of the sprint, teams employed ChatGPT to define user personas, map user journeys, and identify key pain points. Those who crafted detailed, layered prompts—such as requesting personas with defined roles, needs, and frustrations, followed by a problem map outlining the main steps of the current process—received structured, context-aware outputs that informed later stages. In contrast, teams that issued broad or vague requests obtained generic, less actionable results, with personas disconnected from the actual problem space.

During the ideation stage, many teams simply asked for "eight ideas," which often led to repetitive or overly similar suggestions. The most effective teams specified constraints to ensure diversity, such as requiring at least one idea focused on automation, one on gamification, and another on community-driven solutions. This structure encouraged ChatGPT to explore multiple perspectives rather than converging on common patterns, resulting in a richer and more creative solution set.

When selecting a solution, some teams relied on ChatGPT to directly choose the "best" idea, leading to subjective and poorly justified recommendations. Others adopted a more analytical approach by asking the AI to compare solutions based on clear evaluation criteria, such as feasibility, scalability, and user experience, and then to

V. DISCUSSION

Our findings align with and extend existing literature on *AI-assisted Design Sprints* and *LLMs in User-Centered Design (UCD)*. Prior studies have emphasized the potential of LLMs to enhance creativity and streamline software engineering tasks [6], [13]; our work provides empirical evidence on how these tools influence the design process within an agile, sprint-based framework, revealing both significant efficiency gains and important creative trade-offs. A central outcome was ChatGPT's ability to accelerate the Design Sprint, enabling students to complete multi-day activities in a single session by generating structured outputs such as personas, solution maps, and HTML prototypes. This confirms earlier observations of AI as a design accelerator [3], with measurable reductions in cognitive workload and manual effort. However, this compression of time came at the cost of reduced creative depth. AI-generated outputs, while efficient, lacked the richness, spontaneity, and reframing potential that emerge from human-led brainstorming, echoing concerns raised in [10]. Students consistently noted that ChatGPT could not replicate the diversity of ideas fostered by interdisciplinary debate, highlighting the need to balance speed with opportunities for deeper creative exploration.

Another important finding relates to the role of AI in prototyping and documentation. Similar to results reported in [17], several teams used ChatGPT to produce HTML and CSS code, effectively lowering technical barriers for non-programmers and accelerating interface development. This supports the view that LLMs can democratize prototyping by enabling participants with limited coding skills to produce functional outputs. Yet, heavy reliance on AI-generated code raised concerns about correctness, maintainability, and alignment with usability standards. These concerns mirror prior warnings in [12], [13], underscoring the need for human validation and revision to ensure both technical quality and design integrity.

The study also reinforces the literature on the importance of human-AI collaboration and critical thinking in design contexts [10], [19]. While some teams used ChatGPT iteratively by refining, contextualizing, and integrating outputs into their workflow, others accepted its suggestions with minimal adaptation, often resulting in generic solutions. This variation reflects findings by [18] and the cautions outlined in [7], which emphasize that LLMs are most effective when used as collaborative partners rather than autonomous design agents. The ability to critically evaluate AI outputs remains a decisive factor in achieving meaningful results.

Our results also contribute to the ongoing debate on AI's role in multidisciplinary teams [11], [18]. While ChatGPT streamlined aspects of the Design Sprint, it did not replace the value of collaborative human interaction. Students stressed that discussions and debates were essential for refining ideas and ensuring solutions were aligned with real-world constraints, supporting the perspective in [19] that AI should be integrated into iterative, human-centered workflows rather than used as a substitute for team deliberation.

Drawing from these findings and their connection to existing research, we suggest that AI-assisted Design Sprint methodologies should aim to balance efficiency and creativity, combining the structured outputs of AI with the unpredictability of human ideation. Training in prompt engineering can help maximize the relevance of AI responses, while strengthening critical evaluation skills is essential to prevent uncritical adoption of flawed outputs [7]. AI can be leveraged effectively for prototyping, provided that human review ensures usability and technical feasibility, and its integration into multidisciplinary workflows should prioritize complementing, not replacing, collaborative decision-making. Key recommendations and lessons learned from this study are summarized in the *Takeaway Messages* box below:

Takeaway Messages

- **Balance speed and creativity:** ChatGPT accelerates Design Sprint activities, but excessive reliance may limit creative depth and problem reframing.
- **Human oversight is critical:** Well-structured prompts improve output quality, yet AI-generated content requires validation and adaptation to context.
- **AI as a complement, not a replacement:** ChatGPT can support prototyping and ideation, but multidisciplinary collaboration remains essential for effective design.

A. Limitations

This study has limitations that should be considered when interpreting the results. Participants were students in a short-term course, most without prior experience in UCD or Design Sprint; to mitigate this, a theoretical phase was included to establish a shared foundation. The evaluation was primarily qualitative, with no formal usability testing or external expert reviews; two researchers independently analyzed the deliverables, triangulating data from reports, AI logs, and transcripts to reduce subjectivity. Finally, students used the free version of ChatGPT (GPT-4o mini), which limited access to advanced features but provided ecological validity by reflecting common conditions for non-specialist users.

VI. Conclusion

This work contributes to an understanding of how LLMs intersect with UCD practices in a sprint-based framework. By providing empirical findings and actionable guidance, we aim to advance the conversation from hype to thoughtful integration, where AI accelerates, but does not overshadow, human-centered innovation.

Our results demonstrate that LLMs can act as accelerators for ideation and prototyping, enabling rapid iteration. However, we also highlight risks associated with over-reliance, reduced creativity, and diminished critical engagement. The study emphasizes the importance of human oversight, thoughtful prompt design, and multidisciplinary collaboration in AI-supported design workflows.

Future work should explore how these findings scale to professional contexts, test the long-term effects of AI-supported design learning, and examine how different LLMs impact creativity, inclusivity, and collaboration. As AI capabilities evolve, our design methods and pedagogical strategies should also evolve.

ACKNOWLEDGMENTS

We thank the participants of the empirical study and the members of the USES Research Group for their valuable support. This work was financially supported by CNPq (grants 314797/2023-8, 443934/2023-1, 445029/2024-2, 420406/2023-9, 442779/2023-2, 444802/2024-0 and 313137/2025-0), CAPES (AUXPE-CAPES-PROEX, Funding Code 001), and FAPEAM through the POSGRAD 25-26 program.

REFERENCES

[1] Bill Albert, Tom Tullis, and Donna Tedesco. *Beyond the usability lab: Conducting large-scale online user experience studies*. Morgan Kaufmann, 2009.

[2] Richard Banfield, C Todd Lombardo, and Trace Wax. *Design sprint: A practical guidebook for building great digital products*. " O'Reilly Media, Inc.", 2015.

[3] Hubert Ekvall and Patrik Winnberg. Integrating chatgpt into the ux design process: Ideation and prototyping with llms. Master's thesis, Department of Informatics, Umeå University, 2023.

[4] Susan Gasson. The reality of user-centered design. *Journal of Organizational and End User Computing (JOEUC)*, 11(4):5–15, 1999.

[5] Holly AH Handley, Judi E See, and Pamela A Savage-Knepshield. Human readiness levels and human views as tools for user-centered design. *Systems Engineering*, 27(6):1089–1102, 2024.

[6] Xinyi Hou, Yanjie Zhao, Yue Liu, Zhou Yang, Kailong Wang, Li Li, Xiapu Luo, David Lo, John Grundy, and Haoyu Wang. Large language models for software engineering: A systematic literature review. *ACM Trans. Softw. Eng. Methodol.*, 33(8), December 2024.

[7] Artur Klingbeil, Cassandra Grützner, and Philipp Schreck. Trust and reliance on ai — an experimental study on the extent and costs of overreliance on ai. *Comput. Hum. Behav.*, 160(C), November 2024.

[8] Jake Knapp, John Zeratsky, and Braden Kowitz. The design sprint, 2016. Accessed: 2025-04-02.

[9] Jake Knapp, John Zeratsky, and Braden Kowitz. *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*. Simon and Schuster, 2016.

[10] Yuwen Lu, Yuewen Yang, Qinyi Zhao, Chengzhi Zhang, and Toby Jia-Jun Li. Ai assistance for ux: A literature review through human-centered ai. *Proceedings of ACM Conference*, 2024.

[11] Neil Maiden, Konstantinos Zachos, Kostas Petrianakis, James Lockerbie, Chantakan Chanpalangsri, Holger Ernst, and Seyfettin Kara. Beyond just generative ai for discovering software opportunities. *IEEE Software*, 42(3):41–49, 2025.

[12] Nhan Nguyen and Sarah Nadi. An empirical evaluation of github copilot's code suggestions. In *Proceedings of the 19th International Conference on Mining Software Repositories*, MSR '22, page 1–5, New York, NY, USA, 2022. Association for Computing Machinery.

[13] Eleftherios Papachristos, Yavuz Inal, Carlos Vicient Monllao, Eivind Arnstein Johansen, and Mari Hermansen. Integrating ai into design ideation: Assessing chatgpt's role in human-centered design education. *TechRxiv Preprints*, 2024.

[14] Hasso Plattner, Christoph Meinel, and Ulrich Weinberg. *Design thinking*. Springer, 2009.

[15] E Frank Ritter, D Gordon Baxter, and F Elizabeth Churchill. *Foundations for designing user-centered systems: What system designers need to know about people*. Springer, 2014.

[16] Frank E. Ritter, Gordon D. Baxter, and Elizabeth F. Churchill. *Foundations for Designing User-Centered Systems*. Springer, 2021.

[17] Shuhong Xiao, Yunnong Chen, Jiazhi Li, Liuqing Chen, Lingyun Sun, and Tingting Zhou. Prototype2code: End-to-end front-end code generation from ui design prototypes. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 88353, page V02BT02A038. American Society of Mechanical Engineers, 2024.

[18] Qian Yang, Aaron Steinfeld, and John Zimmerman. Re-examining whether, why, and how human-ai interaction is uniquely difficult to design. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–23, 2020.

[19] Zhibin Zhou, Yaoqi Li, and Junnan Yu. Exploring the application of llm-based ai in ux design: An empirical case study of chatgpt. *Human-Computer Interaction*, 2024.