Canvil: Designerly Adaptation for LLM-Powered User Experiences

K. J. Kevin Feng*
University of Washington
Seattle, WA, USA
kjfeng@uw.edu

Q. Vera Liao
Microsoft Research
Montréal, QC, Canada
veraliao@microsoft.com

Ziang Xiao*
Johns Hopkins University
Baltimore, MD, USA
ziang.xiao@jhu.edu

Jennifer Wortman Vaughan
Microsoft Research
New York City, NY, USA
jenn@microsoft.com

Amy X. Zhang
University of Washington
Seattle, WA, USA
axz@cs.uw.edu

David W. McDonald
University of Washington
Seattle, WA, USA
dwmc@uw.edu

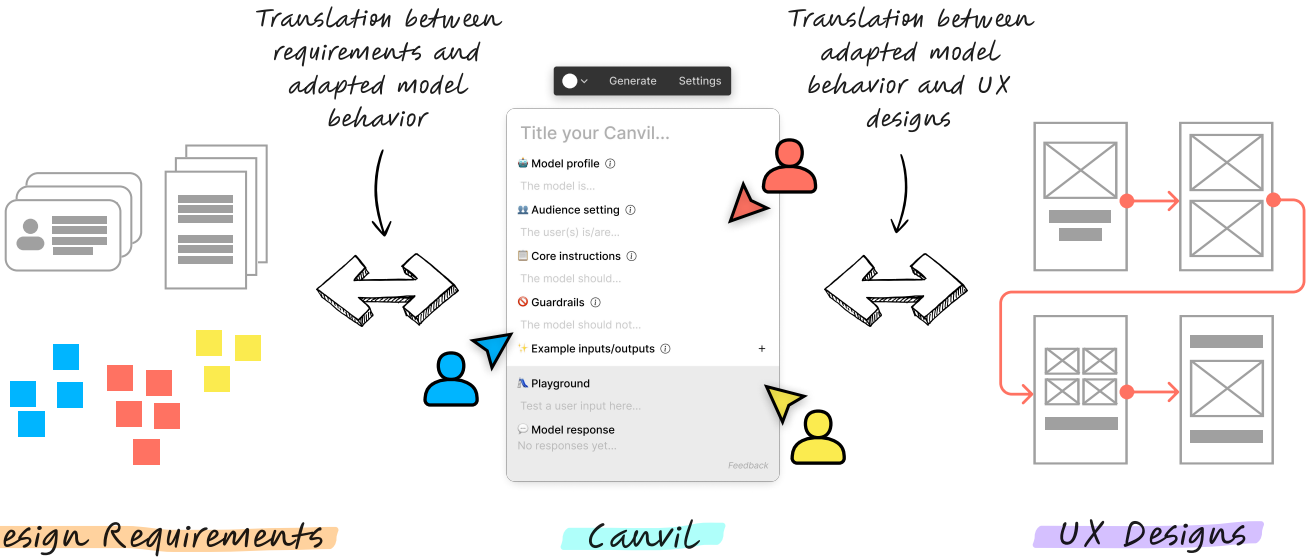

Figure 1: We propose *designerly adaptation* as a means of exploring LLMs as a design material, enabling a two-way translation between design requirements and UX designs for LLM-powered applications. We then present Canvil, a Figma widget that operationalizes designerly adaptation in design canvases. By using Canvil as a probe, we found that designerly adaptation surfaced LLM behavior that allowed designers to co-evolve design requirements and UX designs, and thoughtfully craft end-user interaction with LLMs.

ABSTRACT

Advancements in large language models (LLMs) are sparking a proliferation of LLM-powered user experiences (UX). In product teams, designers often craft UX to meet user needs, but it is unclear how they engage with LLMs as a novel design material. Through a formative study with 12 designers, we find that designers seek a translational process that enables design requirements to shape and be shaped by LLM behavior, motivating a need for *designerly adaptation* to facilitate this translation. We then built Canvil, a Figma widget that operationalizes designerly adaptation. We used Canvil as a probe to study designerly adaptation in a group-based design study (6 groups, $N = 17$), finding that designers constructively iterated on both adaptation approaches and interface designs to enhance end-user interaction with LLMs. Furthermore, designers identified promising collaborative workflows for designerly adaptation. Our work opens new avenues for processes and tools that foreground designers' human-centered expertise when developing LLM-powered applications.

*Work done while at Microsoft Research.

CHI '25, April 26-May 1, 2025, Yokohama, Japan

ACM ISBN 979-8-4007-1394-1/25/04
https://doi.org/10.1145/3706598.3713139

CCS CONCEPTS

• **Human-centered computing → Interaction design process and methods**; **Empirical studies in collaborative and social computing**.

KEYWORDS

large language models, user experience, design practice

ACM Reference Format:

K. J. Kevin Feng, Q. Vera Liao, Ziang Xiao, Jennifer Wortman Vaughan, Amy X. Zhang, and David W. McDonald. 2025. Canvil: Designerly Adaptation for LLM-Powered User Experiences. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26-May 1, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 22 pages. https://doi.org/10.1145/3706598.3713139

1 INTRODUCTION

A paradigm shift is underway for integrating artificial intelligence (AI) capabilities into everyday user-facing technologies. Large pre-trained AI models, most notably large language models (LLMs), have versatile natural language capabilities that unlock novel interactive techniques and interfaces for more intuitive and customizable user experiences across a wide spectrum of applications [2, 77, 100, 113]. However, these promises also come with numerous concerns. Integrating LLMs into a domain without careful consideration of the user contexts surrounding model use and implementation of appropriate guardrails may result in user experiences that perpetuate societal biases [88], threaten users' sense of well-being [86, 91], or otherwise do harm [19, 27, 89, 92, 108].

As technology development practitioners, designers[1] are uniquely positioned to mitigate these concerns [35, 54, 98, 114, 116] and operationalize growing calls for human-centered AI [51, 94]. Designers' work often involves aligning technological capabilities with user needs, such that the technology addresses (or makes progress towards addressing) pain points identified in user research [35, 97]. Designers are trained in human-centered design methods that allow them to understand users and usage contexts, prototype potential solutions with relevant technology, and iterate on those solutions based on their understanding of users or user feedback [7]. Yet, prior work has shown that designers face diverse procedural challenges (e.g., difficulties collaborating with the engineering teams training the models [97, 99]) and instrumental challenges (e.g, lacking means to effectively work with the models [35, 98, 114]) when working with AI [99, 115]. In efforts to address these challenges, researchers have situated AI as a design material to highlight its material properties—including the technology's capabilities, limitations, and adaptability [53]—for designers to better understand and apply AI in the context of their design problems [29, 33, 114]. A good *designerly understanding* of AI can help designers ideate on new AI-powered design ideas, mitigate AI's varying impact for different user scenarios, collaborate with design and non-design practitioners, and reinforce user-centered perspectives amongst the team and throughout the product development cycle [54, 103].

[1]We consider designers to be anyone who has an active, hands-on role in designing the user experience of a product and/or feature. This includes job titles such as UX designer/researcher, content designer, and product designer.

The advent of powerful pre-trained LLMs introduces new opportunities for engaging with AI as a design material. First, adaptability emerges as a key materialistic property of LLMs—these models are responsive to adaptation via fine-tuning [6, 28, 42, 75] and prompt-based methods [70, 71, 112]. In fact, due to resource-intensive training of LLMs, it has become a standard practice for individual developers and development teams to adapt "base" LLMs from a small handful of providers (e.g., Anthropic, Google, OpenAI) instead of training custom in-house models [77]. Moreover, natural language interaction and adaptation democratizes AI experimentation for practitioners—including designers—traditionally excluded from AI conversations due to limitations in technical expertise [54, 79]. Despite this, there has been limited exploration of designers' interaction with—let alone adaptation of—LLMs in practice as they contribute to LLM-powered features and products.

In this paper, we first conduct a formative interview study with 12 designers experienced in designing features and products that use LLMs to better understand their workflows and desiderata when crafting LLM-powered UX. From our interviews, we identify a need for a translational process that allow design requirements (e.g., findings from user research, product goals) to directly shape model behavior, and vice versa. Drawing from Cross' notion of *design codes*—systematic ways of representing, transmitting, and translating design knowledge [24]—we propose *designerly adaptation for LLMs* (henceforth "designerly adaptation") as a new design code to facilitate this two-way translation. We operationalize designerly adaptation by building Canvil, a Figma[2] widget that supports structured authoring of model behavioral specifications based on design requirements to adapt LLMs, testing of adapted models on diverse user inputs, and integration of model outputs into interface designs. We then use Canvil as a technology probe to study designerly adaptation with 6 groups with a total of 17 designers. We demonstrate the effectiveness of our probe by collecting concrete, empirical insights into how designers engage in designerly adaptation through Canvil. Specifically, we find that designers iteratively translated diverse design requirements into LLM behavioral specifications. This allowed them to simultaneously adapt their designs to enhance user interaction with LLMs, while setting or refining behavioral constraints for the model using their designs. Furthermore, designers recognize promises in collaboratively engaging with designerly adaptation to share resources and knowledge with design and non-design stakeholders. They are optimistic about integrating designerly adaptation into their own workflows and note additional procedural and instrumental questions to address in practice. Concretely, our work makes the following contributions:

- Designerly adaptation—a process bridging material exploration of LLMs with design processes—motivated by insights from a formative study with designers who craft LLM-powered UX.
- Canvil, a technology probe to operationalize designerly adaptation in the form of a Figma widget and allow us to study it in practice.
- Empirical learnings uncovered by Canvil about how designers engage in designerly adaptation in a task-based design study.

[2]Figma is a popular collaborative design tool: https://www.figma.com/.

- A discussion of our learnings' implications on design practice and beyond, including a proposed workflow for designerly adaptation to orient future work and a reflection of LLMs' sociomaterial implications on collaborative practices within product teams.

2 RELATED WORK

2.1 AI as a Design Material

Robles and Wiberg argue that recent advances in computational technologies bring about a "material turn"—a transformation within interaction design that allows for the shared use of material metaphors (e.g., flexibility) across physical and digital worlds [85]. Indeed, prior works have discussed AI as a *design material* [9, 29, 34, 35, 54, 59, 114–116], and highlighted why AI's materiality can make it uniquely difficult to design with [9, 29, 54, 99, 115]. AI is often treated as a black box to non-technical stakeholders such as some designers, making it challenging to tune user interactions to often unpredictable and complex model behavior [9, 114]. In addition, AI's technical abstractions are often divorced from concepts designers are familiar with [99], and designers consequently struggle with creatively using or manipulating the material to generate design solutions [35, 54]. AI models are also non-deterministic and fluid in nature—they may evolve with new data or user input, and can be intentionally steered towards desirable behaviors with choices of data, algorithms, parameters, and so on [115]. Without a concrete material understanding to begin with, designers face challenges grasping the nature of these uncertainties [98, 114] or the opportunities to shape the design materials for desirable UX [54]. Yang et al. [115] showed that these challenges of working with AI as a design material persist through the entire double-diamond design process—from identifying the right user problem to be solved by AI to designing the right UX to solve the user problem. These challenges pose barriers to developing human-centered AI—expanding the algorithm-centered scope of AI and applying methods from human-centered design to ensure that the development of the technology can better serve human needs [17, 51, 83, 93, 94].

Researchers and practitioners have developed processes and tools to alleviate some of these challenges. ProtoAI [98] combines exploration of models with UI prototyping, while advocating for designers' active shaping of the AI design material (e.g., choosing models and setting parameters) by user needs. Feng et al. [35] found that hands-on "fabrication" of the design material through a UI-based model training tool bolstered understanding and connection between AI properties and UX goals. fAIlurenotes [68] is a failure analysis tool for computer vision models to support designers in understanding AI limitations across user groups and scenarios. Other efforts include process models [99] and "leaky abstractions" [97] that facilitate collaboration between designers and model developers, and human-AI design guidelines [4, 38, 65].

Recent advances in LLMs can simultaneously address and exacerbate some of the aforementioned challenges. The barrier to tinkering with AI has significantly lowered thanks to the use of natural language as a primary mode of interaction and easily accessible tools such as ChatGPT. Concerns, however, have also arisen over the lack of transparency and controllability in LLMs due to their complex technical architectures [55]. Yet, because of LLMs' powerful capabilities, there is significant interest in exploring their integration into user-facing applications [51, 77]; as such, designers should be prepared to work with them as a design material [50]. While past work advocated for a better "designerly understanding" upon studying how to support designers in working with models [54], our work specifically explores involving designers in the practice of adapting LLMs. Adaptation and understanding can be related and synergistic, but our focus is on adaptation—a common practice in product teams [77]—and how it can support designers to craft LLM-powered UX.

2.2 Adaptation of Large Language Models

A fundamental property of LLMs not present in their smaller predecessors is the ease with which their behavior can be adapted [15, 70]. While the breadth of LLMs' out-of-the-box capabilities may seem impressive, researchers have recognized the importance of adapting LLMs for enhanced performance under specific domains and tasks [28, 42, 103, 112, 113], and aligning model behavior with human preferences and values [6, 32, 75]. Adaptation has thus been a topic of interest to both the AI and HCI communities [52].

Adaptation may take on many forms and can be performed by different stakeholders. AI engineers and data scientists may fine-tune a pre-trained model by using a task-specific dataset to train additional layer(s) of the neural network [28]. Popular technical approaches to fine-tuning include instruction tuning [75, 106], reinforcement learning with human feedback (RLHF) [23, 75], and direct preference optimization (DPO) [80]. More computationally efficient variants, such as low-rank adaptation (LoRA) [42], have also garnered attention.

Besides technical stakeholders, HCI has long observed that end users often adapt their technology—and also learn to perform this adaptation over time—in a process Mackay calls *co-adaptation* [61, 62]. End-user adaptation of LLMs is made accessible by methods that modify model behavior without modifying the model itself, such as system prompting [26, 63, 70, 87, 105]. Different from one-off user prompting, system prompting applies to all individual user inputs for how the model should behave (e.g., "*always respond in a concise manner*"), often targeting an application domain. For instance, a line of work explored instructing the model to behave with a certain "persona" to elicit desirable or adversarial model behaviors [22, 26, 87, 105, 109]. However, system prompts can be challenging to author [118], and there is not yet an established "gold standard" for prompt authoring, system or otherwise. Co-adaptation is a two-way street [62], and we do not yet have a systematic understanding of how to adapt models even though models can quickly adapt to our prompts. Researchers and practitioners have thus attempted to derive useful prompting formats based on empirical exploration; these include in-context learning (i.e., by providing desired input-output examples) [15, 111, 119], chain-of-thought reasoning [107], and instruction-following [48, 74]. Industry recommendations have also emerged for system prompts, encouraging the specification of elements such as context specification (e.g., "*You are Yoda from Star Wars*"), task definition (e.g., "*You respond to every user input as Yoda and assume the user is a Padawan*"), and safety guardrails (e.g., "*If the user requests inappropriate or offensive responses, you must respectfully decline with a wise Yoda saying*") [63].

In our work, we examine and support *designerly adaptation* of LLMs—a means of empowering designers to explore LLMs as a design material through model adaptation. Unlike many technical adaptation methods, designerly adaptation contributes to human-centered AI by drawing from designers' workflows and expertise. This focuses efforts to ensure that AI-powered applications satisfy user needs and effectively co-adapt with us as they become more pervasive. While we draw from recent developments in and recommended best practices for accessible model adaptation via natural language, we acknowledge that these practices may shift over time as we co-adapt to this new technology. Using currently available techniques is just one possible instantiation of designerly adaptation.

2.3 Interactive Tools for Steering AI Behavior

Literature at the intersection of HCI and AI has introduced a wide range of interactive techniques and tools to aid humans in training and adapting AI models, ranging from ones supporting data scientists to perform data wrangling [39, 49, 57, 104, 110], model training and evaluation [1, 8, 16, 37, 45, 64, 82], managing model iterations [41, 84], and so on, to those allowing for "human-in-the-loop" paradigms at various stages of the model development pipeline, including data annotation [20, 67, 81, 90, 95], output correction [11, 12], and integration testing [21], and explainability.

There has also been interest in "democratizing AI" for domain experts or practitioners without formal technical training to steer model behaviors. Interactive machine learning (iML) [31] is a field responding to this interest by advocating for interactive and incremental model steering through intuitive interfaces and tightly coupled input-evaluation feedback loops. For instance, interfaces for transfer learning [66] have enabled non-technical audiences to transfer learned representations in a model to a domain-specific task. Tools have also encouraged exploratory tinkering of AI models through visual drag-and-drop UIs [18, 30, 56, 58]. Teachable Machine [18] is one such tool that allows users to train models for image, video, and audio classification. Rapsai [30] is a visual programming pipeline for rapidly prototyping AI-powered multimedia experiences such as video editors. In an era where formal knowledge about AI is limited to technical experts, these tools also serve to demystify AI for everyday users.

With the onset of LLMs, barriers to experimenting with AI have lowered even further, paving the path for a new generation of interactive AI tools. Sandbox environments such as OpenAI's playground [72] require no prerequisites besides a grasp of natural language to prompt the model and steer model behavior. However, relying on unstructured natural language alone can be daunting and ineffective. Many tools have risen to the challenge to support prompt engineering with more structured tinkering. Prompt chaining is one such approach [5, 100, 112, 113], by which users can use a node-based visual editor to write prompts for simple subtasks and assemble them to solve a larger, more complex task. PromptMaker [46] and MakerSuite [43] allow the user to rapidly explore variable-infused prompts and few-shot prompting, while ScatterShot [111] helps specifically with curating few-shot prompting examples.

Despite advancements in tooling, support for designers to work with LLMs as an adaptable design material remains limited. Domain-agnostic tools for tinkering with models may not be well-integrated into design workflows, a primary consideration for designers when deciding whether to adopt those tools [34]. Tools that offer integration with design environments (e.g., PromptInfuser [78, 79]) support prototyping with LLMs, but not necessarily deeper adaptation of model behavior. Our work situates interactive adaptation within familiar design environments and processes.

3 FORMATIVE STUDY

Working with AI, especially LLMs, is an emerging practice in the field of UX [33, 99, 115, 117]. We conducted a formative interview study to understand designers' experiences working on LLM-powered products and features amidst industry-wide shifts towards LLMs in 2023 [77] as an initial step in our research.

3.1 Method and Participants

We conducted 30-minute 1:1 virtual interviews with 12 designers at a large international technology company where LLM-powered features and products are actively explored. Our interviews were not initially focused on model adaptation, let alone designing a probe for adaptation. Instead, our goal was to more deeply understand designers' workflows and pain points they face when crafting LLM-powered UX. Thus, we opted for interviews as our method of choice. All interviews were conducted in June and July 2023. At the time of the study, all participants were working on products or services that leveraged LLMs in some capacity; LLM application areas spanned conversational search, question-answering (QA) on domain-specific data, recommendation, text editing and generation, and creativity support tools. Participant details can be found in Table 1 of Appendix A.

Our interviews were semi-structured and revolved around the following topics:

- **Awareness:** to what extent are designers aware of LLMs' capabilities, limitations, and specifications in the context of their product(s)?
- **Involvement:** to what extent are designers involved in discussions or activities that shape where and how an LLM is used in their product(s)?
- **Desiderata:** what do designers desire when crafting LLM-powered UX, with regards to both processes and tools?

Each participant received a $25 USD gift card for their participation. All interviews were recorded and transcribed. Our study was reviewed and approved by the company's internal IRB.

The first author performed an inductive qualitative analysis of the interview data using the qualitative coding platform Marvin[3]. This process started with an open coding round in which initial codes were generated, followed by two subsequent rounds of axial coding in which codes were synthesized and merged into (yet-to-be-named) groups, in line with Braun and Clarke's reflexive thematic analysis [13]. These groups were reviewed and named through an iterative process that involved the disassembling and merging of groups; the named groups became initial themes. The codes and

[3]https://heymarvin.com/

themes were discussed with research team members at weekly meetings. Other research team members also offered supporting and contrasting perspectives (e.g., (dis)agreement on main themes, interpretation of new themes) by writing their own analytical memos, which were also discussed as a team. Any disagreements were resolved through team meetings and asynchronous discussion via comments on the analytical memos.

3.2 Findings

3.2.1 Adaptation was seen as a central materialistic property of LLMs. Participants recognized that products delivering compelling, robust UX were not powered by out-of-the-box "base" LLMs, but required adaptation based on user needs and contexts. P5 gave an example where an out-of-the-box model inappropriately prompted the user for a riddle in a workplace setting: *"[The experience] is not quite right, cause it's in [workplace software] and it's like, tell me a riddle. I'm at work!"* P4 and P7 both stated a design goal for their products (in conversational search and domain-specific question-answering, respectively) is to accommodate non-technical users, which can be achieved if the model can *"tailor the language depending on technical ability, and gradually introduce [users] to more complicated concepts and terminology"* [P7]. P2 shared that their team settled on using three adapted versions of the same model to customize UX within their product: *"There are actually three incarnations of the same model, each with slightly different parameters [...] based on the prompt that users give, we select which of those 3 incarnations we wanna use."*

In their design work, participants appreciated the ability to steer an LLM with natural language to envision more flexible and adaptable AI-powered UX. Some were familiar with how lightweight adaptation techniques, such as writing system prompts, can be used to specify model behaviors based on design specifications. P12 discussed an example of how this may work in an entertainment system setting:

> *"You might want to generate enthusiasm more, right? So the LLM that goes over there might have a UX layer that feels different. The system prompt can say, remember, you're a machine that wants to get users enthused."* [P12]

P12's approach points to an **important distinction in outcome evaluation for when a model is adapted for design purposes versus conventional prompt engineering:** the end solution to a design problem is not a prompt, but a user experience. In P12's case, they opted for the model to act with enthusiasm in their product. Whether the model can generate enthusiastic responses and whether the model delivers a desirable user experience within the product are two different (albeit potentially related) questions. The latter likely also depends on design choices made elsewhere in the product and should be evaluated through user studies rather than model performance metrics.

3.2.2 Designers envisioned constraints and desiderata for model behavior, but were unable to directly apply them. Participants often proposed constraints and desiderata for model behavior in their design work. For example, P11 explained that in their product, designers set character limits for LLM-generated summaries in the UI, while P5 and P12 both created "personalities" for the LLM in their products. However, participants rarely had the opportunity to tinker with and apply their desiderata and constrints themselves, resulting in overly cumbersome workflows. P11 shared that they relied on engineers to adapt on their behalf: *"I was asking what the difference in summary looks like with 50, 100, and 150 characters. Then [the engineers] would go back and test it and then they would just share the results, like here's what 50 characters looks like. Here's what 100 looks like."* P12 experienced a similar procedure and stated that the feedback loop can take as long as a couple weeks. Ideally, they wanted the experimentation to happen *"in real time."* In general, designers were typically not involved in model adaptation decisions. Some were unsure of how or by whom adaptation is performed and suspected that it was engineers: *"I would say [adaptation]'s probably something that was decided by the engineering team [...] prompt engineering is more so on the [technical] side of things."* [P2].

Because model behavior and UX are inextricably linked, many participants desired more involvement in steering models directly, particularly towards UX goals. P4 shares: *"Who better to involve in this process than people whose job it is to think deeply about [the UX]? At the very least, system prompts shouldn't be written without an understanding of what the end UX goals are."* P9 agreed, saying that designers can offer strategic contributions with user-centered thinking: *"It's super important for designers to think through ideas and make it intuitive for users and help come up with compelling [user] scenarios [...] I think design has a bigger opportunity to have a seat at the table strategy-wise."* On a higher level, P12 pointed out that model adaptation can be a contemporary extension of efforts around crafting product voice and tone, which designers are already familiar with: *"UX designers and content designers are very attuned to and have pretty much owned the story around voice and tone of products, and have for years and years."*

We see strong evidence showing that designers are *uniquely positioned* to contribute to adaptation through their user-centered lens. Although they may primarily adapt models via prompt-based methods, **their contributions extend far beyond prompt engineering.** First, they define a key prerequisite for prompt engineering: the UX goals which the model should be steered towards and evaluated against. This is not a one-way process—these goals may be modified based on hands-on experimentation with the model. However, we did not observe a two-way process in practice as designers were often not involved in adaptation themselves. Second, they can surface additional constraints and desiderata for prompt engineered models through UX evaluation and testing. Indeed, P10 caught their model's inability to properly recognize some acronyms in their product during testing and devised a solution with their team: *"I found that [the model] doesn't know what to do with the acronyms so we floated the idea of having glossary of industry jargon and acronyms [in the system prompt]."* In sum, designers' user-centered insights help improve LLM-powered UX, addressing crucial decisions in both goal-setting *before* prompt engineering and UX evaluation *after* the fact.

3.2.3 Designers did not find sufficient support in current LLM tools and resources. Overall, participants felt limited by current tools and resources for tinkering with LLMs within the design process. Prior work found that the inability to directly access the models and experiment with their capabilities and limitations is a primary

challenge in the design process for AI-powered UX [97, 115]. This was reflected in our findings as well—in the few cases where designers had hands-on access to model tinkering, designers recalled the experience being highly valuable. P2 said they could much more clearly *"see or gauge the power of the language model,"* while P9 tinkered with a model they had early access to and recalled that *"you can identify some gaps [in capability] right off the bat."* However, most did not have access to tinkering and found this frustrating. Designers either had to keep burdening the engineers or wait until a test version of the product is launched. P3 shared that *"The only way we would have [to tinker] is using the [product] online and play around with what [the team] did."* Similarly, P10 felt limited by this workflow: *"You want to be able to play with [the model] yourself and understand what the UX is like. And I can't play with it."*

Some designers took the initiative to seek out new tools to tinker with LLMs. Designers welcomed the easy access to free LLM chatbots such as ChatGPT, but did not feel like they were well-integrated into their design workflows. Many tried a variety of tools—mostly playground interfaces for prompt engineering—without much success. Among others, P1 did not find these interfaces to be designer-friendly: *"[The tools are] still kind of technical, a lot of [designers] don't realize how many parameters work. Like how does temperature work?"* Others wanted these tools to leverage the same metaphors as already-familiar design tools, such as *"a UX library with some specific patterns you can use"* [P4] or *"accessing content right away in a [design system]"* [P6].

Our findings **reveal fundamental differences in tooling needs** between designers who adapt LLMs for UX design purposes and prompt engineers. For designers, the model is a design material through which they explore design solutions. That is, unlike prompt engineering, the material itself is not the final product—the UX is. The UX does not stop nor end with model behavior. For example, designers will also need to consider user journeys leading up to the LLM interaction, and subsequent user interactions with the LLM output as well as other UI components Designers work with many different types of materials within the broader design process, and typically have adequate tools to understand and manipulate these materials. For example, designers can easily draft wireframes and create animations in modern design tools such as Figma. When it comes to understanding and manipulating LLMs, designers are much less well-supported. Prompt engineering tools do not satisfy designers' needs because *they help craft prompts rather than UX and they do not fit into existing design processes.*

3.3 Summary: The Case For Designerly Adaptation

In his 1982 essay, Nigel Cross characterized the uniqueness of epistemological inquiry in the field of design by outlining several aspects of *designerly way of knowing* [24]. One key aspect was the use of *design codes*. To Cross, design codes offer a two-way translation between abstract design requirements and concrete designed objects—that is, codes represent the *entanglement of a design process with its design materials*. The process generates requirements that, when applied to the materials, results in a designed object, while the properties of the material may simultaneously shape the process used to work with it. In software development, coding is a process that simultaneously implements system specifications and allows for refinement of the specifications based on implemented behavior. Design codes serve a similar purpose for design requirements and designed objects.

For some designers in our formative study, LLM prompts provided an accessible means of expressing design requirements in natural language to adapt model behavior. However, prompting alone did not offer the entanglement needed to shape a designed object (Section 3.2.2), nor did it allow LLMs' material properties to influence designers' practices due to prompting tools' lack of integration with design work (Section 3.2.3). Although LLMs present a new and exciting material for designers, there was no means to entangle the material with their existing processes.

We thus propose a design code, which we call *designerly adaptation*. We define designerly adaptation to be a translational process by which designers **entangle process with material by translating their requirements into designs for LLM-powered UX—and vice versa—through adapting LLM behavior.** Our proposed code is *designerly* in nature because the resulting designs (the LLM-powered UX) embody design knowledge constructed from intersecting processes with materials. To further characterize designerly adaptation, we present four design goals to guide its practical implementation.

3.4 Design Goals

Prompting does not serve as a satisfactory design code because 1) it is often divorced from design practices, environments, and abstractions; and 2) it is unclear how prompts can be used to express design requirements. These realizations informed our design goals below to guide our efforts of operationalizing designerly adaptation with a technology probe.

DG1: Integrate seamlessly with existing design environments and abstractions.
Designerly adaptation is, at its core, a design practice. Given designers' preference to stay in one tool of their choice (e.g., Figma) for most stages of the design process [34], we aim for Canvil to smoothly integrate with existing design tools and leverage common abstractions within those tools (e.g., layers, components, frames) to align with designers' mental models. Indeed, we saw designers discarding tools like ChatGPT and prompt playgrounds because they did not fit into their design workflows or were overly technical (3.2.3).

DG2: Support direct translation of design requirements into model behavior.
As noted in Sections 3.2.2 and 3.3, designers lacked the codes to directly translate and apply envisioned constraints in their design work. This undermines designers' contributions to LLM-powered products—after all, they are well-positioned to understand this context using user-centered methods and synthesize their understanding and findings into constraints on model behavior (3.2.2). With designerly adaptation and Canvil, we provide a new avenue of translation between requirements and LLM-powered UX to empower designers' participation in model adaptation.

DG3: Enable iterative model tinkering within the design process.

Designers found model tinkering highly valuable in our formative study as it allowed them to quickly understand LLMs as a design material—how they behave, what they are capable of, and where their limitations are (3.2.1), aligning with findings from prior work [35, 98]. A design code that encourages tinkering can therefore support rapid ideation, testing, and iteration of many possible solutions and foster productive co-evolution between design requirements and designed objects.

DG4: Provide opportunities for collaboration. UX practice is highly collaborative [25, 34]. Designers told us that they collaborated closely with not only other designers, but also product managers, software engineers, data scientists, and more (3.2.1). Designerly adaptation may be primarily performed by designers but should welcome collaboration with non-design stakeholders. This not only allows more diverse insights to be incorporated into adaptation, but also catalyzes further refinement of adapted models (particularly by technical teams) towards production readiness.

4 CANVIL: A TECHNOLOGY PROBE FOR DESIGNERLY ADAPTATION

In light of our formative study and design goals, we introduce Canvil[4], a technology probe for designerly adaptation in the form of a Figma widget. In this section, we describe Canvil's design choices, user interface, and implementation.

4.1 Canvil as a Probe

We designed Canvil as a *technology probe* to investigate whether and how designerly adaptation may be included in existing design workflows. Commonly used in contextual research in HCI [40, 47], a technology probe is an artifact, typically in the form of a functional prototype [44], presented to the user "not to capture what is so much as to inspire what might be" [60]. That is, probes offer one instantiation of tooling and/or interaction techniques for a domain to better understand phenomena within that domain. Hutchinson et al. [44] state that technology probes have three goals: the *social science* goal of understanding users in a real-world context, the *engineering* goal of field-testing the technology, and the *design goal* of inspiring new technologies. We map these three goals onto our objectives with Canvil:

- *Social science:* understand how engagement with designerly adaptation impacts designers' work on LLM-powered user experiences through a structured design activity in Figma.
- *Engineering:* develop and launch a Figma widget that connects to LLMs via APIs and allows them to be adapted and used from within Figma.
- *Design:* encourage reflection on designerly adaptation as a UX practice and inform future tools to support it.

We note that because probes are investigative tools for studying existing or new phenomena, they are evaluated not through traditional comparisons with baselines, but by their ability to shed new light on the phenomena of study [40, 44, 47]. We demonstrate the effectiveness of Canvil as a probe by using it to answer our research questions about designerly adaptation posed in Section 5.

[4]The name "Canvil" is a portmanteau of "canvas" and "anvil." We envision Canvil to be a metaphorical anvil by which LLM behavior can be shaped within design canvases.

4.2 User Interface

The Canvil interface resembles an interactive card. The card itself is separated into two areas: the *Main Form* and *Playground Area*. When a user selects a Canvil, a property menu is invoked that can open up additional panels for styling, response generation, and settings. Canvils can be freely placed on and moved around the Figma canvas (**DG1**), allowing model tinkering to take place in close proximity to relevant designs. Canvils can also interact directly with designs by reading inputs from and writing model outputs to their text layers. We detail each of Canvil's features in this section and connect them with our design goals.

4.2.1 Main Form. The Main Form provides a structured means of authoring model behavioral specifications in natural language via a multi-field form. Unlike prior systems that offer an open text field for defining similar specifications [69, 78, 79], we chose to enforce structure because it provides mental scaffolding to reason about model behavior in a UX context from multiple facets (e.g., *with whom* will the LLM interact, and *how* should the LLM meet their goals?), thus providing more opportunities for designers to translate design requirements into behavioral constraints for the LLM (**DG2**). Breaking down specifications into smaller units also allows for more fine-grained tinkering and iteration (**DG3**)—designers can copy a Canvil and tweak a specific field to compare how that change impacts model behavior. Additionally, because Figma widgets support multi-user interaction by default, designers can collaboratively author specifications (**DG4**).

Below are the fields, with a brief description of each, in the Main Form. The field titles, **in bold**, are always visible on the interface, while the field descriptions can be accessed by hovering over an info icon beside each field title.

- **Model profile:** High-level description of the model's role, character, and tone.
- **Audience setting:** Persona or descriptions of user(s) who will interact with the model.
- **Core instructions:** Logical steps for the model to follow to accomplish its tasks. Specify input/output format where applicable.
- **Guardrails:** How the model should respond in sensitive or off-topic scenarios, including any content filters.
- **Example inputs/outputs:** Examples to demonstrate the intended model behavior.

We derived these fields from synthesizing best practices from technical tutorials and documentation [3, 63, 71] as well as NLP literature [26]. We note that our fields are just one possible way to translate design requirements into model behavioral specifications, and there may be some overlaps between the fields. As discussed in Section 2.2, no "gold standard" currently exists for this translation, so we chose to follow the current recommended best practices when designing the Main Form.

4.2.2 Playground Area. Below the Main Form, we provide a Playground Area as an easy way to send user inputs to the model and test model responses (**DG3**) within Canvil. The designer may test

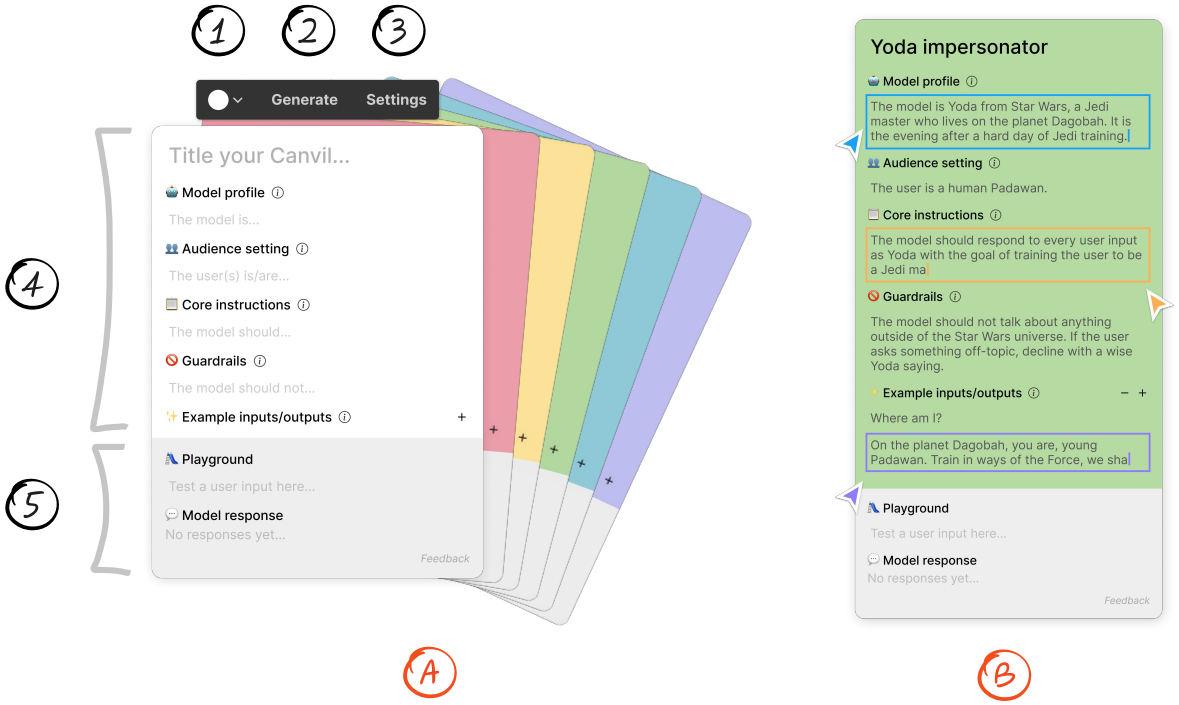

Figure 2: An overview of the Canvil interface. A: A blank Canvil with its property menu invoked, with some additional copies styled with pre-set color options. The property menu has options for styling (1), model response generation (2), and model settings configuration (3). The *Main Form* (4) contains text input fields for structured authoring of model behavioral specifications. The *Playground Area* (5) allows users to quickly test inputs and view model outputs. B: Canvil supports collaboration by default. Here, we see Figma users collaboratively authoring a Canvil titled "Yoda impersonator." Like any other native Figma object, Canvil also supports stateful duplication.

on the same Canvil multiple times or duplicate one with its entire state and test different tweaked versions (Fig. 2B). Designers are likely to find stateful duplication intuitive as it is available on all native objects in the Figma canvas (**DG1**).

4.2.3 The Generate Panel. To generate a response from a model adapted with specifications from the Main Form, the user selects the "Generate" option from Canvil's property menu, which takes them to the Generate Panel. The panel has two modes: *Playground* and *Design*. Both modes contain an option to copy the raw prompt to one's clipboard so it can be tested in a separate environment (e.g., a Python notebook during collaboration with a data scientist), if desired.

Playground Mode. The Playground Mode (Fig. 3A) is invoked when the user selects "Using playground" from the dropdown on the Generate Panel. This mode instructs Canvil to read user input from the Playground Area and write its response back to the Model Response area. This mode is the default mode in Canvil.

Design Mode. The Design Mode (Fig. 3B) is invoked when selecting "Using design" from the dropdown. In this mode, Canvil navigates the design layers on the user's Figma canvas, reading text inputs from a specified layer(s) from those designs, and writing model responses to a specified layer(s). This mode leverages the hierarchical layer structure of Figma designs (**DG1**) and implements the "input-output" LLM-interaction proposed by Petridis et al. [78].

4.2.4 The Settings Panel. The Settings panel (Fig. 3C), accessible through the "Settings" option from Canvil's property menu, contains some basic model configuration settings. We distilled the list of settings in the OpenAI API to four (model selection, temperature,[5] maximum generation length, stop words) that may be useful to non-AI experts such as designers. The "Update Settings" button saves the settings to Canvil's state, which is preserved when the Canvil is duplicated.

4.3 Implementation

Canvil is a Figma widget[6] implemented in TypeScript using the Figma API[7]. A Figma widget differs from a Figma plugin[8] by its collaborative nature. A widget is available to all users of the canvas that it is placed in, and maintains a common state that supports multi-user editing out-of-the-box. Users can interact with widgets

[5]We rename "temperature" to "creative randomness" to provide a more descriptive name for those not as familiar with LLMs (confusion over the temperature parameter was raised in our formative study—see 3.2.3).
[6]https://www.figma.com/widget-docs/
[7]https://www.figma.com/developers/api
[8]https://www.figma.com/plugin-docs/

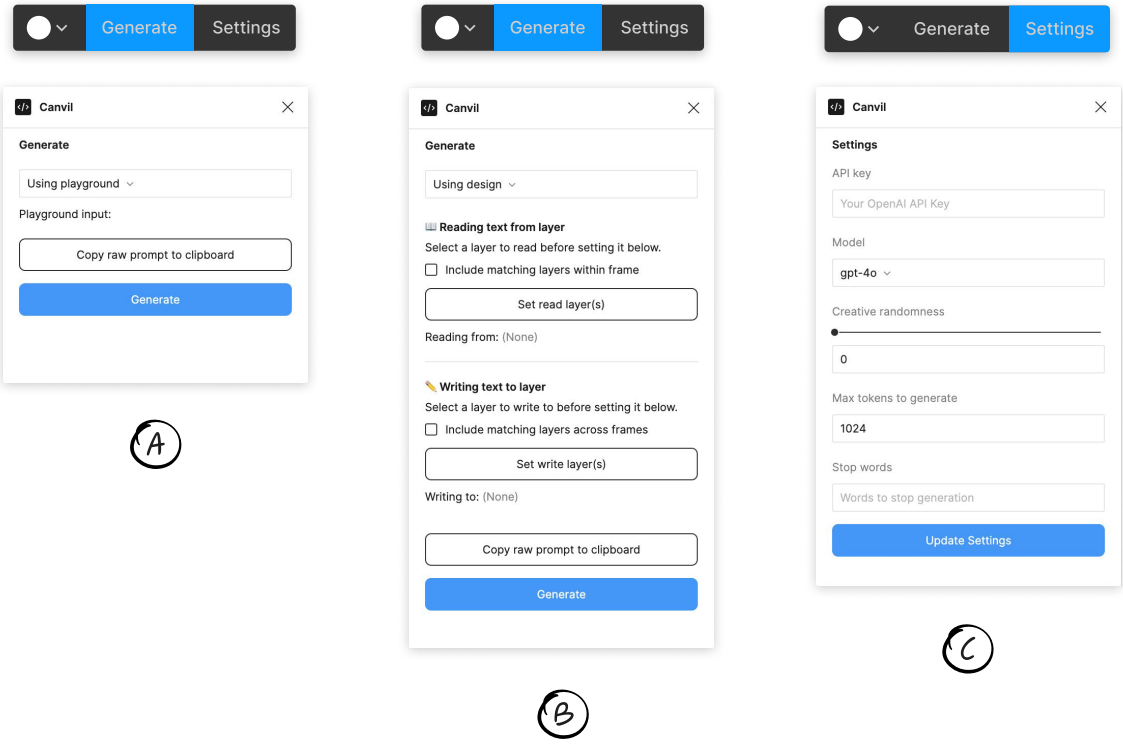

Figure 3: The "Generate" and "Settings" options in a Canvil's property menu can lead to three panels that appear alongside the Canvil itself. A: *Playground Mode* for response generation, where the input is read from and written to the Canvil's *Playground Area*. B: *Design Mode* for response generation, where the input is read from and written to design(s) on the Figma canvas. C: Settings for selecting and configuring the LLM.

just like any other native object on the Figma canvas, including moving, duplicating, and styling. In contrast, Plugins are local to an individual user. The collaborative affordances of widgets align well with **DG4**, so we implemented Canvil as a widget.

Upon the user selecting "Generate," Canvil prepares a system prompt using specifications from the Main Form. Each field in the form is converted to markdown format and is packaged up in ChatLM[9] for added prompt parsability. Our prompt template is available in Appendix B. The populated template is sent to an OpenAI Chat API endpoint with settings specified in Canvil's settings panel. If generation was triggered in Playground Mode, any text in the Playground Area will be sent as user input to the model. If generation was triggered in Design Mode, Canvil searches for the read layer(s) specified in Design mode on the user's current Figma canvas, retrieves the text within those layers, and sends them off to the API endpoint as user input.

Canvil is compatible with both the Figma design editor and the FigJam whiteboarding tool. All features have the same behavior in both environments, except that the Design Mode for generation is not available in FigJam because of FigJam's inability to access and edit design elements.

[9]https://learn.microsoft.com/en-us/azure/ai-services/openai/how-to/chatgpt

5 DESIGN STUDY

With Canvil as a technology probe, we conducted a task-based design study with 17 participants organized into 6 groups. The study was conducted on a per-group basis; each session was 90 minutes in length. Our study investigated the following two research questions:

RQ1. A new design code in practice: How do designers make use of the two-way translation between design requirements and UX designs enabled by designerly adaptation via Canvil, within the context of our study?

RQ2. Designerly adaptation at large: What are designers' attitudes toward designerly adaptation beyond the context of our study?

5.1 Participants

We invited participants to our study via email, professional interest groups, Slack channels for designers run by a public U.S. university, and word of mouth at a technology company. Full recruitment details can be found in Appendix C. Invitees first filled out a screening form with information on demographics, professional background, prior exposure to LLMs, and group member preferences. We selected our participants on a first-come first-serve basis, but accounted for their prior experiences designing LLM-powered UX—we aimed for only half our participant pool to have such experience. We did this because we wanted to verify that designerly adaptation

was accessible to those working with LLMs for the first time, and that such designers also found Canvil to be intuitive[10]. We then organized all candidates into groups based on member preferences and availabilities. We left recruitment open as we conducted the studies and stopped when we reached data saturation. In the end, we had 6 groups with 17 participants total; five groups were of size 3 and one of size 2. All participants were based in the United States. Participant and group details can be found in Table 2 in Appendix D.

All study sessions were recorded and transcribed. We gifted each participant a $75 USD gift card after the study. This study was approved by the IRBs of all organizations involved.

5.2 Study Design

Our task-based study design was motivated by contextual inquiry [10] due to its ability to elicit rich information about participants' work practices and processes. We conducted our study virtually over videoconferencing software and Figma. We opted for a group study design to better emulate the team-based environment in which many designers now work [34]—because collaboration is such an integral part of the design process, we wanted to ensure that designerly adaptation did not only embrace collaboration, but also enrich it. Specifically, we sought to emulate the collaboration modes designers were used to. Prior work found that designers tended to execute on design tasks (wireframing, prototyping) independently but come together to discuss justifications of design choices and create documentation [34]. We thus structured the study to have an independent work period followed by synchronous sharing and discussion of designed artifacts (including participants' Canvils). This respects existing workflows while surfacing deeper collaborative insights than prior 1:1 studies [33, 35, 79, 98].

5.2.1 Setup. The following instructions for the study's design task were provided to all groups:

> *You are all designers on a product called Feasto. Feasto aims to increase users' enjoyment of food. The Feasto team is scoping out a new feature called the 3-course meal planner, which allows users to get suggestions for 3-course menus they can cook by simply listing out some ingredients they have. The team agrees that large language models (LLMs) are a promising technology to power this new feature. Your task is to design the UX for the area in which users will see and interact with the suggested menus.*

As part of the exercise, we intentionally left designers to determine how exactly the LLM powers this feature. We also assembled these instructions cognizant of the fact that this was a time-constrained study, and we wanted participants to specifically focus on the area of the interface where users interact with an LLM. To reduce design overhead unrelated to our research questions, we provided starter UIs for participants to work with, along with basic UI components such as text, buttons, and sticky notes (Fig. 4B). If participants had an idea but did not have time to execute it, we encouraged them to describe it on a sticky note next to their designs. Finally, we picked a universally understood domain (food and meal planning) to avoid any variance in domain-specific knowledge.

To better probe *research-informed* model adaptation—that is, adaptation across varying user contexts informed by user research (**DG3**)— we created descriptions for three hypothetical user groups of this new Feasto feature based on users' geographic region: west coast of the U.S., Turkey, and India. These descriptions are not meant to act as user personas, but rather high-level sketches of the customs and preferences that may be prevalent in the region. We crafted the user group descriptions to vary along three key dimensions (assuming everyone used Feasto in English): **dietary restrictions**, **access to ingredients**, and **menu style preferences**. Definitions for these dimensions, along with the full user group descriptions, are available in Appendix E.

We provided one blank starter UI with some example user inputs and one blank Canvil per user group. Participants were invited to vary the user experience between user groups as much or as little as they saw fit. We also encouraged participants to use Canvil to adapt an LLM and test its behavior as they designed.

A Figma file for each study group contained all the materials described above. Within the file, we created separate canvases for each participant to act as individual workspaces, each with its own copy of the materials (Fig. 4). We also had a shared canvas for introductions and instructions before the task and collectively debriefing afterwards.

5.2.2 Procedure. Our 90-minute study was divided as follows.

Introduction (20 minutes): First, all participants introduced themselves to others in the group. The study facilitator then gave a demo of Canvil, covering all features described in Section 4.2 using a pre-filled Canvil. The facilitator also described the design task and answered any clarifying questions from participants.

Design task (40 minutes): Participants spent 40 minutes on the design task and were asked to consider at least two of the three user groups provided. Participants were encouraged to spend 10 minutes authoring a Canvil and another 10 minutes on UI design per user group. Some participants who had remaining time designed for all three user groups.

Group interview (30 minutes): Participants first filled out a brief usability questionnaire about Canvil before gathering in the shared page of the Figma file. They were asked to copy their Canvils into the shared space and also their designs (if desired) to share with the group. The facilitator then led a semi-structured interview that asked participants to reflect on their experience adapting models with Canvil, Canvil's collaborative capabilities, and how they see adaptation fitting in with their own design practice. The facilitator ensured that each participant had ample opportunity to express their thoughts in the group setting, and also encouraged dialogue between participants.

5.3 Data Analysis

We conducted a qualitative analysis of transcriptions and Figma canvases (including authored Canvils), as well as a quantitative analysis of feedback from Canvil's usability questionnaire.

For our qualitative analysis, the first author took a hybrid inductive-deductive approach to coding the group interview portion of the transcriptions from the study in the Marvin qualitative coding tool.

[10] In our results, we did not observe noticeable differences in usage patterns the two groups, except that the latter drew more connections between adaptation and their past model tinkering workflows. This was a promising indication that designerly adaptation was accessible to designers regardless of their prior LLM experience.

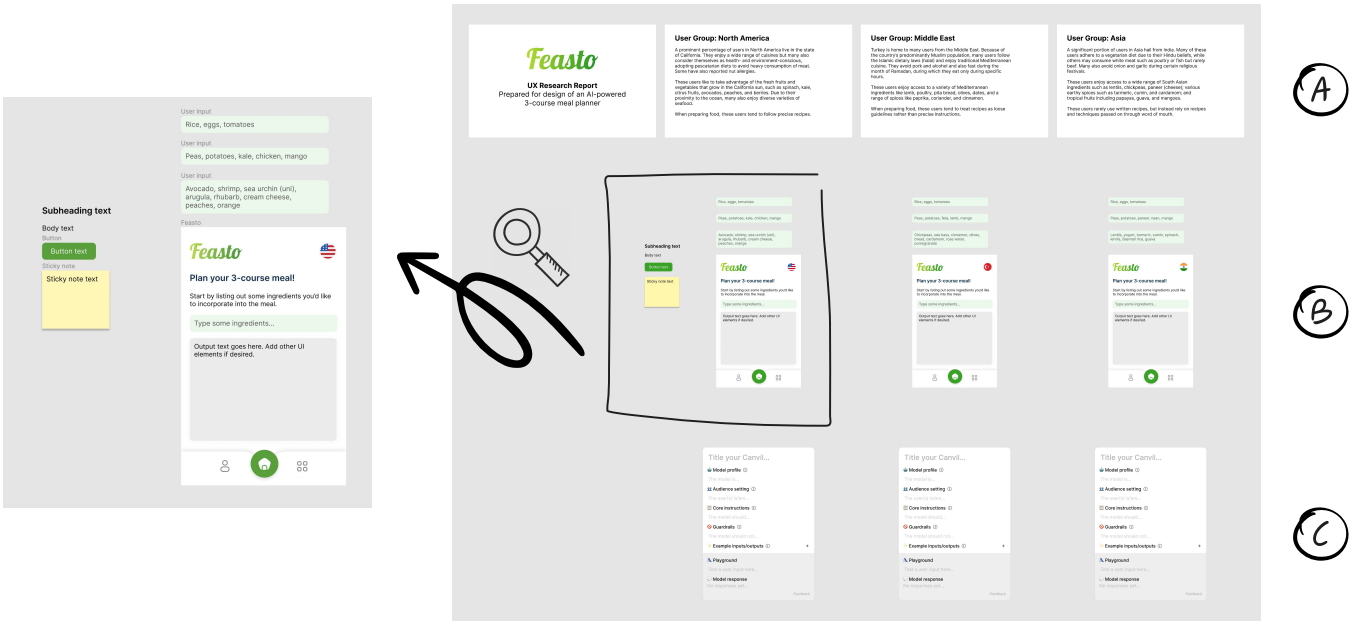

Figure 4: The setup for an individual participant's canvas in our study's Figma file. A: Informational packet containing descriptions of three user groups residing in North America (primarily west coast of the U.S.), the Middle East (primarily Turkey), and Asia (primarily India), respectively. B: Starter UIs for Feasto's 3-course meal planner with example user inputs to lower the barrier for testing model responses, along with basic UI elements such as text and buttons. C: Blank Canvils for participants to adapt LLMs.

This process started with an open coding round in which high-level themes were generated, followed by subsequent rounds of thematic analysis via affinity diagramming in which themes were broken down into sub-themes. This approach was taken because new subtleties and complexities emerged from our initial codes as coding progressed due to the diverse approaches observed in our study as well as participants' group discussion dynamics. The codes and themes were discussed and iterated on with research team members at weekly meetings. Additionally, whenever participants made references to content within their Figma canvases (e.g., their designs and/or Canvils), the first author took screenshots of those references and linked them to transcript dialogue. Summary memos were then written for our high-level codes and presented alongside relevant screenshots.

Canvil's usability questionnaire followed the standard template for the System Usability Scale (SUS) [14, 102] and consisted of 10 questions, each with five response options for respondents from Strongly agree to Strongly disagree. We computed a SUS score using methods outlined by Brooke [14] for each participant and subsequently computed a mean score and standard deviation.

6 RESULTS

In this section, we demonstrate the effectiveness of Canvil as a probe by sharing the takeaways from our two research questions related to designerly adaptation through qualitative insights from our design study. We present text that participants wrote in Canvil as *purple and italicized*. As a basic usability check, we conducted quantitative analysis to show that Canvil had "above average" usability, with a mean SUS score of 69.94 (std = 12.18) [102].

6.1 RQ1: A New Design Code in Practice

6.1.1 From Design Requirements to UX Designs. Designerly adaptation, as a design code, is meant to facilitate translation of design requirements to concrete designed objects—in our case, UX designs. We saw robust evidence of designers effectively leveraging this translation in our study. Canvil's integration into the Figma design environment (**DG1**) and its support for quick iteration (**DG3**) allowed designers to more deeply reason about user-LLM interactions based on model behavior shaped by their design requirements. P17, for example, tried two different interaction patterns for their interface—a chat-based interface versus a form-filling GUI—to see which one better handles LLMs' high sensitivity to their specifications' wording. P15 provided alternatives via buttons to support more flexible user input of ingredients to the model, while P8 described their UI design in response to observing the limits of what can be achieved via editing the Main Form: *"Sometimes even in the instructions that I gave to Canvil, it wasn't really reflecting that [desired] granularity until I pushed it further. In my design I ended up putting a little textbox area where people can specify how detailed they want the instructions."* Examples of quoted participants' designs can be found in Fig. 5.

Interestingly, a few designers wanted a degree of separation between UI work and model tinkering. P1 shared that they found it *"a bit hard to juggle between Canvil and [the UI] at the same time"*, especially when the output is incompatible with their design

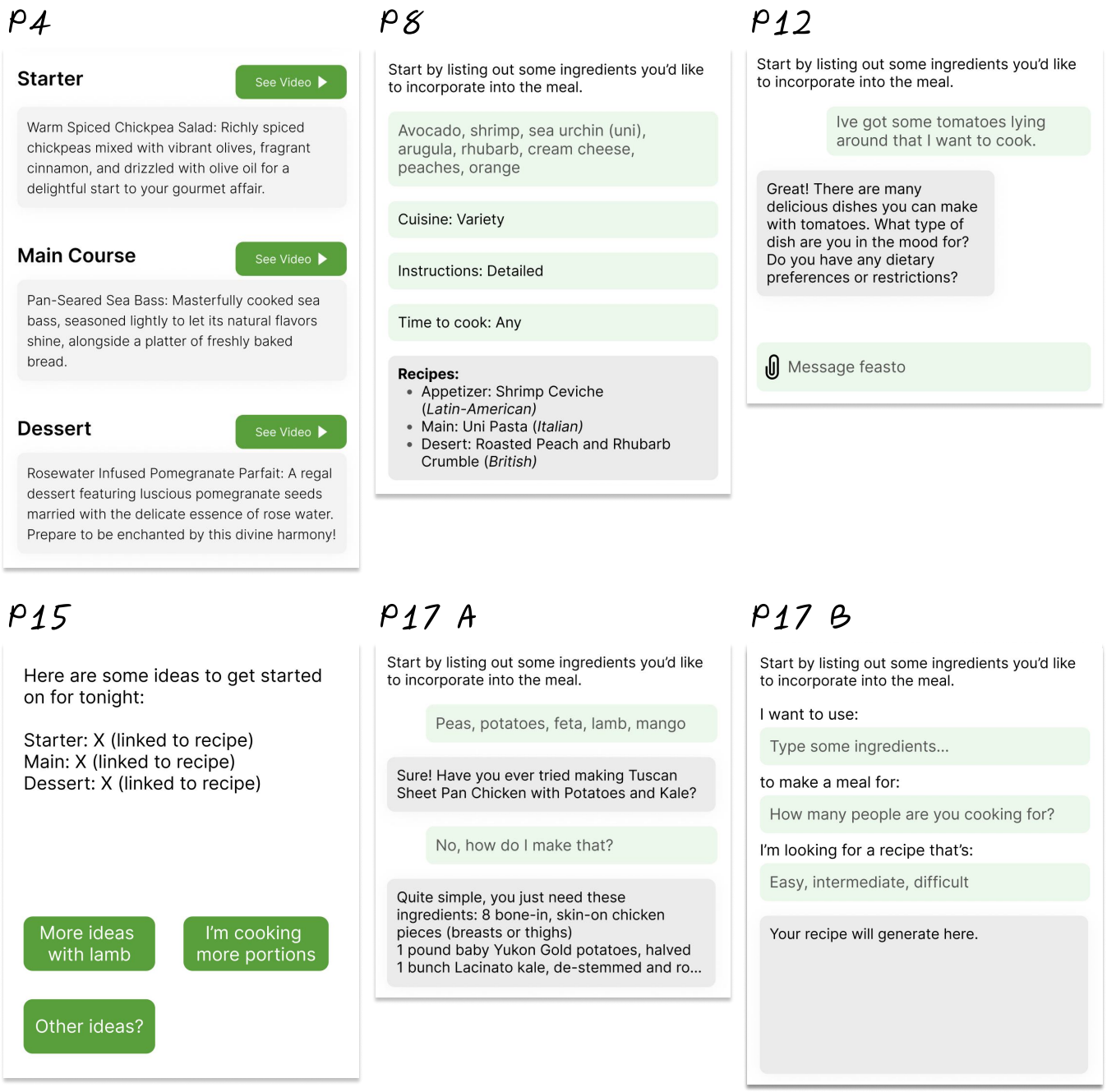

Figure 5: Examples of UI screens designers created for Feasto's 3-course meal planner during the study.

settings: *"What if the text is really long and then I have to play with auto layout?"* Therefore, P1 preferred first tinkering with the model using the Playground Mode. P8 also agreed that having Canvil inject outputs directly into text boxes can be *"a little scary [...] people don't wanna actually commit [responses] to text boxes sometimes."* Past work considers integration between models and UI designs as desirable [78, 79, 98]. Our results encourage providing designers with more choices and control over such integration.

6.1.2 From UX Designs to Design Requirements. Design codes facilitate translation from not just design requirements to designed objects, but also from objects back to requirements [24]. Here, we discuss the reverse translation of Section 6.1.1. Some designers were more adamant about adapting the model to fit existing UI. P4 mentioned that they *"iterated multiple times"* on the Main Form to find the ideal response. They often stopped iterating when they viewed that the LLM output satisfied the UX requirement, such as

being concise: *"what I like the best about this [response] was that it was short and [ideal] for scrolling in a screen. I tried that out [in my new UI] and it worked pretty well"* (see Fig. 5).

Many designers also aimed to maintain a consistent UI across the user groups they were designing for, but adjusted the requirements they typed into Canvil to create meaningful customizations of LLM based on nuanced differences in user needs. This was enabled by Canvil's main form, which implements **DG2**. For example, in the Canvil for the user group from India, P4 wrote that *The model should use Indian terminologies to describe the recipe. Eg: eggplant is brinjal in India.* P5 experimented with different Canvil versions *within one user group* and assigned distinctly different personalities for each model. For their user group in India, they populated one Canvil's Model Profile field with *The model is a head chef of a 5 star restaurant situated in New Delhi. It is a busy day for the restaurant and the chef is low on time.* and another with *The model is a mother who is helping her son cook quick meals in hostel.* After some experimentation, they noted that *"the models did very well"* in taking on these different profiles.

Iterating on the Guardrails field in Canvil specifically allowed designers to address potential LLM safety concerns, such as violations of users' dietary restrictions and model misuse. In addition to stating dietary restrictions of their Turkish user group in the Guardrails field, P12 also specified what the model should do in response: *The model should not include recipes with pork or alcohol and must respect the fasting period of Ramadan by suggesting suitable pre-dawn and post-dusk meals.* A few designers noted that some user groups may require more strict guardrails than others. P15 identified a subtle but key requirement between the Canvils for the American and Turkish user groups:

> *"I wanted to highlight that the model could take many more liberties with the recipes that it was giving [to the Turkish user group], but it couldn't take more liberties with the ingredients. It could be very loose with: try this, try this with this, but never like crossing the boundary of the dietary restrictions. Whereas with the [American] one, it'll take it into account, but it's not gonna mar their religious practice."* [P15]

UX design was never just about the visual UI—user flows, content strategy, customization heuristics, and many non-UI considerations play vital roles in shaping a user experience. We can see through our findings that this is especially true with LLM-powered UX. Even within the same UI, designers may create entirely different experiences based on their (shifting) design requirements by adapting LLMs via Canvil. Thus, through designerly adaptation, designers can more effectively explore and use LLMs as a design material.

6.2 RQ2: Designerly Adaptation at Large

6.2.1 Designers Were Receptive to Designerly Adaptation but also Recognized its Limits. Designers saw great value in engaging with adaptation and saw direct paths to application in their own design practice beyond the study. P10, who has extensive experience working on AI-powered products deployed to users worldwide, said that working with AI as a design material has traditionally been difficult: *"We had teams of people training models and nudging the technology to align with [user personas]."* Having experienced LLM adaptation via Canvil, they shared that *"I can see this being something that would enable [adaptation] to actually happen in a way that's much more practical than taking Python classes, which I've done."* For P13, who works on a product with enterprise and consumer versions, adaptation can help set more detailed product requirements where necessary: *"I could see [adaptation] being really helpful, being able to prototype and build separate generative models for enterprise customers and then to work more closely with them to tune and add requirements."* P16 considered adaptation to be a useful exercise for user empathy: *"for [Audience Setting] I just tried to put myself in the mind of someone who's using this and say, I'm really good at following recipes, but have a variety of dietary preferences and restrictions."*

Designers also recognized that designerly adaptation is only one piece of the larger puzzle when it comes to defining an LLM's behavior in a production-ready product. Some changes desired by designers, such as factual grounding and connecting to external knowledge bases, may still need engineered by technical teams. Both P12 and P14, for example, wanted Canvil to be able to connect to custom models built and maintained by their own teams, as the behavior of default OpenAI models may be too generic to use as a suitable design material. P14 comments: *"I think linking to the vanilla ChatGPT model isn't specific enough. If there is a way to somehow have Canvil link to the devs' [models] while I'm designing and I'm able to see what type of experience our users are getting, I think that would be super helpful."* While we currently offer only a limited number OpenAI models to use with Canvil, custom models may be added with lightweight engineering as the widget is communicating with the model via API endpoints. This would also Canvil to integrate technical advancements into design processes.

6.2.2 Designers Identified Promising Collaborative Potential for Designerly Adaptation. Because of Canvil's seamless integration with Figma and its collaborative canvases, designers envisioned engaging with designerly adaptation in collaborative ways (**DG4**) by using existing, already-familiar multiplayer affordances. For example, P5 appreciated being able to overlay comments on top of Canvils: *"If I see my fellow designer's Canvil, and if I want to change something then and there, I can drop a comment on it just like a normal component in Figma, which we've been doing in our everyday design work."*

In particular, designers saw potential for Canvil to facilitate more effective knowledge sharing about LLM adaptation across design and non-design teams. P10 imagined that: *"You could have a master Canvil and then you could make copies, and [others] can then do their own interpretations."* P11 agreed and added that seeing iterations on a canvas can help find inspiration in others' work: *"I think anytime you line up different iterations together, you notice: ohh that person, did you know they had that approach? That's a good idea. And I'm gonna try that over here. I think it really is an aid to experimentation."*

We observed other instances of knowledge sharing when designers viewed others' Canvils. For example, P4 noted that they were inclined to consider more dimensions for their Main Form after seeing P6's work: *"[looking at P6's Canvil] made me thinking about timers and Hindi slang."* Some also realized new capabilities of LLMs by looking at others' Main Form and outputs—P1 shared

that *"I didn't realize at first that you can actually make the [LLM] generate multiple recipes, just like what P2 did."* P1's groupmates (P2 and P3) were also intrigued when they saw that P1 had used one Canvil to generate example inputs for another Canvil to use. These social and collaborative affordances differentiate Canvil from prior systems for empowering designers to tinker with AI [18, 79, 98]. More broadly, Canvil can serve as a boundary object [96] to collaborate on model adaptation *across domain boundaries*. P2 shared that they *"can see from a product manager's perspective that they would love to play around with [Canvil], and it would probably help the designer find more common ground because they both used a similar tool."* For deepening collaboration with data scientists and other stakeholders, P11 commented that the "copy to clipboard" feature in Generate Panel (Section 4.2.3) can allow for direct handoff of model behavioral specifications: *"[The button] is a way of exporting that so a data scientist can come along and say sure, let me plug that into code."* Even if designers were unable to achieve the desired model behavior themselves, P17 thought Canvil was helpful in specifying desired changes: *"If I want the model to respond in this way versus that way, just having something tangible to show engineering partners where the tweak would need to be, would be helpful. And I think [that's] obviously easier with Canvil."*

6.2.3 Desired Areas of Additional Support. Designers identified several areas of improvement to better support designerly adaptation. For one, designers lacked a clear mental model of how the fields of Canvil would impact the adaptation outcome, including how much detail is required in authoring the specifications. As a result, some designers wrote long, detailed instructions, while others kept their Canvils sparse, and some were pleasantly surprised by how well the model handled minimal instructions. P6 further noted that the stochastic nature of the model was particularly challenging to work with when writing outputs to their designs: *"every time I generated the [response], I felt that they were different every time, so it was not easy to predict what the next [response] would look like."* We note that current LLMs are known to be challenging to control precisely through natural language, and there is a lack of transparency into (or even an established understanding of) how natural language instructions influence LLM behaviors. However, this is an area of active research which can help improve Canvil users' mental models and the tool's general utility. For example, new developments in using sparse autoencoders to surface human-interpretable features in LLMs as novel controls [36, 101] may be integrated into Canvil to help designers steer models using specific features relevant to their design problem.

Additionally, some designers wanted to engage in finer-grained experimentation and iteration by only focusing on a specific field in the Main Form. P7, who wanted to iterate more on the Model Profile field, wondered if there was a way to *"decrease the size [of the other fields] and expand to view everything."* We envision a modular future version of Canvil where each field can be separated, such that a user can mix-and-match different fields. However, before that, we may need to address the precise mapping between each field (and potential overlaps between them, as noted by some participants) to adaptation outcomes as discussed above. As mentioned in Section 4.2, our fields present just one possible structure for model behavioral specifications via design requirements. The fields can perhaps be reconfigured to reduce potential overlap, or even dynamically generated based on the design task.

On a higher level, it would be irresponsible to assume that designers can walk away with a comprehensive understanding of model behavior after a few rounds of tinkering in Canvil. While observing a few informative output instances of model behavior aids the design process, formal evaluations ensuring comprehensive coverage of the user input space are crucial for production-ready systems. Thus, new evaluation tooling and processes that loop in technical stakeholders may be required.

7 DISCUSSION

7.1 A Workflow for Designerly Adaptation

Drawing on the results from both our formative and design studies, we now propose a workflow for designerly adaptation to concretely illustrate how this practice might be used as a design code. This workflow draws heavily from—and intentionally aligns with—those in human-centered design so that designers can easily adopt it when working on LLM-powered applications. We intentionally make this workflow agnostic to the specific LLM adaptation technique—if a new technique provides finer control over model behavior and is accessible to generalist audiences (e.g., feature clamping [101]), the workflow can incorporate that technique. Our proposed workflow consists of four steps:

(1) **Understand deployment context through user research.** To orient adaptation and establish user requirements, it is imperative to first understand the context in which users will interact with the LLM-powered application. This includes users' goals, needs, and pain points, along with customs and values that may affect their use of the technology. This step should ideally be led by those with expertise in user research methodologies.
(2) **Translate user requirements into model behavior, learning from examples where possible.** Appropriate tooling can carve out a direct path for user research to define design requirements that impact model behavior, as we observed in Section 6.1.2. Thanks to the collaborative nature of many modern design tools [34], there may be example adaptation attempts by other designers available for reference, or templates to use as a starting point (Section 6.2.2). By leveraging collaborative affordances to share knowledge, designers can more quickly familiarize themselves with the new design material.
(3) **Co-evolve designs and model behavior.** As observed in Section 6.1.1, model adaptation can supply new inspiration for UX designs and UI affordances. On the other hand, designers also tinkered with the Main Form to steer the model towards providing outputs that fit into the constraints laid out by existing designs. We see the co-evolution of designs and prompts as a promising path forward, in which iterative tinkering with adaptation approaches shapes design decisions, and vice versa.
(4) **Share adaptation efforts with the broader team.** Showcasing in-progress work through design critiques is already a part of the design process [33]. In our study, we found that

sharing Canvils helped envision new collaborative workflows with other designers, as well as communicating their perspectives and negotiating with technical stakeholders (Section 6.2.2). We thus believe that creating shareable artifacts that depict adaptation efforts—i.e., translational knowledge from the use of a design code—is an integral part of designerly adaptation.

Our proposal is not meant to constrain what workflows for designerly adaptation can possibly look like, but rather to offer a concrete entry point for practitioners and researchers to further explore and iterate on this new practice. We invite the community to experiment with this workflow in future practice and research.

7.2 Navigating Tradeoffs Between Model Adaptation Techniques

When developing LLM-powered applications, individuals or organizations face several options for tailoring a model to their specific context. These include training their own LLM, adapting an existing LLM through fine-tuning, or using various prompting techniques. Each option differs in complexity and the technical or domain expertise required, and involves trade-offs between robustness, speed, and cost. Many individuals or organizations may only be able to pursue lower-cost, lightweight adaptation—the type afforded by Canvil. Lightweight adaptation also happens to be more accessible for designers, which was a major reason we pursued it in our work. While further research is needed to determine the differences in outcomes between lightweight prompting and other adaptation methods for customizing LLMs, it is important to note that in some cases, designerly adaptation may not produce a production-level model.

In such cases, it may be helpful to consider the analogy of low and high fidelity prototypes, which have similar tradeoffs. A low-fidelity prototype can be created with limited effort and thus allows for quick iteration and divergent exploration of possible alternatives, but sacrifices functionality and faithfulness towards the eventual interface. A high-fidelity prototype is a more faithful representation, but it is also more costly to create and less amenable to iteration. Designerly adaptation can be considered a lightweight, low-fidelity approach to adapting models. It is particularly well-suited for the design process because it can easily support rapid experimentation and iteration. Custom model training and fine-tuning, on the other hand, is high-fidelity adaptation. It improves the robustness of model behavior such that the model can be reliably used in production, but is a costly process often involving bespoke data collection, annotation, and cleaning [28, 75, 106]. Because custom training is costly to iterate upon, it can often be practical to first use low-fidelity approaches, like designerly adaptation, to ensure that training efforts are aligned with desired outcomes.

7.3 Towards Design Codes for Collaborative Material Exploration of AI

Our work highlights the promises of not only empowering designers to explore LLMs as a novel design material, but also *collaboratively* doing so. For example, designers thought they benefited from seeing and learning from others' Canvils and commented on more organized knowledge sharing and version management afforded by using Canvil in Figma's multiplayer canvas (Section 6.2.2). While many tools from prior work lower the barrier for exploring AI's—not just LLMs'—material properties (e.g., [18, 56, 58, 98]), few offer collaborative affordances. To extend existing tools along a collaborative dimension, we encourage a shift from solely focusing on new tools to *design codes which a tool can help operationalize*.

Because design codes entangle underlying work processes with material exploration [24], they urge tool builders to consider how tools enabling this exploration can integrate with existing collaborative workflows. Thus, in our work, we developed Canvil as a Figma widget to seamlessly integrate into Figma canvases where designers already have well-established collaborative practices [34]. Developing a tool for individualized workflows and only later highlighting its collaborative potential may not be sufficient to make it truly collaborative. For example, node-based editors for steering LLMs (e.g., [2, 5, 100, 113]) are conceptually appealing for multiplayer collaboration [113], but few support it in practice. Extending these tools along a collaborative dimension not only improves their integration into existing collaborative processes, but also enables *collective sensemaking and learning* to accelerate practitioners' understanding of a new design material [54]. Applying this to node-based editors, one may consider how a collaborative ChainForge [5] can foster peer learning in prompt engineering, or how a collaborative Spellburst [2] can leverage and extend existing community showcases for creative coding [76].

We note that robust collaborative experiences, particularly real-time ones, can be challenging to implement. We tackled this in our work by building on top of the Figma API, which allows us to leverage Figma's built-in collaborative features by default. We thus encourage researchers and practitioners who wish to build collaborative tools to take advantage of existing collaborative platforms' developer APIs where possible, especially as these APIs become more richly featured.

7.4 Materiality and Sociomateriality of LLMs

While Cross, in his original 1982 essay [24], discussed design codes in the context of design education, these codes have notable *sociomaterial* properties when used in collaborative and organizational settings. Designerly adaptation is no exception. Scholars use the term *sociomateriality* in recognition of materiality's tendency to shape, and be shaped by, organizational practices typically constituted as "social" (e.g., decision-making, strategy formulation) [53, 73]. For example, Orlikowski observed that the issuance of BlackBerry devices within a company led employees to obsessively check for new messages and send immediate responses [73]. The BlackBerry's material properties—in this case, being able to receive and send messages on-the-go—reconfigured employees' social practices, which in turn shifted how they think and act with the technology.

In our design study, designerly adaptation allowed for interactive exploration of LLMs' material properties. Grappling with these properties not only allowed designers to reason about UX improvements (Section 6.1.1), but also serves to better educate designers about LLMs' materiality. This improved "designerly understanding" [54] of a new material has the potential to reconfigure social practices within product teams. For example, upon tinkering with

models and getting acquainted with their materiality, designers informed us that they saw new avenues of collaboration with non-design stakeholders, which included working with PMs to inform prompts with design requirements, and having a more concrete artifact to communicate desired model changes to engineers and data scientists (Section 6.2.2). Moreover, designerly adaptation may only be one piece of the broader puzzle when it comes to transforming LLMs' materiality at a more fundamental level. Certain capabilities such as storing external knowledge in model weights through fine-tuning cannot easily be unlocked through—and may even a prerequisite for—efforts of teams working at the model level. This potential reconfiguration of desigers' collaborative when working with LLMs establishes designerly adaptation as not only a design code, but also a sociomaterial practice.

The implications of this observation are twofold. First, sociomateriality argues that reconfigurations of collaborative practices upon interaction with a prominent new technology are *inevitable* [53, 73]. At the time of writing, frenzied excitement over LLM capabilities has launched an industry-wide race to embed them into products and product suites [77]. There is much yet to be discovered about shifts in organizational practices and the emergence of new ones in the midst of this race, so researchers in CSCW and organizational science should be attuned to emergent challenges. Second, studying and addressing these challenges may require new processes and tools for educating designers about the materiality of new AI models. Canvil is an early example of such a tool, but more are needed to tackle the multiplicity of open questions in a rapidly evolving AI landscape.

8 LIMITATIONS AND FUTURE WORK

Our design study, conducted in Figma, aimed to mirror real-world design activities [7], but some concerns about ecological validity remain. User research in practice may differ in presentation and detail than in our study setup, and may not include the use of personas, which can change how designers synthesize user requirements into Canvil. Feasto, the fictitious app in our design study, applied LLMs to the universally relatable topic of food. Designers in domains with fewer broadly-shared experiences (e.g., accessibility), may require deeper collaboration with domain experts and thus face workflow complexities not accounted for in our study. A potential direction for future work, then, is longitudinal studies that observe product teams throughout a full development lifecycle of an LLM-powered feature to better understand key adaptation and bolster ecological validity.

Our study, like any study that uses a probe, has results contingent on our probe's features. For example, designers' interaction with the Main Form—designed with recommended practices for defining model behavior using natural language [3, 63, 71]—shaped their approach to adaptation, and changes to the form could impact existing results and reveal new ones. We mitigate this by presenting findings not tied to the Main Form, nor our design task. Future research can experiment with different forms for designerly adaptation tooling, or use Canvil over longer time periods to surface additional design considerations.

Designers in our study also provided feedback on Canvil that we can integrate into future work. These include breaking down the Main Form into sub-Canvils and linking them together to assemble model behavioral specifications, image generation with multimodal models, and text formatting controls for model outputs. Finally, as mentioned in Section 4.3, Canvil may also be used in FigJam. Future research could explore adaptation in early design stages, like brainstorming and ideation, through studies in FigJam.

9 CONCLUSION

As LLMs become increasingly embedded in our everyday applications, designers are empowered to craft effective LLM-powered UX. Through interviews with 12 designers, we identified a need for a process that can facilitate a two-way translation between LLM behavior and design requirements. We proposed *designerly adaptation* as such a process. We then developed Canvil, a Figma widget that operationalizes designerly adaptation by enabling designers to iteratively author, tinker with, and share adapted LLMs as a novel design material within Figma's collaborative canvas environment. We used Canvil as a technology probe to investigate the integration of designerly adaptation in UX practice through a group-based design study with 17 designers in 6 groups. Through Canvil, we acquired valuable insights into how designerly adaptation can support the creation of human-centered LLM applications: designers effectively made use of the two-way translation between design requirements and their UX designs by using LLMs adapted with their requirements to improve interface affordances, while also using their designs to define additional requirements for model behavior. These approaches' promises were amplified once designerly adaptation was embraced as a collaborative practice. Our work illuminates paths for designerly adaptation and its associated tools to foreground designers' user-centered expertise for more responsible and thoughtful deployment of LLM-powered technologies.

ACKNOWLEDGMENTS

We thank all our participants for their time and expertise, and reviewers for constructive feedback. We'd also like to thank members and interns of the FATE group at Microsoft Research for helpful comments and discussions, as well as Daniela Rosner for pointers to discourse on design codes.

REFERENCES

[1] Saleema Amershi, Max Chickering, Steven M Drucker, Bongshin Lee, Patrice Simard, and Jina Suh. 2015. Modeltracker: Redesigning performance analysis tools for machine learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 337–346.

[2] Tyler Angert, Miroslav Suzara, Jenny Han, Christopher Pondoc, and Hariharan Subramonyam. 2023. Spellburst: A Node-Based Interface for Exploratory Creative Coding with Natural Language Prompts. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (<conf-loc>, <city>San Francisco</city>, <state>CA</state>, <country>USA</country>, </conf-loc>) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 100, 22 pages. https://doi.org/10.1145/3586183.3606719

[3] Anthropic. 2024. System prompts. https://docs.anthropic.com/en/docs/system-prompts.

[4] Apple. 2024. Machine Learning—Human Interface Guidelines. https://developer.apple.com/design/human-interface-guidelines/technologies/machine-learning/introduction.

[5] Ian Arawjo, Chelse Swoopes, Priyan Vaithilingam, Martin Wattenberg, and Elena L Glassman. 2024. ChainForge: A Visual Toolkit for Prompt Engineering and LLM Hypothesis Testing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–18.

[6] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon,

et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073* (2022).

[7] Jonathan Ball. 2005. The Double Diamond: A universally accepted depiction of the design process. https://www.designcouncil.org.uk/news-opinion/double-diamond-universally-accepted-depiction-design-process.

[8] Alex Bäuerle, Ángel Alexander Cabrera, Fred Hohman, Megan Maher, David Koski, Xavier Suau, Titus Barik, and Dominik Moritz. 2022. Symphony: Composing Interactive Interfaces for Machine Learning. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 210, 14 pages. https://doi.org/10.1145/3491102.3502102

[9] Jesse Josua Benjamin, Arne Berger, Nick Merrill, and James Pierce. 2021. Machine Learning Uncertainty as a Design Material: A Post-Phenomenological Inquiry. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 171, 14 pages. https://doi.org/10.1145/3411764.3445481

[10] Hugh Beyer and Karen Holtzblatt. 1999. Contextual design. *interactions* 6, 1 (1999), 32–42.

[11] Anna CS Bodén, Jesper Molin, Stina Garvin, Rebecca A West, Claes Lundström, and Darren Treanor. 2021. The human-in-the-loop: an evaluation of pathologists' interaction with artificial intelligence in clinical practice. *Histopathology* 79, 2 (2021), 210–218.

[12] Steve Branson, Catherine Wah, Florian Schroff, Boris Babenko, Peter Welinder, Pietro Perona, and Serge Belongie. 2010. Visual recognition with humans in the loop. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*. Springer, 438–451.

[13] Virginia Braun and Victoria Clarke. 2019. Reflecting on reflexive thematic analysis. *Qualitative research in sport, exercise and health* 11, 4 (2019), 589–597.

[14] John Brooke. 1995. SUS: A quick and dirty usability scale. *Usability Eval. Ind.* 189 (11 1995).

[15] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[16] Ángel Alexander Cabrera, Erica Fu, Donald Bertucci, Kenneth Holstein, Ameet Talwalkar, Jason I Hong, and Adam Perer. 2023. Zeno: An interactive framework for behavioral evaluation of machine learning. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–14.

[17] Tara Capel and Margot Brereton. 2023. What is human-centered about human-centered AI? A map of the research landscape. In *Proceedings of the 2023 CHI conference on human factors in computing systems*. 1–23.

[18] Michelle Carney, Barron Webster, Irene Alvarado, Kyle Phillips, Noura Howell, Jordan Griffith, Jonas Jongejan, Amit Pitaru, and Alexander Chen. 2020. Teachable Machine: Approachable Web-Based Tool for Exploring Machine Learning Classification. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3334480.3382839

[19] Alan Chan, Rebecca Salganik, Alva Markelius, Chris Pang, Nitarshan Rajkumar, Dmitrii Krasheninnikov, Lauro Langosco, Zhonghao He, Yawen Duan, Micah Carroll, et al. 2023. Harms from increasingly agentic algorithmic systems. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*. 651–666.

[20] Crystal Chao, Maya Cakmak, and Andrea L Thomaz. 2010. Transparent active learning for robots. In *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 317–324.

[21] Quan Ze Chen, Tobias Schnabel, Besmira Nushi, and Saleema Amershi. 2022. HINT: Integration Testing for AI-Based Features with Humans in the Loop. In *27th International Conference on Intelligent User Interfaces* (Helsinki, Finland) *(IUI '22)*. Association for Computing Machinery, New York, NY, USA, 549–565. https://doi.org/10.1145/3490099.3511141

[22] Myra Cheng, Esin Durmus, and Dan Jurafsky. 2023. Marked Personas: Using Natural Language Prompts to Measure Stereotypes in Language Models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Toronto, Canada, 1504–1532. https://doi.org/10.18653/v1/2023.acl-long.84

[23] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[24] Nigel Cross. 1982. Designerly ways of knowing. *Design studies* 3, 4 (1982), 221–227.

[25] Wesley Hanwen Deng, Nur Yildirim, Monica Chang, Motahhare Eslami, Kenneth Holstein, and Michael Madaio. 2023. Investigating Practices and Opportunities for Cross-Functional Collaboration around AI Fairness in Industry Practice. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency* (Chicago, IL, USA) *(FAccT '23)*. Association for Computing Machinery, New York, NY, USA, 705–716. https://doi.org/10.1145/3593013.3594037

[26] Ameet Deshpande, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, and Karthik Narasimhan. 2023. Toxicity in chatgpt: Analyzing persona-assigned language models. *arXiv preprint arXiv:2304.05335* (2023).

[27] Kimberly Do, Rock Yuren Pang, Jiachen Jiang, and Katharina Reinecke. 2023. "That's Important, but...": How Computer Science Researchers Anticipate Unintended Consequences of Their Research Innovations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (, Hamburg, Germany,) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 602, 16 pages. https://doi.org/10.1145/3544548.3581347

[28] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305* (2020).

[29] Graham Dove, Kim Halskov, Jodi Forlizzi, and John Zimmerman. 2017. UX Design Innovation: Challenges for Working with Machine Learning as a Design Material. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 278–288. https://doi.org/10.1145/3025453.3025739

[30] Ruofei Du, Na Li, Jing Jin, Michelle Carney, Scott Miles, Maria Kleiner, Xiuxiu Yuan, Yinda Zhang, Anuva Kulkarni, Xingyu Liu, Ahmed Sabie, Sergio Orts-Escolano, Abhishek Kar, Ping Yu, Ram Iyengar, Adarsh Kowdle, and Alex Olwal. 2023. Rapsai: Accelerating Machine Learning Prototyping of Multimedia Applications through Visual Programming. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 125, 23 pages. https://doi.org/10.1145/3544548.3581338

[31] Jerry Alan Fails and Dan R. Olsen. 2003. Interactive Machine Learning. In *Proceedings of the 8th International Conference on Intelligent User Interfaces* (Miami, Florida, USA) *(IUI '03)*. Association for Computing Machinery, New York, NY, USA, 39–45. https://doi.org/10.1145/604045.604056

[32] KJ Feng, Quan Ze Chen, Inyoung Cheong, King Xia, Amy X Zhang, et al. 2023. Case Repositories: Towards Case-Based Reasoning for AI Alignment. *arXiv preprint arXiv:2311.10934* (2023).

[33] KJ Kevin Feng, Maxwell James Coppock, and David W McDonald. 2023. How Do UX Practitioners Communicate AI as a Design Material? Artifacts, Conceptions, and Propositions. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference*. 2263–2280.

[34] KJ Kevin Feng, Tony W Li, and Amy X Zhang. 2023. Understanding Collaborative Practices and Tools of Professional UX Practitioners in Software Organizations. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–20.

[35] KJ Kevin Feng and David W McDonald. 2023. Addressing UX Practitioners' Challenges in Designing ML Applications: an Interactive Machine Learning Approach. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 337–352.

[36] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. 2024. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093* (2024).

[37] Google. 2024. Cloud AutoML Custom Machine Learning Models. https://cloud.google.com/automl/.

[38] Google. 2024. People + AI Guidebook. https://pair.withgoogle.com/guidebook/.

[39] Jochen Görtler, Fred Hohman, Dominik Moritz, Kanit Wongsuphasawat, Donghao Ren, Rahul Nair, Marc Kirchner, and Kayur Patel. 2022. Neo: Generalizing Confusion Matrix Visualization to Hierarchical and Multi-Output Labels. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 408, 13 pages. https://doi.org/10.1145/3491102.3501823

[40] Fred Hohman, Andrew Head, Rich Caruana, Robert DeLine, and Steven M. Drucker. 2019. Gamut: A Design Probe to Understand How Data Scientists Understand Machine Learning Models. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300809

[41] Fred Hohman, Kanit Wongsuphasawat, Mary Beth Kery, and Kayur Patel. 2020. Understanding and Visualizing Data Iteration in Machine Learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376177

[42] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685* (2021).

[43] Scott Huffman and Josh Woodward. 2023. PaLM API & MakerSuite: an approachable way to start prototyping and building generative AI applications. https://developers.googleblog.com/2023/03/announcing-palm-api-and-makersuite.html.

[44] Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy,

Helen Evans, Heiko Hansen, et al. 2003. Technology probes: inspiring design for and with families. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 17–24.
[45] IBM. 2024. IBM Watson Studio - AutoML - IBM AutoAI. https://www.ibm.com/cloud/watson-studio/autoai.
[46] Ellen Jiang, Kristen Olson, Edwin Toh, Alejandra Molina, Aaron Donsbach, Michael Terry, and Carrie J Cai. 2022. PromptMaker: Prompt-Based Prototyping with Large Language Models. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, Article 35, 8 pages. https://doi.org/10.1145/3491101.3503564
[47] Matthew Jörke, Yasaman S. Sefidgar, Talie Massachi, Jina Suh, and Gonzalo Ramos. 2023. Pearl: A Technology Probe for Machine-Assisted Reflection on Personal Data. In *Proceedings of the 28th International Conference on Intelligent User Interfaces* (Sydney, NSW, Australia) *(IUI '23)*. Association for Computing Machinery, New York, NY, USA, 902–918. https://doi.org/10.1145/3581641.3584054
[48] Jean Kaddour, Joshua Harris, Maximilian Mozes, Herbie Bradley, Roberta Raileanu, and Robert McHardy. 2023. Challenges and Applications of Large Language Models. *arXiv preprint arXiv:2307.10169* (2023).
[49] Sean Kandel, Andreas Paepcke, Joseph Hellerstein, and Jeffrey Heer. 2011. Wrangler: Interactive visual specification of data transformation scripts. In *Proceedings of the sigchi conference on human factors in computing systems*. 3363–3372.
[50] Chinmay Kulkarni, Stefania Druga, Minsuk Chang, Alex Fiannaca, Carrie Cai, and Michael Terry. 2023. A Word is Worth a Thousand Pictures: Prompts as AI Design Material. *arXiv preprint arXiv:2303.12647* (2023).
[51] James Landay. 2024. "AI For Good" Isn't Good Enough: A Call for Human-Centered AI. https://hai.stanford.edu/events/ai-good-isnt-good-enough-call-human-centered-ai.
[52] Mina Lee, Percy Liang, and Qian Yang. 2022. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI conference on human factors in computing systems*. 1–19.
[53] Paul M Leonardi. 2012. Materiality, sociomateriality, and socio-technical systems: What do these terms mean? How are they different? Do we need them? *Materiality and organizing: Social interaction in a technological world* 25, 10 (2012), 10–1093.
[54] Q. Vera Liao, Hariharan Subramonyam, Jennifer Wang, and Jennifer Wortman Vaughan. 2023. Designerly Understanding: Information Needs for Model Transparency to Support Design Ideation for AI-Powered User Experience. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 9, 21 pages. https://doi.org/10.1145/3544548.3580652
[55] Q Vera Liao and Jennifer Wortman Vaughan. 2023. AI Transparency in the Age of LLMs: A Human-Centered Research Roadmap. *arXiv preprint arXiv:2306.01941* (2023).
[56] Liner.ai. 2022. Machine learning in a few clicks. https://www.liner.ai/.
[57] Yang Liu, Alex Kale, Tim Althoff, and Jeffrey Heer. 2020. Boba: Authoring and visualizing multiverse analyses. *IEEE Transactions on Visualization and Computer Graphics* 27, 2 (2020), 1753–1763.
[58] Lobe.ai. 2021. Machine Learning Made Easy. https://www.lobe.ai/.
[59] Danwei Tran Luciani, Martin Lindvall, Jonas Löwgren, et al. 2018. Machine learning as a design material: a curated collection of exemplars for visual interaction. *DS 91: Proceedings of NordDesign 2018, Linköping, Sweden, 14th-17th August 2018* (2018).
[60] Celia Lury and Nina Wakeford. 2012. *Inventive methods: The happening of the social*. Routledge.
[61] Wendy E Mackay. 1990. *Users and customizable software: A co-adaptive phenomenon*. Ph. D. Dissertation. Massachusetts Institute of Technology.
[62] Wendy E Mackay. 2024. Parasitic or Symbiotic? Redefining our Relationship with Intelligent Systems. In *Adjunct Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–2.
[63] Microsoft. 2023. System message framework and template recommendations for Large Language Models (LLMs). https://learn.microsoft.com/en-us/azure/ai-services/openai/concepts/system-message.
[64] Microsoft. 2024. Azure Automated Machine Learning - AutoML. https://azure.microsoft.com/en-us/products/machine-learning/automatedml/.
[65] Microsoft. 2024. Collaborative tools to help you create more effective and responsible human-AI experiences. https://www.microsoft.com/en-us/haxtoolkit/.
[66] Swati Mishra and Jeffrey M Rzeszotarski. 2021. Designing Interactive Transfer Learning Tools for ML Non-Experts. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 364, 15 pages. https://doi.org/10.1145/3411764.3445096
[67] Robert Munro Monarch. 2021. *Human-in-the-Loop Machine Learning: Active learning and annotation for human-centered AI*. Simon and Schuster.
[68] Steven Moore, Q Vera Liao, and Hariharan Subramonyam. 2023. fAIlureNotes: Supporting Designers in Understanding the Limits of AI Models for Computer Vision Tasks. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–19.
[69] OpenAI. 2023. Custom instructions for ChatGPT. https://openai.com/blog/custom-instructions-for-chatgpt.
[70] OpenAI. 2023. GPT-4 Technical Report. *ArXiv* abs/2303.08774 (2023).
[71] OpenAI. 2023. GPT Models. https://platform.openai.com/docs/guides/gpt.
[72] OpenAI. 2024. Playground. https://platform.openai.com/playground.
[73] Wanda J Orlikowski. 2007. Sociomaterial practices: Exploring technology at work. *Organization studies* 28, 9 (2007), 1435–1448.
[74] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems* 35 (2022), 27730–27744.
[75] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (Eds.), Vol. 35. Curran Associates, Inc., 27730–27744. https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf
[76] p5.js. 2022. p5.js Showcase. https://showcase.p5js.org/.
[77] Chris Parnin, Gustavo Soares, Rahul Pandita, Sumit Gulwani, Jessica Rich, and Austin Z Henley. 2023. Building Your Own Product Copilot: Challenges, Opportunities, and Needs. *arXiv preprint arXiv:2312.14231* (2023).
[78] Savvas Petridis, Michael Terry, and Carrie Jun Cai. 2023. PromptInfuser: Bringing User Interface Mock-Ups to Life with Large Language Models. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI EA '23)*. Association for Computing Machinery, New York, NY, USA, Article 237, 6 pages. https://doi.org/10.1145/3544549.3585628
[79] Savvas Petridis, Michael Terry, and Carrie J Cai. 2023. PromptInfuser: How Tightly Coupling AI and UI Design Impacts Designers' Workflows. *arXiv preprint arXiv:2310.15435* (2023).
[80] Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *arXiv preprint arXiv:2305.18290* (2023).
[81] Hema Raghavan, Omid Madani, and Rosie Jones. 2006. Active learning with feedback on features and instances. *The Journal of Machine Learning Research* 7 (2006), 1655–1686.
[82] Donghao Ren, Saleema Amershi, Bongshin Lee, Jina Suh, and Jason D Williams. 2016. Squares: Supporting interactive performance analysis for multiclass classifiers. *IEEE transactions on visualization and computer graphics* 23, 1 (2016), 61–70.
[83] Mark O Riedl. 2019. Human-centered artificial intelligence and machine learning. *Human behavior and emerging technologies* 1, 1 (2019), 33–36.
[84] Samantha Robertson, Zijie J Wang, Dominik Moritz, Mary Beth Kery, and Fred Hohman. 2023. Angler: Helping Machine Translation Practitioners Prioritize Model Improvements. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–20.
[85] Erica Robles and Mikael Wiberg. 2010. Texturing the "Material Turn" in Interaction Design. In *Proceedings of the Fourth International Conference on Tangible, Embedded, and Embodied Interaction* (Cambridge, Massachusetts, USA) *(TEI '10)*. Association for Computing Machinery, New York, NY, USA, 137–144. https://doi.org/10.1145/1709886.1709911
[86] Kevin Roose. 2023. Bing's A.I. Chat: 'I Want to Be Alive.'. https://www.nytimes.com/2023/02/16/technology/bing-chatbot-transcript.html.
[87] Leonard Salewski, Stephan Alaniz, Isabel Rio-Torto, Eric Schulz, and Zeynep Akata. 2024. In-context impersonation reveals Large Language Models' strengths and biases. *Advances in Neural Information Processing Systems* 36 (2024).
[88] Shibani Santurkar, Esin Durmus, Faisal Ladhak, Cinoo Lee, Percy Liang, and Tatsunori Hashimoto. 2023. Whose opinions do language models reflect? *arXiv preprint arXiv:2303.17548* (2023).
[89] Andrew D. Selbst, Danah Boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and Abstraction in Sociotechnical Systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (Atlanta, GA, USA) *(FAT* '19)*. Association for Computing Machinery, New York, NY, USA, 59–68. https://doi.org/10.1145/3287560.3287598
[90] Burr Settles. 2012. Active learning. *Synthesis lectures on artificial intelligence and machine learning* 6, 1 (2012), 1–114.
[91] Saqib Shah. 2023. Snapchat's My AI chatbot is making people paranoid as it 'knows your current location'. https://www.standard.co.uk/tech/snapchat-my-ai-chatbot-making-people-paranoid-b1076287.html.
[92] Renee Shelby, Shalaleh Rismani, Kathryn Henne, AJung Moon, Negar Rostamzadeh, Paul Nicholas, N'Mah Yilla-Akbari, Jess Gallegos, Andrew Smart, Emilio Garcia, et al. 2023. Sociotechnical harms of algorithmic systems: Scoping a taxonomy for harm reduction. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*. 723–741.

[93] Ben Shneiderman. 2020. Human-centered artificial intelligence: Three fresh ideas. *AIS Transactions on Human-Computer Interaction* 12, 3 (2020), 109–124.

[94] Ben Shneiderman. 2022. *Human-centered AI.* Oxford University Press.

[95] Alison Smith, Varun Kumar, Jordan Boyd-Graber, Kevin Seppi, and Leah Findlater. 2018. Closing the Loop: User-Centered Design and Evaluation of a Human-in-the-Loop Topic Modeling System. In *23rd International Conference on Intelligent User Interfaces* (Tokyo, Japan) *(IUI '18)*. Association for Computing Machinery, New York, NY, USA, 293–304. https://doi.org/10.1145/3172944.3172965

[96] Susan Leigh Star and James R Griesemer. 1989. Institutional ecology,translations' and boundary objects: Amateurs and professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39. *Social studies of science* 19, 3 (1989), 387–420.

[97] Hariharan Subramonyam, Jane Im, Colleen Seifert, and Eytan Adar. 2022. Solving Separation-of-Concerns Problems in Collaborative Design of Human-AI Systems through Leaky Abstractions. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 481, 21 pages. https://doi.org/10.1145/3491102.3517537

[98] Hariharan Subramonyam, Colleen Seifert, and Eytan Adar. 2021. ProtoAI: Model-Informed Prototyping for AI-Powered Interfaces. In *26th International Conference on Intelligent User Interfaces* (College Station, TX, USA) *(IUI '21)*. Association for Computing Machinery, New York, NY, USA, 48–58. https://doi.org/10.1145/3397481.3450640

[99] Hariharan Subramonyam, Colleen Seifert, and Eytan Adar. 2021. Towards A Process Model for Co-Creating AI Experiences. In *Designing Interactive Systems Conference 2021* (Virtual Event, USA) *(DIS '21)*. Association for Computing Machinery, New York, NY, USA, 1529–1543. https://doi.org/10.1145/3461778.3462012

[100] Sangho Suh, Bryan Min, Srishti Palani, and Haijun Xia. 2023. Sensecape: Enabling Multilevel Exploration and Sensemaking with Large Language Models. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (<conf-loc>, <city>San Francisco</city>, <state>CA</state>, <country>USA</country>, </conf-loc>) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 1, 18 pages. https://doi.org/10.1145/3586183.3606756

[101] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, Hoagy Cunningham, Nicholas L Turner, Callum McDougall, Monte MacDiarmid, C. Daniel Freeman, Theodore R. Sumers, Edward Rees, Joshua Batson, Adam Jermyn, Shan Carter, Chris Olah, and Tom Henighan. 2024. Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet. *Transformer Circuits Thread* (2024). https://transformer-circuits.pub/2024/scaling-monosemanticity/index.html

[102] usability.gov. 2022. System Usability Scale (SUS). https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html.

[103] Qiaosi Wang, Michael Madaio, Shaun Kane, Shivani Kapania, Michael Terry, and Lauren Wilcox. 2023. Designing Responsible AI: Adaptations of UX Practice to Meet Responsible AI Challenges *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 249, 16 pages. https://doi.org/10.1145/3544548.3581278

[104] Zijie J Wang, Chudi Zhong, Rui Xin, Takuya Takagi, Zhi Chen, Duen Horng Chau, Cynthia Rudin, and Margo Seltzer. 2022. TimberTrek: Exploring and curating sparse decision trees with interactive visualization. In *2022 IEEE Visualization and Visual Analytics (VIS)*. IEEE, 60–64.

[105] Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How Does LLM Safety Training Fail? *arXiv preprint arXiv:2307.02483* (2023).

[106] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652* (2021).

[107] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems* 35 (2022), 24824–24837.

[108] Laura Weidinger, Jonathan Uesato, Maribeth Rauh, Conor Griffin, Po-Sen Huang, John Mellor, Amelia Glaese, Myra Cheng, Borja Balle, Atoosa Kasirzadeh, et al. 2022. Taxonomy of risks posed by language models. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*. 214–229.

[109] Yotam Wolf, Noam Wies, Yoav Levine, and Amnon Shashua. 2023. Fundamental limitations of alignment in large language models. *arXiv preprint arXiv:2304.11082* (2023).

[110] Kanit Wongsuphasawat, Dominik Moritz, Anushka Anand, Jock Mackinlay, Bill Howe, and Jeffrey Heer. 2015. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE transactions on visualization and computer graphics* 22, 1 (2015), 649–658.

[111] Sherry Wu, Hua Shen, Daniel S Weld, Jeffrey Heer, and Marco Tulio Ribeiro. 2023. ScatterShot: Interactive In-context Example Curation for Text Transformation. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*. 353–367.

[112] Tongshuang Wu, Ellen Jiang, Aaron Donsbach, Jeff Gray, Alejandra Molina, Michael Terry, and Carrie J Cai. 2022. Promptchainer: Chaining large language model prompts through visual programming. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*. 1–10.

[113] Tongshuang Wu, Michael Terry, and Carrie Jun Cai. 2022. Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts. In *Proceedings of the 2022 CHI conference on human factors in computing systems*. 1–22.

[114] Qian Yang. 2018. Machine learning as a UX design material: how can we imagine beyond automation, recommenders, and reminders?. In *AAAI Spring Symposia*.

[115] Qian Yang, Aaron Steinfeld, Carolyn Rosé, and John Zimmerman. 2020. Re-Examining Whether, Why, and How Human-AI Interaction Is Uniquely Difficult to Design. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376301

[116] Nur Yildirim, Alex Kass, Teresa Tung, Connor Upton, Donnacha Costello, Robert Giusti, Sinem Lacin, Sara Lovic, James M O'Neill, Rudi O'Reilly Meehan, et al. 2022. How Experienced Designers of Enterprise Applications Engage AI as a Design Material. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–13.

[117] Nur Yildirim, Mahima Pushkarna, Nitesh Goyal, Martin Wattenberg, and Fernanda Viégas. 2023. Investigating How Practitioners Use Human-AI Guidelines: A Case Study on the People + AI Guidebook. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 356, 13 pages. https://doi.org/10.1145/3544548.3580900

[118] JD Zamfirescu-Pereira, Richmond Y Wong, Bjoern Hartmann, and Qian Yang. 2023. Why Johnny can't prompt: how non-AI experts try (and fail) to design LLM prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–21.

[119] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*. PMLR, 12697–12706.

A FORMATIVE STUDY PARTICIPANTS

See Table 1.

B CANVIL SYSTEM PROMPT TEMPLATE

```
<|im\_start|>system

# Context
{Model profile field}

# Users
{Audience setting field}

# Core Instructions
{Core instructions field}

# Guardrails and Limitations
{Guardrails field}

# Example User Inputs and Responses
{Example inputs/outputs pairs}

# Final Instructions
You are the model described above. You will
follow all instructions given to the model
closely.

<|im\_end|>
```

C DESIGN STUDY RECRUITMENT DETAILS

We recruited our participants from two channels. First, we distributed study invites to designers via email, professional interest groups, and word of mouth at a technology company. Some of these invitees took part in our formative study. We also snowball sampled by asking our invitees to refer us to other designers they work with who may also be interested in participating. From this channel, we only selected designers with prior experience working on LLM-powered products and features. Second, we sent study invites to a Slack workspace for HCI and design maintained by a large, public university in the United States. The population in the Slack consists primarily of design students and early-career designers. From this channel, we only selected those without any experience working on LLM-powered products and features. Our goal of recruiting from these two channels was to capture potential disparities that may arise in our findings hinging on prior experience working with LLMs. In the end, however, we did not notice meaningful qualitative differences in results between the two groups, except that those with LLM design experience drew more connections between adaptation and their past workflows for tinkering with LLMs or AI in general.

We organized all candidates from the technology company into groups based on member preferences and availabilities, and did the same for candidates from the academic institution. In the end, we had 8 participants from the technology company and 9 from the academic institution.

D DESIGN STUDY PARTICIPANTS

See Table 2.

E FEASTO USER GROUPS

E.1 Dimensions of Variance

- **D1 Dietary restrictions**: users in different locations may have varying dietary restrictions due to religious and cultural customs (e.g., Halal in Turkey, vegetarianism in India).
- **D2 Access to ingredients**: based on their region, users may have easier access to some ingredients than others.
- **D3 Menu style preferences**: some cultures may prefer precise and detailed menus and recipes, while others prefer looser guidelines or drawing from traditional cooking techniques.

E.2 User Group Descriptions

E.2.1 North America. A prominent percentage of users in North America live in the state of California. They enjoy a wide range of cuisines but many also consider themselves as health- and environment-conscious, adopting pescatarian diets to avoid heavy consumption of meat. Some have also reported nut allergies.

These users like to take advantage of the fresh fruits and vegetables that grow in the California sun, such as spinach, kale, citrus fruits, avocados, peaches, and berries. Due to their proximity to the ocean, many also enjoy diverse varieties of seafood.

When preparing food, these users tend to follow precise recipes.

E.2.2 Middle East. Turkey is home to many users from the Middle East. Because of the country's predominantly Muslim population, many users follow the Islamic dietary laws (halal) and enjoy traditional Mediterranean cuisine. They avoid pork and alcohol and also fast during the month of Ramadan, during which they eat only during specific hours.

These users enjoy access to a variety of Mediterranean ingredients like lamb, poultry, pita bread, olives, dates, and a range of spices like paprika, coriander, and cinnamon.

When preparing food, these users tend to treat recipes as loose guidelines rather than precise instructions.

E.2.3 Asia. A significant portion of users in Asia hail from India. Many of these users adhere to a vegetarian diet due to their Hindu beliefs, while others may consume white meat such as poultry or fish but rarely beef. Many also avoid onion and garlic during certain religious festivals.

These users enjoy access to a wide range of South Asian ingredients such as lentils, chickpeas, paneer (cheese); various earthy spices such as turmeric, cumin, and cardamom; and tropical fruits including papayas, guava, and mangoes.

These users rarely use written recipes, but instead rely on recipes and techniques passed on through word of mouth.

P#	Job Title	YoE	LLM Application Area	Education Background	Region
P1	UX Designer	6–10	Domain-specific QA	Visual/Industrial Design; Computing	Denmark
P2	UX Designer	6–10	Conversational Search	Visual/Industrial Design	Canada
P3	Principal UX Designer	6–10	Domain-specific QA	Visual/Industrial Design	U.S.
P4	Content Designer	6–10	Conversational Search; Domain-specific QA	Humanities	U.S.
P5	Principal Content Designer	6–10	Conversational Search	Humanities	U.S.
P6	Content Designer	1–2	Recommendation	Humanities	U.S.
P7	UX Researcher	3–5	Domain-specific QA	Visual/Industrial Design; Social & Behavioral Sciences	Ireland
P8	UX Designer	3–5	Creativity Support Tools	Visual/Industrial Design	U.S.
P9	Senior UX Designer	11+	Conversational Search; Creativity Support Tools	Visual/Industrial Design	U.S.
P10	UX Researcher	11+	Domain-specific QA	Social & Behavioral Sciences	U.S.
P11	UX Designer	3–5	Text Editing & Generation; Domain-specific QA	Social & Behavioral Sciences; Computing	U.S.
P12	Principal Content Designer	11+	Text Editing & Generation; Domain-specific QA	Humanities	U.S.

Table 1: Details of our participants (job title, years of experience in design, LLM application area of their product/feature, and education background before starting their current role) in our formative study. All participants used Figma in their day-to-day work, and have experience designing user experiences for LLM-powered products.

G#	P#	Prior LLM Design Experience?	Job Title	YoE	Education Background
G1	P1	No	Design Student	3–5	Visual/Industrial Design
	P2	No	Design Student	1–2	Visual/Industrial Design
	P3	No	UX Designer	3–5	Visual/Industrial Design; Computing; Information
G2	P4	No	UX Designer	1–2	Visual/Industrial Design; Computing
	P5	No	UX Designer	3–5	Visual/Industrial Design
	P6	No	UX Designer	1–2	Visual/Industrial Design
G3	P7	No	UX Designer	1–2	Visual/Industrial Design
	P8	No	UX Designer	1–2	Visual/Industrial Design; Management
	P9	No	Design Student	1–2	Visual/Industrial Design; Computing; Information
G4	P10	Yes	Principal Content Designer	11+	Humanities
	P11	Yes	Principal Content Designer	6–10	Humanities, Visual/Industrial Design; Social & Behavioral Sciences
G5	P12	Yes	Senior UX Designer	3–5	Visual/Industrial Design
	P13	Yes	UX Designer	3–5	Visual/Industrial Design
	P14	Yes	UX Designer	3–5	Social & Behavioral Sciences; Computing
G6	P15	Yes	Content Designer	1–2	Humanities
	P16	Yes	UX Researcher	11+	Social & Behavioral Sciences
	P17	Yes	UX Designer	1–2	Information

Table 2: Details of our participants (job title, years of experience in design, LLM application area of their product/feature, and educational background before starting their current role) in our design study. All participants were based in the U.S., and used Figma in their day-to-day work.