Design Theater: A Benchmark for Generative UI

Kashif Imteyaz[1*], **Kaif Imteyaz**[2], **Nakul Rajpal**[1], **Kaif Shaikh**[3], **Michael Muller**[4], **Saiph Savage**[1]

[1]Northeastern University, United States
[2]Jamia Hamdard, India
[3]Saarland University, Germany
[4]Independent Researcher, United States
imteyaz.k@northeastern.edu

Abstract

Generative UI tools promise to democratize UI design by turning natural language descriptions into complete interfaces. Alongside the interface, these tools generate user-facing design rationales that explain their layout, accessibility, and design choices. However, it remains unclear whether these stated rationales are actually reflected in the interfaces they produce. We call this disconnect "Design Theater": plausible and confident design rationales that have little relationship to the actual implementation. To study this phenomenon, we introduce a benchmark and three metrics for measuring Design Theater. The benchmark includes 24 UI generation tasks spanning structural, styling, and functional design requirements. Using this benchmark, we evaluate 120 interfaces created by five generative UI tools. On average, over 25% of user-facing design rationales are not implemented in the generated interface, and the implementation failure increases to 34% for functional requirements. Tools recognize roughly half of the UX principles embedded in prompts (mean = 0.54), with four of five tools implementing 6% or fewer functional principles. We also measure interface similarity across tools and find convergence in visual appearance and layout organization, with greater variation in color choices. Overall, we contribute: 1) the concept of Design Theater; 2) a benchmark with metrics for assessing whether the stated reasoning of generative UI tools is reflected in their implementations; 3) and findings from a systematic evaluation of these tools. We discuss what these findings mean for the design and evaluation of generative UI tools.

Introduction

Generative UI tools are increasingly positioned as a way to lower barriers to interface design by allowing users to generate interface prototypes from natural-language prompts (Takaffoli, Li, and Mäkelä 2024; Chen, Knearem, and Li 2025; Zhou, Li, and Yu 2024). Recent work describes such tools as valuable not only for UX designers, but also for adjacent roles such as developers and product managers (Chen, Knearem, and Li 2025). Generative UI tools such as Vercel v0, Claude Artifacts, ChatGPT Canvas, Firebase Studio, and Bolt support prompt-based workflows in which users describe an interface or application idea and receive generated code, previews, or deployable prototypes (Vercel 2023; Anthropic 2026; OpenAI 2024; Google 2026; StackBlitz 2026).

Beyond generating interfaces, these tools often narrate their design process through what we call *user-facing design reasoning*: explanations of what they intend to create, why particular design decisions are appropriate, and how the output addresses layout, accessibility, color, interaction, or broader usability principles (Nielsen 1994). For first-time creators, product managers, engineers, or other non-design stakeholders, such reasoning can function as a signal of design competence, making generated interfaces appear more deliberate, principled, and trustworthy (Liao and Sundar 2022; Sun et al. 2026). As AI-generated interfaces move from experimentation into organizational design and development workflows, this reasoning may become part of how teams explain, review, and justify design decisions (Son et al. 2026; Takaffoli, Li, and Mäkelä 2024). As a result, users may treat the tool's reasoning as evidence that usability, accessibility, and interaction-design concerns were properly addressed, even when they lack the expertise to verify (Kaur et al. 2020; Sun et al. 2026). For example, when a generative UI tool states that it will create "a responsive three-column grid with accessible contrast ratios and keyboard-navigable tabs", it creates the impression of careful, principle-driven design. Yet it remains unclear whether this reasoning is actually reflected in the generated interface. This is especially concerning as design work is increasingly delegated to AI tools and reviewed by product managers, engineers, or other stakeholders who may not have formal design training. Without methods for evaluating whether generative UI tools follow through on their user-facing design reasoning, teams may over-trust interfaces that appear professionally designed while failing to implement critical usability, accessibility, or responsiveness requirements (Liao and Sundar 2022).

To better understand and study this phenomenon, we introduce the concept of *Design Theater*: the production of user-facing design reasoning that bears little relationship to the actual design implementation. We define it by its effect on the reader. A rationale written in the language of professional design reads as evidence of deliberate, principle-driven decision-making, whether or not those commitments are realized in the artifact. Persuasive reasoning invites less scrutiny of the generated artifact, allowing overreliance to emerge before verification occurs (Sun et al. 2026).

However, defining Design Theater is not enough. We also

*Corresponding author.

need benchmarks and metrics that can measure how often generative UI tools produce mismatches between their user-facing design reasoning and actual implementations. Quantifying Design Theater allows researchers and practitioners to move beyond anecdotal examples, compare tools systematically, and identify when interfaces appear trustworthy despite failing to implement key usability, accessibility, or interaction-design requirements.

To address this need, this paper presents a benchmark and three metrics for measuring Design Theater in generative UI tools. The benchmark evaluates reasoning-to-implementation alignment and design homogenization across 120 UI designs generated by five state-of-the-art UI agent tools across 24 design tasks. These tasks span three tiers of design tasks: structural tasks, including layout and information architecture; styling tasks, including color, typography, and spacing; and functional tasks, including interactivity and user flows. We pair the benchmark with three metrics: Thinking Fidelity Score (TFS), Principle Adherence Score (PAS), and Design Homogeneity Index (DHI). Together, these metrics quantify whether tools' user-facing design reasoning is reflected in their generated interfaces and whether different tools produce similar designs for the same prompt. Using this benchmark and set of metrics, we study the following research questions:

1. To what extent is the user-facing design reasoning produced by generative UI tools reflected in the interfaces they implement?
2. To what extent do generative UI tools recognize and implement UX principles that are implicitly embedded in natural-language prompts (tasks)?
3. How much do generated interfaces vary in their visual appearance, color schemes, and layouts when different generative UI tools are given the same task?

Our findings suggest that, on average, over 25% of user-facing design reasoning is not implemented in the generated interface, with implementation failures increasing to 34% for functional requirements. Tools recognize roughly half of the UX principles implicitly embedded in prompts (mean PAS = 0.54), with four of the five tools implementing 6% or fewer functional principles. We also find convergence across tools: when given the same prompt, generated interfaces showed narrow pairwise differences in visual appearance and layout organization, while color choices varied more widely. Based on these findings, we discuss the implications for how generative UI tools are evaluated and for the stakeholders who increasingly review their output.

Our paper makes the following contributions:

- **Conceptualization of Design Theater.** We introduce the concept of *Design Theater*, describing how generative UI tools produce plausible design rationales that may not correspond to their implemented interfaces.
- **A benchmark and metrics for evaluating Design Theater.** We introduce a benchmark with 24 UI design tasks and three complementary metrics, TFS, PAS, and DHI, for evaluating reasoning-to-implementation alignment, implicit UX principle adherence, and cross-tool design homogenization.
- **Empirical evaluation of generative UI tools.** Through a systematic study of 120 interfaces generated by five UI tools across 24 design tasks, we show that misalignment between user-facing reasoning and implementation is widespread and concentrates in functional design. We find that designs converge in visual appearance and layout when tools are given the same prompt.

Related Work

Vibe Coding and AI-Assisted Co-Creation

Vibe coding is an emerging practice in which developers build software by describing intent in natural language rather than writing code directly (Sarkar and Drosos 2025). Its recent rise aligns with coding tools such as OpenAI Codex, Claude Code, Cursor, and GitHub Copilot (OpenAI 2025; Inc. 2024; Anthropic 2025; GitHub 2021), which promise to reduce barriers to software development. These coding agents can produce code that compiles and runs, though prior work suggests that generated code matches users' intended outputs only at moderate rates (Yetiştiren et al. 2023).

Researchers have therefore begun examining how users interact with coding agents. A common theme is the redistribution of effort from writing code to evaluating AI output and managing conversational context (Sarkar and Drosos 2025). In this workflow, users may skip quality assurance steps, reducing the reliability of the final output (Fawzy, Tahir, and Blincoe 2025). These omissions can carry real consequences: studies have found that 40% of GitHub Copilot-generated code contained security weaknesses (Pearce et al. 2025), and an analysis of production repositories found similar issues in 30% of Python and 24% of JavaScript snippets (Fu et al. 2025).

These concerns are not limited to code correctness. AI-assisted creation also raises questions about output diversity (Doshi and Hauser 2024). In a large-scale writing experiment, access to a generative AI tool improved individual output quality but reduced collective output diversity across participants (Doshi and Hauser 2024). Similar homogenization effects have been identified in design tasks (Wadinambiarachchi et al. 2024) and creative ideation (Anderson, Shah, and Kreminski 2024). Related work also shows that generative systems can default to Western cultural conventions (Agarwal, Naaman, and Vashistha 2025), a concern that is especially relevant for interface design, where layout, color, typography, and interaction patterns are culturally situated.

Concerns about homogenization also build on a longer history of web design convergence. Goree et al. (2021) found that web designs became more similar after 2007, with layout distance alone dropping over 30%, driven largely by the adoption of shared libraries and frameworks like Bootstrap. Our work asks whether generative UI tools extend this trajectory while obscuring it, layering rationales of deliberate design choice over outputs that may default to a narrow band of conventions.

Beyond generating outputs, these systems increasingly narrate their process through user-facing reasoning traces that explain why particular choices were made. Prior work

suggests that users may scan or rely on these traces instead of evaluating outputs directly (Fawzy, Tahir, and Blincoe 2025). What remains unexamined is whether such traces accurately reflect what was built, and whether users can distinguish competent design from a persuasive description of competence.

From Design Templates to Generative UI Tools

Early low-code and template-based interface tools lowered the barrier to creating interactive systems by allowing users to assemble interfaces from predefined components rather than writing code from scratch. While simple and effective, these tools constrained users to the structures and interaction patterns provided by the template system. Subsequent research moved beyond templates toward machine-learning systems for translating interface images or sketches into code, detecting UI components, and recovering interface structure from screenshots (Beltramelli 2018; Jain et al. 2019; Wu et al. 2021, 2023). These systems advanced computational interface understanding, but often remained tied to specific input modalities, datasets, or interaction domains.

The integration of large language models further expanded design-generation workflows beyond fixed templates and task-specific models. LLM-powered systems now support a range of creative design tasks, including text-to-image product design (Tao et al. 2025), animation design (Tseng, Cheng, and Nichols 2024; Liu et al. 2024), and interactive story generation (Chung et al. 2022), and the use of generated images as probes in co-design (Imteyaz et al. 2026).

More recently, generative UI tools have extended this trajectory by producing interface artifacts from natural-language prompts. Tools such as Vercel v0 (Inc. 2026), Lovable (Dev 2025), and Replit (Replit 2026) support prompt-based workflows for generating interface code, previews, and deployable prototypes. Many also produce user-facing reasoning traces that explain the design choices behind the generated interface. This shift has prompted renewed attention to computational approaches for UI understanding, generation, and evaluation (Jiang et al. 2024; Peng et al. 2026), including work arguing that interfaces must increasingly be designed not only for human users but also for AI agents acting on users' behalf (Peng et al. 2026).

Although these tools can produce visually plausible and functional artifacts, many evaluations of AI code-generation systems still emphasize functional or code-level correctness (Chen et al. 2021; Yetiştiren et al. 2023). Design quality, however, is not reducible to functional correctness (Stone et al. 2005). A UI that renders without errors can still exhibit poor information architecture, incoherent visual hierarchy, inaccessible interaction patterns, or cultural and aesthetic biases embedded in training data (Agarwal, Naaman, and Vashistha 2025). Current generative UI tools offer limited support for verifying whether these qualities are present in the generated output, particularly when the tool's own reasoning traces suggest they have already been addressed.

LLM Reasoning Traces and User Interpretation

Chain-of-thought (CoT) prompting instructs language models to generate intermediate reasoning steps before producing a final answer, improving performance on arithmetic, common-sense, and symbolic reasoning tasks (Wei et al. 2022; Kojima et al. 2022; Wang et al. 2022). Subsequent work has treated these generated rationales as both a performance enhancer and a possible window into model behavior.

However, a separate line of work questions whether these traces faithfully reflect the model's internal reasoning. When biased features are introduced into prompts, model answers can shift while the accompanying rationales fail to account for those shifts, producing plausible but unfaithful explanations (Turpin et al. 2023). Other work finds that early reasoning steps can be modified without changing final outputs, further suggesting that generated rationales may function as post-hoc reconstructions rather than faithful process logs (Lanham et al. 2023; Ji et al. 2023).

The faithfulness of generated rationales is not only a model-centric concern; it also affects how users interpret and rely on AI systems. Empirical work shows that users treat reasoning traces as trust-calibration tools, using them to interpret code explanations and decide whether to accept or revise AI outputs (Sun et al. 2026). The presentation of rationales alone can shape trust in AI decisions, even when the underlying model behavior remains unchanged (Sun et al. 2026; Bertrand et al. 2022).

Most work on reasoning-trace faithfulness examines tasks with externally checkable answers, such as arithmetic, fact verification, and symbolic reasoning (Wei et al. 2022; Turpin et al. 2023; Chen et al. 2025; Bertrand et al. 2022). Design reasoning presents a different challenge. UI outputs are visual and interactive artifacts whose quality depends on contextual, multi-dimensional, and often subjective criteria, including usability, accessibility, visual hierarchy, and interaction flow. Unlike this literature, which asks whether rationales reflect the model's internal reasoning process, we ask a downstream question: whether the rationale is consistent with the artifact the user actually receives. For non-expert stakeholders, who often read these rationales as design documentation, the key issue is whether the rationale accurately describes the resulting artifact. Whether this relationship holds in open-ended creative tasks such as UI generation remains unexplored.

Automated Design Evaluation and UX Metrics

Usability evaluation has a longer history than its current automated forms suggest. Before heuristic methods became central in HCI, human factors research developed standardized instruments for evaluating workload, usability, situation awareness, and readability in safety-critical domains (Hart 2006; Brooke et al. 1996; Wickens et al. 2021). Heuristic evaluation entered this landscape as a deliberately lightweight alternative (Nielsen 1994), valued for making usability problems inspectable without requiring full user studies. We draw on heuristic and guideline-based evaluation pragmatically: these frameworks do not exhaust design quality, but they provide inspectable commitments that can

be checked against generated artifacts. Since design quality is situated and shaped by context, values, and power relations (Haraway 2013; Costanza-Chock 2020), our benchmark evaluates whether tools implement the commitments they themselves make, without claiming that any checklist can fully determine design appropriateness. Contextual Inquiry (Karen and Sandra 2017) and Scandinavian Participatory Design (Ehn 2017) offer a complementary tradition, grounding design assessment in real work contexts and the expertise of affected users rather than expert inspection alone. This tradition reminds us that heuristic and guideline-based evaluation can be useful without fully capturing situated design quality.

Alongside heuristic methods, design communities developed more implementable specifications, including platform interface guidelines (Apple Inc. 2023) and the Web Content Accessibility Guidelines (W3C 2024). Automated evaluation tools later operationalized some of these specifications and perceptual principles. Axe-core and Lighthouse evaluate accessibility and performance against established web standards (Systems 2024; Chrome 2024), while Aalto Interface Metrics computes perceptual measures such as visual clutter and learnability (Oulasvirta et al. 2018). More recent AI-driven methods provide design feedback on layout, color, and typography (Lee et al. 2020), predict visual attention on interfaces (Jiang et al. 2023), or use LLMs as design critics grounded in heuristic and guideline-based datasets (Duan et al. 2024; Jiang et al. 2024).

Across these approaches, evaluation criteria are typically grounded in external specifications such as Nielsen's heuristics, WCAG, platform guidelines, perceptual models, or human-factors instruments. However, they generally do not examine whether a tool's *own stated design reasoning* is reflected in its generated output. A tool may provide a rationale explaining that it prioritizes accessibility, describe its color choices in terms of WCAG contrast ratios, and narrate a deliberate information hierarchy, yet produce an interface that contradicts those commitments. Our work addresses this gap by treating usability, accessibility, and interaction-design principles as rationales that can be checked against generated artifacts, rather than as assurances that should be simply accepted.

Methodology

Our methodology is designed to quantify *Design Theater*: mismatches between the user-facing design reasoning produced by generative UI tools and the interfaces they actually implement. To do so, we introduce a benchmark composed of 24 UI generation tasks and three evaluation metrics: Thinking Fidelity Score (TFS), Principle Adherence Score (PAS), and Design Homogeneity Index (DHI). Together, the benchmark and metrics allow us to assess whether generative UI tools follow through on their stated design reasoning, recognize UX principles embedded in natural-language prompts, and converge toward similar interface designs, revealing potential design homogenization.

Generative UI Tools

We evaluate five widely used generative UI tools: ChatGPT, Claude, Firebase Studio, Vercel v0, and Bolt (Vercel 2023; Anthropic 2026; OpenAI 2024; Google 2026; StackBlitz 2026). We selected tools that: (1) accept natural-language descriptions as input; (2) generate functional HTML, CSS, JavaScript, or equivalent frontend code; and (3) produce user-facing reasoning traces that explain their design decisions.

For each tool, we used its default user-facing configuration. We did not modify system-level or developer-level prompts, and we left each tool's default model settings unchanged. This setup reflects how non-expert creators are likely to encounter and use these tools in practice (Subramonyam et al. 2025).

Benchmark Design

We introduce a benchmark of 24 UI generation tasks designed to evaluate reasoning-to-implementation alignment in generative UI tools. In the benchmark, each task consists of a natural-language prompt that asks a generative UI tool to create a complete interface for a particular scenario. The prompts are designed to implicitly embed UX principles through user needs, contextual constraints, audience characteristics, and usage scenarios, rather than explicitly instructing the tool to implement specific design principles. This setup allows us to evaluate whether generative UI tools can recognize and operationalize fundamental UX requirements that are implied by the prompt, reflecting how novice designers and non-expert users are likely to request interfaces in practice (Chen, Knearem, and Li 2025; Raees 2026; Zamfirescu-Pereira et al. 2023).

Our benchmark is organized into three tiers of design tasks that reflect core dimensions of UX design: structural organization, visual presentation, and functional interaction (Rosenfeld, Morville, and Arango 2015; Jiang et al. 2023; Nielsen 1994). Each tier foregrounds a different kind of design work, allowing us to examine where Design Theater might appear in the generated interfaces. Structural tasks help us to examine how generative UI tools organize information, styling tasks examine how tools make visual design decisions, and functional tasks examine how tools implement interactive behavior. This organization allows us to study whether reasoning-to-implementation gaps emerge differently across what an interface: 1) contains; 2) how it looks; and 3) how it behaves.

Tier 1: Structural Tasks Structural tasks ask generative UI tools to create interfaces where the central challenge is how information is structured (Rosenfeld, Morville, and Arango 2015). These tasks focus on whether tools can organize information in ways that help users understand what content exists, how different pieces of content relate to each other, where to find relevant information, and what actions to take next. We use these tasks to study whether tools can translate reasoning about information structure into concrete interface decisions, such as grouping related content, ordering sections meaningfully, creating navigation paths, routing different user groups, and surfacing high-priority infor-

mation. The tasks progress from simple single-audience interfaces, such as a local business website, to more complex civic or institutional systems where multiple user groups must locate different types of information.

Tier 2: Styling Tasks Styling tasks ask generative UI tools to create interfaces where the main design challenge is visual communication (Jiang et al. 2023). These tasks help us study whether tools can translate visual design reasoning into concrete styling decisions, such as color choices, typography, spacing, visual hierarchy, brand expression, readability, accessibility, and emotional tone. The tasks progress from straightforward branding scenarios to more complex interfaces for users with constrained devices, varying literacy levels, or heightened emotional stress.

Tier 3: Functional Tasks Functional tasks ask generative UI tools to create interfaces where the central challenge is how the interface behaves during user interaction (Nielsen 1994). These tasks focus on whether tools can implement interactive behavior that supports users as they move through a workflow, encounter errors, change input states, navigate between roles, or use accessibility features. We use these tasks to study whether tools can translate interaction-design reasoning into working interface behavior. The tasks progress from simple workflows, such as a reservation flow where users select a date, choose a time, enter their information, and confirm a booking, to more complex interactive systems that must serve large and diverse user populations.

Within each tier, tasks increase in complexity along four dimensions. First, they vary in the number of user audiences, from interfaces designed for one clear group to interfaces that must serve multiple groups, such as residents, administrators, and visitors. Second, they vary in the number of competing design requirements, such as balancing simplicity, accessibility, branding, and detailed information. Third, they vary in the depth of domain-specific constraints, such as bilingual content, regulatory requirements, limited device access, or low literacy. Fourth, they vary in the severity of consequences if the design fails, ranging from user confusion in a low-stakes website to missed emergency, health, or financial information in a high-stakes interface. This progression allows us to examine whether Design Theater becomes more pronounced as design scenarios become more complex.

Prompting and Generative UI Output Collection

We administered the 24 benchmark tasks, eight per tier, to each of the five generative UI tools in our study, producing 120 generated interfaces in total (5 tools $\times$ 24 tasks). For each tool-task combination, the prompt included the same fixed implementation instruction: the tool had to generate the interface using only HTML, CSS, and JavaScript, without external frameworks, libraries, or web searches. We kept this instruction identical across all tools and tasks so that differences in the generated interfaces would reflect each tool's interpretation of the task prompt, user-facing design reasoning, and implementation choices, rather than differences in external resources or selected libraries. For each output, we captured the generated code, the rendered interface, and the full user-facing reasoning trace, including design commentary, explanations, and any rationale produced before, during, or alongside code generation.

Metric 1: Thinking Fidelity Score (TFS)

TFS measures the extent to which a generative UI tool's user-facing design reasoning is reflected in the interface it generates. TFS operationalizes Design Theater by quantifying how much of the concrete design reasoning articulated by the tool is fully, partially, or not implemented in the final interface. In order to calculate this metric, we conduct:

1) Reasoning Element Extraction. For each reasoning trace, we extract every concrete and verifiable design reasoning element. A reasoning element must be specific enough to evaluate against the generated interface or code. For example, statements such as "I will use a three-column grid," "I will include keyboard-navigable tabs," or "I will add a dark mode toggle" are included. Vague reasoning, such as "I will make the design user-friendly," is excluded because it cannot be reliably verified against the final interface.

2) Reasoning Verification. Two independent evaluators compare each extracted element of user-facing design reasoning against the generated interface. Each reasoning element is classified as *Fully Implemented* (1.0), *Partially Implemented* (0.5), or *Not Implemented* (0.0), depending on the extent to which the stated design reasoning is realized in the implemented interface. Each reasoning element is scored against the generated interface rather than the task prompt. An element is counted as Not Implemented when the interface does not contain what the reasoning described, including cases where the tool implemented a different approach. Whether the interface meets the prompt's requirements is measured separately by PAS.

3) TFS Metric Computation.

$$\text{TFS} = \frac{1.0 \cdot N_{\text{full}} + 0.5 \cdot N_{\text{partial}}}{N_{\text{total}}} \tag{1}$$

A lower TFS indicates greater Design Theater because the tool's user-facing reasoning is less faithfully reflected in the generated interface. TFS measures the size of this gap, not its cause.

Metric 2: Principle Adherence Score (PAS)

PAS measures whether generative UI tools recognize and implement UX principles that are implicitly embedded in the prompt. Whereas TFS evaluates whether tools follow through on their own stated reasoning, PAS evaluates whether tools implement the underlying UX principles required by each prompt within a task (each task has two UX principles). To avoid defining these principles post hoc, we use a framework-first approach: we select UX principles from established HCI and design frameworks before evaluation, and then design each prompt to naturally embed those principles through user needs and contextual constraints. Structural tasks draw from information architecture principles (Rosenfeld, Morville, and Arango 2015), styling

tasks draw from visual perception and WCAG-related accessibility principles (Jiang et al. 2023; World Wide Web Consortium 2024), and functional tasks draw from usability heuristics (Nielsen 1994). To calculate PAS, we use:

1) Scoring Protocol. Each prompt (task) contains two target UX principles. For each generated interface, two independent evaluators score whether each principle was implemented: 1 if implemented and 0 if not implemented.

2) PAS Metric Computation.

$$\text{PAS} = \frac{\sum_{i=1}^{2} P_i}{2} \tag{2}$$

where $P_i \in \{0, 1\}$ indicates whether principle i is implemented.

For both TFS and PAS, evaluators first completed their ratings independently. Disagreements were then resolved through discussion, and any remaining unresolved cases were adjudicated by a third evaluator before final metric computation (O'Connor and Joffe 2020; McDonald, Schoenebeck, and Forte 2019). Inter-rater reliability on the independent ratings was substantial for TFS (linearly weighted Cohen's $\varkappa$ = 0.70; per-tool range 0.53–0.85) and almost perfect for PAS ($\varkappa$ = 0.90; per-tool range 0.75–1.00). Weighted $\varkappa$ is used for TFS because the classification is ordinal, and disagreements were almost entirely adjacent.

Metric 3: Design Homogeneity Index (DHI)

DHI measures design homogenization by examining how similar the generated interfaces are across tools, especially when those tools are given the same task prompt. Because UI similarity is multidimensional (Jiang et al. 2023), we do not collapse DHI into a single aggregate score. Instead, we report DHI as an index composed of three sub-measures that capture different aspects of interface similarity: overall visual appearance, color usage, and layout structure.

To calculate DHI, we first render all 120 generated interfaces as standardized 1200 $\times$ 1200 PNG screenshots. For each prompt, we compare the five interfaces generated by the five tools. This yields ten pairwise tool comparisons per prompt for each DHI sub-measure.

DHI-Visual Similarity Submeasure. We encode each screenshot using UIClip and compute pairwise visual similarity across tools for the same prompt (Wu et al. 2024). This sub-measure captures the overall visual appearance of the interface, including stylistic patterns, component aesthetics, spacing tendencies, and the general visual impression produced by the design.

DHI-Color Similarity. We represent each screenshot as a color histogram in CIELCh color space and compute pairwise color distance using Earth Mover's Distance (Wyszecki and Stiles 2000; Rubner, Tomasi, and Guibas 2000). This submeasure specifically captures how similar the interfaces are in their use of colors, including palettes, color balance, saturation, and color distribution throughout the interface.

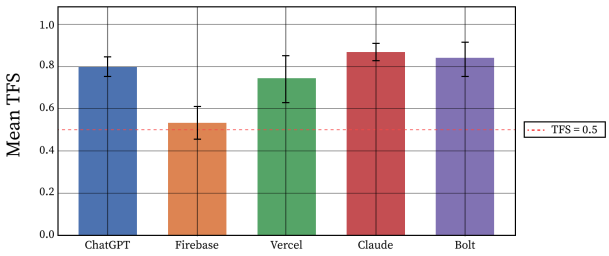

Figure 1: Overall Thinking Fidelity Score by Tool

Tool	Overall	Tier 1 Structural	Tier 2 Styling	Tier 3 Functional
Claude	**0.87** [0.82, 0.91]	0.83 [0.74, 0.91]	0.93 [0.89, 0.96]	**0.85** [0.78, 0.91]
Bolt	0.84 [0.75, 0.91]	0.83 [0.70, 0.95]	**0.95** [0.90, 0.99]	0.73 [0.55, 0.89]
ChatGPT	0.80 [0.75, 0.84]	**0.85** [0.75, 0.94]	0.78 [0.72, 0.85]	0.75 [0.69, 0.81]
Vercel v0	0.74 [0.63, 0.84]	0.83 [0.70, 0.94]	0.83 [0.66, 0.96]	0.56 [0.36, 0.77]
Firebase	0.53 [0.45, 0.61]	0.62 [0.54, 0.69]	0.55 [0.38, 0.70]	0.43 [0.34, 0.56]
Across tools	*0.75*	*0.79*	*0.81*	*0.66*

Table 1: Mean Thinking Fidelity Scores (TFS) by tool and tier. Best score in each column shown in bold.

DHI-Layout Similarity. We parse each screenshot using OmniParser v2.0 to detect UI elements, and then compute structural similarity using tree edit distance (Lu et al. 2024). This submeasure captures how similarly interface components are organized and positioned, including section ordering, navigation placement, grouping of elements, and overall page structure.

We report each DHI sub-measure separately because visual appearance, color usage, and layout structure can capture distinct dimensions of design homogenization. This allows us to identify not only whether generative UI tools converge, but also where that convergence occurs: in stylistic appearance, color choices, or interface layout organization.

Results

Metric 1: Thinking Fidelity Score (TFS) Using our TFS metric, we measured the proportion of stated design reasoning that each generative UI tool actually implemented across its generated interfaces.

Figure 1 shows the average TFS scores for each tool across its generated interfaces, while Table 1 in its second column reports the overall TFS results. No tool consistently delivered on its design reasoning, and the cross-tool mean TFS score was 0.75, indicating that roughly one in four stated design rationales did not fully appear in the generated interface. Four tools (Claude, Bolt, ChatGPT, and Vercel v0) clustered between 0.74 and 0.87, suggesting descriptively similar overall performance, with Claude achieving the highest alignment at 0.87. Firebase scored lowest, with a mean TFS of 0.53, below the range observed for the other four tools.

Figure 2 and Table 1 show that reasoning-to-

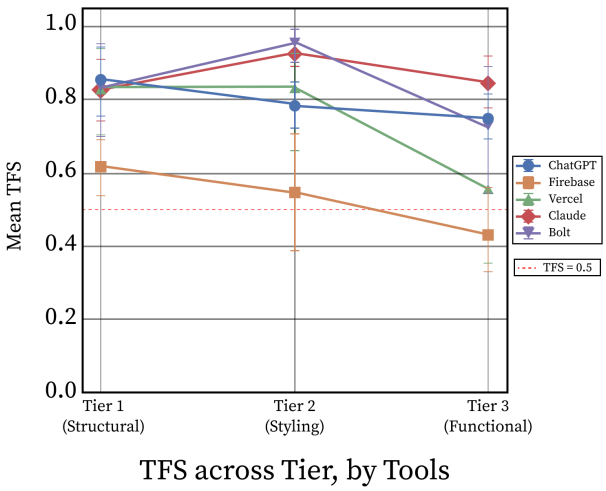

Figure 2: Thinking Fidelity Score across Tier, by Tools

implementation fidelity varied across both tools and task tiers. Functional tasks showed the weakest alignment across tools, with a cross-tool mean TFS of 0.66 compared with 0.79 for structural tasks and 0.81 for styling tasks. However, this decline was not uniform across tools. Claude maintained relatively stable fidelity across tiers, scoring 0.83 on structural tasks, 0.93 on styling tasks, and 0.85 on functional tasks. Bolt peaked on styling tasks before declining on functional tasks, while ChatGPT showed a gradual decline from structural to styling to functional tasks. Vercel v0 performed similarly on structural and styling tasks before dropping sharply on functional tasks, and Firebase remained the lowest-performing tool across all tiers. Together, these results suggest that Design Theater is not distributed evenly across task types: for some tools, reasoning-to-implementation gaps are broadly distributed, while for others they are concentrated especially in functional implementation.

Table 1 also shows that tools differed not only in average fidelity but also in the uncertainty around those estimates. Claude and ChatGPT had the narrowest overall bootstrap confidence intervals, with Claude ranging from 0.82 to 0.91 and ChatGPT from 0.75 to 0.84. In contrast, Vercel v0 and Bolt showed wider intervals, especially for Tier 3 functional tasks. For example, Vercel v0's Tier 3 mean was 0.56, with a confidence interval from 0.36 to 0.77. This suggests greater variability in how reliably Vercel v0 translated functional design claims into implementation across tasks. In some cases, the tool appeared to follow through on its stated reasoning, while in others, much of the promised functionality was absent or incomplete. These results indicate that mean TFS alone can obscure important differences in reliability: a tool with moderate but stable fidelity presents a different risk profile than one whose performance appears more variable across tasks.

Overall, from the perspective of a stakeholder reading a tool's reasoning trace, the practical question is not which tool to trust most, but how any tool's reasoning should be read as a description of what was actually built.

Metric 2: Principle Adherence Score (PAS) Our PAS metric measures whether generative UI tools implement the UX principles implicitly required by each design task. Figure 3 (left) reports the mean PAS for each tool across tasks, while Table 2 in its second column reports these same means with 95% confidence intervals. These intervals capture uncertainty around each tool's estimated average PAS across benchmark tasks. Overall, tools differed in how well they implemented the embedded UX principles. Mean PAS ranged from 0.29 for Firebase Studio to 0.73 for ChatGPT. In concrete terms, Firebase Studio implemented 14 of the 48 required principles, whereas ChatGPT implemented 35 of 48. Vercel v0 and Claude fell near the middle of the range, implementing only slightly more than half of the embedded UX principles. The non-overlapping 95% bootstrap confidence intervals between the highest- and lowest-performing tools provide descriptive evidence that ChatGPT and Firebase Studio occupied meaningfully different parts of the PAS range.

Figure 3 (right) reports each tool's mean PAS across the three task tiers. The clearest pattern is a sharp drop in PAS for functional tasks. Tools implemented structural tasks at relatively high rates, with Tier 1 scores ranging from 0.50 to 0.94, and styling tasks at similarly high rates, with Tier 2 scores ranging from 0.38 to 0.88. By contrast, functional tasks were implemented much less reliably. Four of the five tools scored 0.06 or below on Tier 3, and Firebase Studio implemented none of the functional principles across the eight functional tasks. Even ChatGPT, which scored highest overall, reached only 0.38 on Tier 3, indicating that functionality-related UX principles were the most difficult for these tools to implement. This pattern suggests that current tools share systematic weaknesses in interaction design principles such as visibility of system status, user control and freedom, and error prevention and recovery (i.e., they struggle implementing functional UX principles).

Tool	Mean PAS [95% CI]	Tier 1	Tier 2	Tier 3
ChatGPT	0.73 [0.59, 0.83]	0.94	0.88	0.38
Bolt	0.60 [0.46, 0.73]	0.88	0.88	0.06
Vercel v0	0.56 [0.42, 0.69]	0.81	0.81	0.06
Claude	0.54 [0.40, 0.67]	0.81	0.75	0.06
Firebase	0.29 [0.18, 0.43]	0.50	0.38	0.00

Table 2: Principle Adherence Score (PAS) comparison across generative UI tools.

Table 2 reports the mean PAS score of each tool overall and separately for each task tier (columns 3–5). Because PAS measures whether a tool successfully implements the UX principles implicitly required by a task, higher PAS scores indicate that the generated interface more fully adheres to the expected UX principles. From Table 2, we observe more variability between tools on structural (Tier 1) and styling (Tier 2) principles. ChatGPT and Bolt consistently achieved the highest PAS scores on these tiers. For example, ChatGPT achieved a mean PAS of 0.94 on structural principles and 0.88 on styling principles. In contrast,

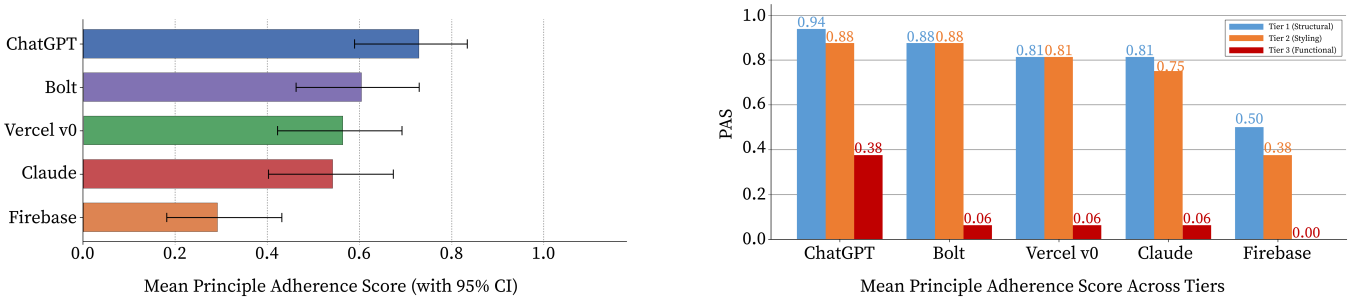

Figure 3: Principle Adherence Score across five UI agent tools. Left: mean PAS with 95% confidence intervals. Right: PAS by principle tier, showing consistently poor performance on functional principles

Firebase Studio achieved the lower score of 0.50 on structural principles and 0.38 on styling principles. This represents gaps of approximately 0.44 and 0.50 PAS points respectively relative to ChatGPT, indicating larger differences in the extent to which tools implemented these two requested UX principles. By contrast, the differences between tools on functional principles (Tier 3) were much smaller, with PAS scores ranging only from 0.00 to 0.38. However, this narrower range did not indicate similar performance quality across tools. Instead, it reflected a broader collapse in the ability of nearly all tools to implement functional UX principles at all, even when those principles were implicitly required by the task. Four of the five tools achieved Tier 3 PAS scores of only 0.06 or lower, while Firebase Studio failed to implement any functional principles across the Tier 3 tasks.

Metric 3: Design Homogeneity Index (DHI) Our DHI metric studies how similar the interfaces generated by the different tools are to one another, particularly when the tools receive the same prompt. We examined this similarity across three submetrics: visual similarity, color similarity, and layout similarity.

Figure 4 visualizes the DHI between pairs of generative UI tools across the three DHI submetrics. Each cell represents the mean distance between the interfaces produced by two tools when given the same prompts, pooled across all three task tiers. Lower values indicate that two tools tended to generate more similar interfaces, while higher values indicate greater divergence in their generated designs. From the heatmap, we observe that similarity patterns varied across submetrics. For example, Claude and Vercel v0 produced the most visually similar interfaces (DHI-Visual = 0.119) and the most similar color palettes (DHI-Color = 20.6), while ChatGPT and Claude produced the most different layouts (DHI-Layout = 0.211). The heatmap highlights how some tools converged toward similar design patterns, whereas others generated interfaces with different visual, color, and layout characteristics.

Table 3 summarizes the most and least similar tool pairs for each DHI submetric. Each value represents the mean pairwise distance between the interfaces generated by two tools across the 24 tasks. Lower values indicate that two tools tended to generate more similar interfaces, while higher values indicate greater divergence in the generated

Sub-metric	Most similar pair	Least similar pair	Range	Mean
DHI-Visual	Claude–v0 (0.119)	ChatGPT–Firebase (0.151)	0.119–0.151	0.137
DHI-Color	Claude–v0 (20.6)	Bolt–ChatGPT (39.7)	20.6–39.7	31.9
DHI-Layout	Bolt–Claude (0.181)	Claude–ChatGPT (0.211)	0.181–0.211	0.196

Table 3: Pairwise tool similarity by sub-metric, averaged across 24 tasks.

designs. From Table 3, we observe that Claude and Vercel v0 consistently produced some of the most similar interfaces, achieving the lowest distances on both visual similarity (0.119) and color similarity (20.6). In contrast, ChatGPT and Firebase produced the most visually different interfaces (0.151), while Bolt and ChatGPT produced the largest differences in color palettes (39.7). For layout similarity, Bolt and Claude generated the most similar layouts (0.181), whereas Claude and ChatGPT produced the most divergent layouts (0.211). The table also shows that the amount of variation differed across submetrics. DHI-Color exhibited the widest range of distances (20.6–39.7), indicating that overall tools varied in their color choices, while DHI-Visual and DHI-Layout showed narrower ranges, suggesting comparatively smaller differences in overall visual appearance and layout organization.

Discussion

Democratizing UI Generation Without Democratizing Evaluation Generative UI tools are often framed as democratizing design because they allow non-designers to rapidly prototype and generate functional interfaces from natural language (Takaffoli, Li, and Mäkelä 2024; Chen, Knearem, and Li 2025; Zhou, Li, and Yu 2024). Yet this democratization also redistributes design labor. Activities that were once mediated by trained designers, such as prototyping, design justification, or preliminary design evaluation, are increasingly being taken up by product managers, clients, and other non-design stakeholders. This shift creates new challenges around evaluation of the designs. Prior work shows that UX practitioners already use generative AI in design workflows (Takaffoli, Li, and Mäkelä 2024), but also report a need for better training to assess the quality of the interfaces. For novice users, this problem is even more pronounced: they may be able to generate an interface, but lack

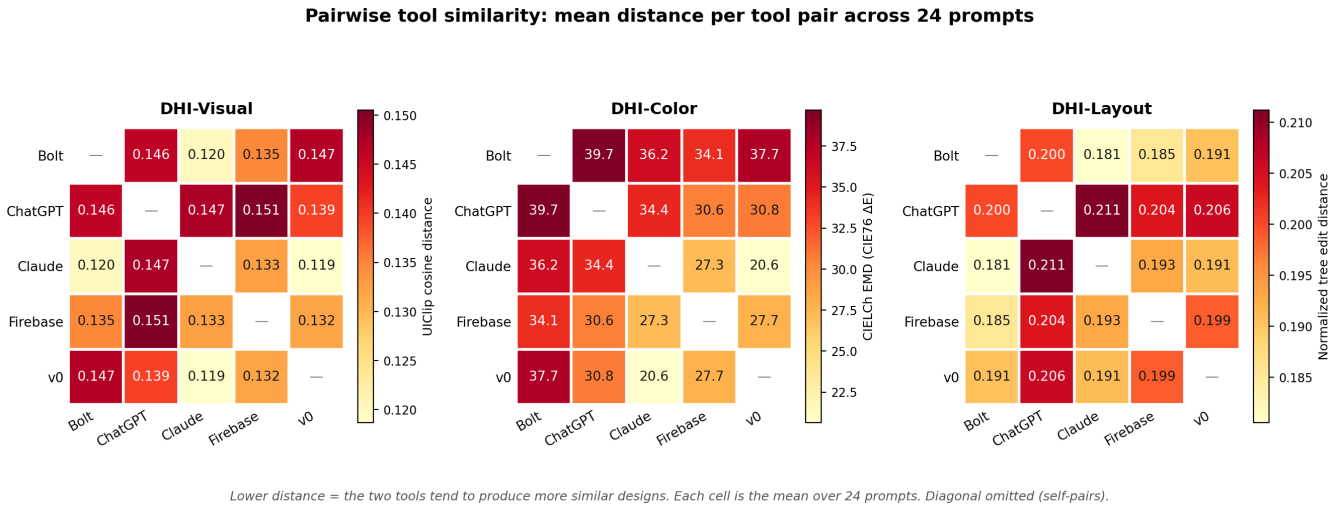

Figure 4: Design Homogeneity Index Heat Map

the expertise to judge whether it reflects their goals, satisfies requirements, or follows sound UX principles. The challenge becomes more consequential when generative UI tools also provide user-facing explanations of what was designed and why (Sun et al. 2026). By articulating layout decisions, invoking design principles, and describing tradeoffs in the language of trained designers (Son et al. 2026), these systems can appear to perform design expertise on the user's behalf. This may lead novice users to perceive generated interfaces as more complete, intentional, or correct than they actually are. In this way, generative UI tools can create a false sense of expertise: users may feel confident that an interface satisfies their design needs, but only because they lack the training to recognize usability problems, missing requirements, or flawed design assumptions.

Across our evaluation, roughly one in four stated design rationales did not fully appear in the generated interface. This gap became especially pronounced for functional tasks, where tools achieved a Tier 3 TFS of 0.66. The PAS results were even more striking. Recall that PAS measured whether generative UI tools implemented the UX principles implicitly required by each task. On functional UX principles, four of the five tools scored ≤ 0.06, meaning they implemented almost none of the interaction-design requirements embedded in the tasks, such as visibility of system status, user control and freedom, error prevention, and recovery. These functional failures are also the hardest for non-experts to detect. A missing color choice or layout inconsistency is often visible in the rendered interface. By contrast, missing state management, inaccessible interactions, weak error recovery, or absent user control may not be obvious from surface-level inspection. Thus, the most consequential failures are often the least visible, while the non-expert designers now responsible for evaluating generated interfaces may be the least equipped to identify these errors (Takaffoli, Li, and Mäkelä 2024).

Design evaluation depends on professional judgment developed through practice (Koskinen et al. 2011). This expertise is tacit, comparative, and built through repeated exposure to critique (Bardzell, Bardzell, and Stolterman 2014; Haraway 2013). It does not transfer to non-experts simply because interface generation has become faster or more fluent. In fact, the ability to recognize when a design rationale sounds convincing but the interface does not behave correctly is precisely what may be lost when trained designers are removed from the loop (Takaffoli, Li, and Mäkelä 2024). In this context, user-facing reasoning traces can begin to function like design documentation, but without the accountability structures that traditionally make such documentation meaningful. Prior work shows that users adjust their trust based on how rationales are presented, even when the underlying decision remains unchanged (Sun et al. 2026; Bertrand et al. 2022). The central risk, then, is not simply that non-designers are using design tools. Rather, it is that the ease of generating interfaces may create a sense of competence in evaluating design decisions, even when users lack the expertise needed to distinguish sound design choices from hollow or problematic ones.

The Evaluation Bottleneck in Generative UI Our findings point to a broader evaluation bottleneck in generative UI workflows. Generative UI tools make interface production faster, cheaper, and rhetorically polished, but evaluating the resulting artifact remains slow, expert-dependent, and multidimensional. A generated interface cannot be assessed only by whether it renders or whether the accompanying rationale sounds plausible. It must be evaluated as an artifact that combines design principles like visual hierarchy, interaction flow, accessibility, responsiveness, and contextual fit. Prior work on generative AI evaluation has warned that static and model-centric evaluations often miss modality, context, deployment conditions, and downstream harms (Rauh et al. 2024). In our study, this gap appears as a reasoning-to-artifact mismatch: generative UI tools can produce a professional-looking interface and a persuasive ex-

planation in seconds, while the work of verifying whether the explanation is actually implemented remains externalized to users (Ibrahim et al. 2025). Design Theater is therefore not only a failure of generation, but also a failure to produce artifacts that can be easily evaluated (Eriksson et al. 2025).

Our benchmark responds to this evaluation bottleneck by making three aspects of generated UI artifacts auditable. TFS evaluates whether an artifact implements the tool's own design rationale. PAS evaluates whether an artifact satisfies UX principles implied by the prompt. DHI evaluates whether tools converge toward similar visual, color, and layout defaults despite presenting their outputs as situated design choices. Together, these metrics do not claim to measure interface quality in full. Instead, following prior work that emphasizes construct clarity in AI measurement (Bommasani and Liang 2024), we treat TFS, PAS, and DHI as targeted instruments for detecting three specific failure modes: unimplemented design rationales, missed implicit UX obligations, and homogenized design defaults. We therefore position the benchmark not as a final ranking of generative UI tools, but as a reusable evaluation framework for asking whether user-facing design reasoning is grounded in implementation.

Design Homogenization and the Loss of Situated Design Our DHI results suggest that Design Theater is not only a problem of missing implementation but also of reduced variation in generated UI artifacts. Prior work has shown that AI-assisted creation can narrow the diversity of results in writing and ideation (Agarwal, Naaman, and Vashistha 2025; Wadinambiarachchi et al. 2024). In generative UI, homogenization is not only an output-level pattern; it is an accountability problem because similar artifacts arrive with rationales that frame them as context-sensitive design decisions (Ibrahim et al. 2025). Interfaces are not only expressive artifacts; they organize attention, access, trust, navigation, and user action. This matters because design knowledge is situated: what counts as clear, trustworthy, accessible, or safe depends on the users, institutions, communities, and power relations a design is meant to serve (Costanza-Chock 2020). When different users, domains, and stakes are repeatedly translated into similar visual and structural patterns, while being described as tailored design choices, generative systems risk flattening the contextual differences that design work is meant to preserve. This produces surface-level contextualization: the rationale names the audience, domain, or constraint, making the interface appear tailored, even when the rendered artifact relies on familiar defaults (Turpin et al. 2023; Sun et al. 2026). This matters because UI similarity is not only aesthetic: interfaces shape where users look (Jiang et al. 2023), how they navigate and act (Nielsen 1994), and who is included or excluded by design decisions (Costanza-Chock 2020). When generated interfaces cluster visually or structurally while being presented as situated design responses, generative UI risks making repeated conventions appear unique and professionally validated.

Our DHI metric does not by itself determine whether a particular similarity is harmful, useful, or contextually appropriate. Rather, it makes visible when generated interfaces converge despite rationales that describe them as tailored to specific users and contexts. This allows us to ask whether AI-generated interfaces reflect meaningful adaptation to context or the repetition of dominant design defaults. Without design review, stakeholders may read a polished explanation as evidence that a generic interface has been meaningfully tailored (Kaur et al. 2020; Sun et al. 2026).

Limitations and Future Work Our evaluation is artifact-centered. We inspect prompts, user-facing rationales, and rendered interfaces, but we do not measure how designers, product managers, engineers, clients, or novice creators interpret those rationales. Our study therefore shows that rationale-to-artifact mismatch exists, but not how often different stakeholders notice the mismatch or how it shapes trust, review, and deployment decisions. We evaluate Design Theater across five generative UI tools, enabling systematic comparison across tools, tiers, and metrics. This scope should not be interpreted as an exhaustive account of generative UI behavior: these systems change rapidly, and outputs may vary across model versions, interface defaults, and repeated generations. Our benchmark required tools to generate interfaces using only HTML, CSS, and JavaScript, which kept outputs comparable but removed component libraries that several of these tools normally rely on. Because each tool produced its reasoning under this same constraint and was scored only against commitments it chose to make, the constraint does not account for our TFS results. It is more relevant to PAS, where implementing functional principles without library support requires more work, and to DHI, where a shared implementation constraint may reduce variation across outputs. Future work should replicate this benchmark across additional tools, repeated samples, and examine whether fluent design rationales affect stakeholder trust in incomplete or weakly implemented interfaces.

Our metrics also capture specific dimensions of Design Theater rather than all the possible dimensions related to design quality. TFS measures whether design rationales are implemented, PAS measures whether selected implicit UX principles are recognized and implemented, and DHI measures visual, color, and layout convergence across tools. DHI is also computed only among the five tools we evaluate, without a reference distribution such as interfaces produced by human designers or a corpus of deployed sites. Our results therefore describe convergence across these tools. These metrics do not fully capture aesthetic judgment, cultural fit, emotional resonance, accessibility in lived use, or whether users can accomplish their goals in deployed contexts. Future work should therefore extend this evaluation to multi-screen flows, mobile interfaces, multi-turn design workflows, accessibility audits, usability testing, participatory evaluation, and multilingual or culturally situated design scenarios.

Conclusion

Generative UI tools produce user-facing design rationales alongside the interfaces they generate, but whether those rationales reflect what is actually implemented has remained

unexamined. In this work, we introduce Design Theater to name this gap and propose a benchmark with three metrics, Thinking Fidelity Score, Principle Adherence Score, and Design Homogeneity Index, to measure it. Across designs from five state-of-the-art tools, we find three consistent patterns. First, roughly one in four design rationales fails to appear in the generated interface. Second, tools recognize only about half of the UX principles implicitly required in their tasks, with near-universal collapse on functional principles such as visibility of system status, user control and freedom, and error prevention and recovery. Third, we find convergence across tools: interfaces generated for the same task showed narrow pairwise differences in visual appearance and layout organization, while color choices varied more widely. As user-facing reasoning traces become a primary signal of design competence for non-expert reviewers, measuring the distance between what tools say and what they build is critical to the responsible deployment of generative UI tools.

Acknowledgments

Special thanks to the anonymous reviewers for their feedback. This work was partially supported by NSF grants 2339443 and 2403252. We used Claude Sonnet 4.6 and Grammarly to refine language and clarity under full author oversight; all study design, analysis, literature review, and writing were conducted and verified by the authors.

References

Agarwal, D.; Naaman, M.; and Vashistha, A. 2025. AI Suggestions Homogenize Writing Toward Western Styles and Diminish Cultural Nuances. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25. New York, NY, USA: Association for Computing Machinery. ISBN 9798400713941.

Anderson, B. R.; Shah, J. H.; and Kreminski, M. 2024. Homogenization effects of large language models on human creative ideation. In *Proceedings of the 16th conference on creativity & cognition*, 413–425.

Anthropic. 2025. Claude Code: Agentic Coding System. https://www.anthropic.com/product/claude-code.

Anthropic. 2026. What Are Artifacts and How Do I Use Them? https://support.claude.com/en/articles/9487310-what-are-artifacts-and-how-do-i-use-them. Accessed: 2026-05-19.

Apple Inc. 2023. Human Interface Guidelines. https://developer.apple.com/design/human-interface-guidelines.

Bardzell, J.; Bardzell, S.; and Stolterman, E. 2014. Reading critical designs: supporting reasoned interpretations of critical design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 1951–1960.

Beltramelli, T. 2018. pix2code: Generating Code from a Graphical User Interface Screenshot. In *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '18. New York, NY, USA: Association for Computing Machinery. ISBN 9781450358972.

Bertrand, A.; Belloum, R.; Eagan, J. R.; and Maxwell, W. 2022. How cognitive biases affect XAI-assisted decision-making: A systematic review. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, 78–91.

Bommasani, R.; and Liang, P. 2024. Trustworthy social bias measurement. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 7, 210–224.

Brooke, J.; et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194): 4–7.

Chen, M.; Tworek, J.; Jun, H.; Yuan, Q.; de Oliveira Pinto, H. P.; Kaplan, J.; Edwards, H.; Burda, Y.; Joseph, N.; Brockman, G.; Ray, A.; Puri, R.; Krueger, G.; Petrov, M.; Khlaaf, H.; Sastry, G.; Mishkin, P.; Chan, B.; Gray, S.; Ryder, N.; Pavlov, M.; Power, A.; Kaiser, L.; Bavarian, M.; Winter, C.; Tillet, P.; Such, F. P.; Cummings, D.; Plappert, M.; Chantzis, F.; Barnes, E.; Herbert-Voss, A.; Guss, W. H.; Nichol, A.; Paino, A.; Tezak, N.; Tang, J.; Babuschkin, I.; Balaji, S.; Jain, S.; Saunders, W.; Hesse, C.; Carr, A. N.; Leike, J.; Achiam, J.; Misra, V.; Morikawa, E.; Radford, A.; Knight, M.; Brundage, M.; Murati, M.; Mayer, K.; Welinder, P.; McGrew, B.; Amodei, D.; McCandlish, S.; Sutskever, I.; and Zaremba, W. 2021. Evaluating Large Language Models Trained on Code. arXiv:2107.03374.

Chen, X.; Knearem, T.; and Li, Y. 2025. The genui study: Exploring the design of generative ui tools to support ux practitioners and beyond. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference*, 1179–1196.

Chen, Y.; Benton, J.; Anil, C.; Elhage, N.; Nguyen, T.; Esin, J.; Olsson, C.; Henighan, T.; Grosse, R.; et al. 2025. Reasoning models don't always say what they think. *Anthropic Research*.

Chrome, G. 2024. Lighthouse: Automated auditing, performance metrics, and best practices. https://github.com/GoogleChrome/lighthouse.

Chung, J. J. Y.; Kim, W.; Yoo, K. M.; Lee, H.; Adar, E.; and Chang, M. 2022. TaleBrush: Sketching Stories with Generative Pretrained Language Models. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22. New York, NY, USA: Association for Computing Machinery. ISBN 9781450391573.

Costanza-Chock, S. 2020. *Design justice: Community-led practices to build the worlds we need*. MIT press.

Dev, L. 2025. Lovable: AI-Powered Full-Stack Application Generator.

Doshi, A. R.; and Hauser, O. P. 2024. Generative AI enhances individual creativity but reduces the collective diversity of novel content. *Science advances*, 10(28): eadn5290.

Duan, P.; Cheng, C.-Y.; Li, G.; Hartmann, B.; and Li, Y. 2024. UICrit: Enhancing Automated Design Evaluation with a UI Critique Dataset. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, UIST '24. New York, NY, USA: Association for Computing Machinery. ISBN 9798400706288.

Ehn, P. 2017. Scandinavian design: On participation and skill. In *Participatory design*, 41–77. CRC press.

Eriksson, M.; Purificato, E.; Noroozian, A.; Vinagre, J.; Chaslot, G.; Gomez, E.; and Fernandez-Llorca, D. 2025. Can we trust ai benchmarks? an interdisciplinary review of current issues in ai evaluation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 8, 850–864.

Fawzy, A.; Tahir, A.; and Blincoe, K. 2025. Vibe Coding in Practice: Motivations, Challenges, and a Future Outlook – a Grey Literature Review. arXiv:2510.00328.

Fu, Y.; Liang, P.; Tahir, A.; Li, Z.; Shahin, M.; Yu, J.; and Chen, J. 2025. Security Weaknesses of Copilot-Generated Code in GitHub Projects: An Empirical Study. *ACM Trans. Softw. Eng. Methodol.*, 34(8).

GitHub. 2021. GitHub Copilot: Your AI Pair Programmer. https://github.com/features/copilot. Accessed: 2026-04-14.

Google. 2026. Firebase Studio. https://firebase.google.com/docs/studio. Accessed: 2026-05-19.

Goree, S.; Doosti, B.; Crandall, D.; and Su, N. M. 2021. Investigating the homogenization of web design: A mixed-methods approach. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, 1–14.

Haraway, D. 2013. Situated knowledges: The science question in feminism and the privilege of partial perspective 1. In *Women, science, and technology*, 455–472. Routledge.

Hart, S. G. 2006. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 50, 904–908. Sage publications Sage CA: Los Angeles, CA.

Ibrahim, L.; Huang, S.; Ahmad, L.; Bhatt, U.; and Anderljung, M. 2025. Towards interactive evaluations for interaction harms in human-AI systems. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 8, 1302–1310.

Imteyaz, K.; Muller, M.; Flores-Saviaga, C.; and Savage, S. 2026. Co-Designing Collaborative Generative AI Tools for Freelancers. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, 1–21.

Inc., A. 2024. Cursor: The AI Code Editor. https://www.cursor.com/.

Inc., V. 2026. v0: Generative UI by Vercel.

Jain, V.; Agrawal, P.; Banga, S.; Kapoor, R.; and Gulyani, S. 2019. Sketch2Code: transformation of sketches to UI in real-time using deep neural network. *arXiv preprint arXiv:1910.08930*.

Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y. J.; Madotto, A.; and Fung, P. 2023. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12): 1–38.

Jiang, Y.; Leiva, L. A.; Rezazadegan Tavakoli, H.; R. B. Houssel, P.; Kylmälä, J.; and Oulasvirta, A. 2023. UEyes: Understanding Visual Saliency across User Interface Types. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23. New York, NY, USA: Association for Computing Machinery. ISBN 9781450394215.

Jiang, Y.; Lu, Y.; Knearem, T.; Kliman-Silver, C. E.; Lutteroth, C.; Li, T. J.-J.; Nichols, J.; and Stuerzlinger, W. 2024. Computational Methodologies for Understanding, Automating, and Evaluating User Interfaces. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '24. New York, NY, USA: Association for Computing Machinery. ISBN 9798400703317.

Karen, H.; and Sandra, J. 2017. Contextual inquiry: A participatory technique for system design. In *Participatory design*, 177–210. CRC Press.

Kaur, H.; Nori, H.; Jenkins, S.; Caruana, R.; Wallach, H.; and Wortman Vaughan, J. 2020. Interpreting interpretability: understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, 1–14.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213.

Koskinen, I.; Zimmerman, J.; Binder, T.; Redström, J.; and Wensveen, S. 2011. *Design research through practice: From the lab, field, and showroom*. Elsevier.

Lanham, T.; Chen, A.; Radhakrishnan, A.; Steiner, B.; Denison, C.; Hernandez, D.; Li, D.; Durmus, E.; Hubinger, E.; Kernion, J.; et al. 2023. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*.

Lee, C.; Kim, S.; Han, D.; Yang, H.; Park, Y.-W.; Kwon, B. C.; and Ko, S. 2020. GUIComp: A GUI Design Assistant with Real-Time, Multi-Faceted Feedback. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, 1–13. New York, NY, USA: Association for Computing Machinery. ISBN 9781450367080.

Liao, Q. V.; and Sundar, S. S. 2022. Designing for responsible trust in AI systems: A communication perspective. In *Proceedings of the 2022 ACM conference on fairness, accountability, and transparency*, 1257–1268.

Liu, V.; Kazi, R. H.; Wei, L.-Y.; Fisher, M.; Langlois, T.; Walker, S.; and Chilton, L. 2024. Logomotion: Visually grounded code generation for content-aware animation. *arXiv preprint arXiv:2405.07065*, 2(3): 4.

Lu, Y.; Yang, J.; Shen, Y.; and Awadallah, A. 2024. Omniparser for pure vision based gui agent. *arXiv preprint arXiv:2408.00203*.

McDonald, N.; Schoenebeck, S.; and Forte, A. 2019. Reliability and inter-rater reliability in qualitative research: Norms and guidelines for CSCW and HCI practice. *Proceedings of the ACM on human-computer interaction*, 3(CSCW): 1–23.

Nielsen, J. 1994. Usability inspection methods. In *Conference companion on Human factors in computing systems*, 413–414.

OpenAI. 2024. Introducing Canvas. https://openai.com/index/introducing-canvas/. Accessed: 2026-05-19.

OpenAI. 2025. Codex. https://openai.com/index/introducing-codex/.

Oulasvirta, A.; De Pascale, S.; Koch, J.; Langerak, T.; Jokinen, J.; Todi, K.; Laine, M.; Kristhombuge, M.; Zhu, Y.; Miniukovich, A.; Palmas, G.; and Weinkauf, T. 2018. Aalto Interface Metrics (AIM): A Service and Codebase for Computational GUI Evaluation. In *Adjunct Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, UIST '18 Adjunct, 16–19. New York, NY, USA: Association for Computing Machinery. ISBN 9781450359498.

O'Connor, C.; and Joffe, H. 2020. Intercoder reliability in qualitative research: Debates and practical guidelines. *International journal of qualitative methods*, 19: 1609406919899220.

Pearce, H.; Ahmad, B.; Tan, B.; Dolan-Gavitt, B.; and Karri, R. 2025. Asleep at the keyboard? assessing the security of github copilot's code contributions. *Communications of the ACM*, 68(2): 96–105.

Peng, K.; Nichols, J.; Lutteroth, C.; Knearem, T.; Kretzer, F.; Bigham, J. P.; Maedche, A.; and Jiang, Y. 2026. Human-AI-UI Interactions Across Modalities. In *Proceedings of the Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems*, 1–8.

Raees, M. 2026. Understanding and Designing Human-Centered AI Interactions for Non-Expert Users. In *Proceedings of the Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems*, 1–6.

Rauh, M.; Marchal, N.; Manzini, A.; Hendricks, L. A.; Comanescu, R.; Akbulut, C.; Stepleton, T.; Mateos-Garcia, J.; Bergman, S.; Kay, J.; et al. 2024. Gaps in the safety evaluation of generative AI. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, volume 7, 1200–1217.

Replit, I. 2026. Replit: The Collaborative Browser-Based IDE.

Rosenfeld, L.; Morville, P.; and Arango, J. 2015. *Information architecture: for the web and beyond*. " O'Reilly Media, Inc.".

Rubner, Y.; Tomasi, C.; and Guibas, L. J. 2000. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2): 99–121.

Sarkar, A.; and Drosos, I. 2025. Vibe coding: programming through conversation with artificial intelligence. arXiv:2506.23253.

Son, K.; Choi, D.; Kim, T. S.; Kim, Y.-H.; Yun, S.; and Kim, J. 2026. ClearFairy: Capturing Creative Workflows through Decision Structuring, In-Situ Questioning, and Rationale Inference. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, 1–31.

StackBlitz. 2026. Introduction to Bolt. https://support.bolt.new/building/intro-bolt. Accessed: 2026-05-19.

Stone, D.; Jarrett, C.; Woodroffe, M.; and Minocha, S. 2005. *User interface design and evaluation*. Elsevier.

Subramonyam, H.; Thakkar, D.; Ku, A.; Dieber, J.; and Sinha, A. K. 2025. Prototyping with prompts: Emerging approaches and challenges in generative ai design for collaborative software teams. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 1–22.

Sun, X.; Wei, S.; Bosch, J. A.; Echizen, I.; Sugawara, S.; and El Ali, A. 2026. Seeing the Reasoning: How LLM Rationales Influence User Trust and Decision-Making in Factual Verification Tasks. In *Proceedings of the Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems*, 1–7.

Systems, D. 2024. axe-core: Accessibility Testing Engine. https://github.com/dequelabs/axe-core. Accessed: 2025-06-01.

Takaffoli, M.; Li, S.; and Mäkelä, V. 2024. Generative AI in user experience design and research: how do UX practitioners, teams, and companies use GenAI in industry? In *Proceedings of the 2024 ACM Designing Interactive Systems Conference*, 1579–1593.

Tao, S.; Liang, I.; Peng, C.; Wang, Z.; Palani, S.; and Dow, S. P. 2025. DesignWeaver: dimensional scaffolding for text-to-image product design. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, 1–26.

Tseng, T.; Cheng, R.; and Nichols, J. 2024. Keyframer: Empowering animation design using large language models. *arXiv preprint arXiv:2402.06071*.

Turpin, M.; Michael, J.; Perez, E.; and Bowman, S. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Advances in Neural Information Processing Systems*, 36: 74952–74965.

Vercel. 2023. Announcing v0: Generative UI. https://vercel.com/blog/announcing-v0-generative-ui. Accessed: 2026-05-19.

W3C. 2024. Web Content Accessibility Guidelines (WCAG) 2.2. World Wide Web Consortium. Accessed: 2026-05-04.

Wadinambiarachchi, S.; Kelly, R. M.; Pareek, S.; Zhou, Q.; and Velloso, E. 2024. The effects of generative AI on design fixation and divergent thinking. In *Proceedings of the 2024 CHI conference on human factors in computing systems*, 1–18.

Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; and Zhou, D. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Wickens, C. D.; Helton, W. S.; Hollands, J. G.; and Banbury, S. 2021. *Engineering psychology and human performance*. Routledge.

World Wide Web Consortium. 2024. Web Content Accessibility Guidelines (WCAG) 2.2. W3c recommendation, World Wide Web Consortium (W3C). Accessed: 2026-05-19.

Wu, J.; Krosnick, R.; Schoop, E.; Swearngin, A.; Bigham, J. P.; and Nichols, J. 2023. Never-ending Learning of User Interfaces. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23.

New York, NY, USA: Association for Computing Machinery. ISBN 9798400701320.

Wu, J.; Peng, Y.-H.; Li, X. Y. A.; Swearngin, A.; Bigham, J. P.; and Nichols, J. 2024. UIClip: a data-driven model for assessing user interface design. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, 1–16.

Wu, J.; Zhang, X.; Nichols, J.; and Bigham, J. P. 2021. Screen Parsing: Towards Reverse Engineering of UI Models from Screenshots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, UIST '21, 470–483. New York, NY, USA: Association for Computing Machinery. ISBN 9781450386357.

Wyszecki, G.; and Stiles, W. S. 2000. *Color science: concepts and methods, quantitative data and formulae*. John wiley & sons.

Yetiştiren, B.; Özsoy, I.; Ayerdem, M.; and Tüzün, E. 2023. Evaluating the code quality of ai-assisted code generation tools: An empirical study on github copilot, amazon codewhisperer, and chatgpt. *arXiv preprint arXiv:2304.10778*.

Zamfirescu-Pereira, J. D.; Wong, R. Y.; Hartmann, B.; and Yang, Q. 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.

Zhou, Z.; Li, Y.; and Yu, J. 2024. Exploring the application of LLM-based AI in UX design: an empirical case study of ChatGPT. *Human–Computer Interaction*, 1–33.

Appendix

Data and Materials Availability

The full evaluation corpus, including the design task prompts, generated UI artifacts, user-facing reasoning traces, and outputs produced by each tool, is publicly available at:

https://github.com/kashifimteyaza/design-theater-main

The underlying models used by the tools during the study included GPT-5 Thinking (ChatGPT), Claude Sonnet 4.5 (Claude and Bolt), Claude Haiku 4.5 (Vercel v0), and Gemini 2.5 Pro (Firebase Studio).

Design Tasks

Tier 1: Structural Tasks

Task 1.1 — Simple Business Site (Low) **Prompt.** I run a small neighborhood coffee shop called 'Morning Brew' that's been serving the community for 3 years. We're known for our artisanal espresso drinks, fresh pastries, and cozy atmosphere where locals work and study. I need a simple website that captures our welcoming vibe and helps new customers find us.

Please create our homepage and walk me through your design strategy. Explain how you'll organize the content and what sections you think are most important for a local coffee shop trying to attract both regular customers and people discovering us for the first time.

Task 1.2 — Multi-Section Site (Medium-Low) **Prompt.** I'm launching 'FitCore Studio,' a boutique fitness center offering yoga, pilates, and strength training classes. We have 6 certified trainers, offer both group classes and personal training, and pride ourselves on creating a non-intimidating environment for all fitness levels.

Our website needs to serve multiple purposes: showcase our class schedule, introduce our trainers, explain our philosophy, and convert visitors into members. Please design our homepage and explain your strategy for organizing these different content areas. How will you help potential members understand what makes us different from big gym chains?

Task 1.3 — Complex Business Site (Medium)

Prompt. Thompson & Associates is a mid-sized law firm with 15 attorneys across 4 practice areas: corporate law, family law, personal injury, and estate planning. We've been serving clients for 20 years and handle everything from simple wills to complex corporate mergers.

Our new website needs to establish credibility, help potential clients find the right attorney for their needs, showcase our case results, and provide existing clients with resources. We also want to educate visitors about legal processes to position ourselves as trusted advisors.

Please create our homepage and detail your approach to organizing this complex information hierarchy. How will you help different types of visitors (individuals vs. businesses, different legal needs) find what they're looking for quickly?

Task 1.4 — Multi-Audience Platform (Medium-High)

Prompt. Riverside University is a mid-sized liberal arts college with 8,000 students. Our homepage serves drastically different audiences: high school students researching colleges, current students accessing resources, faculty managing courses, alumni staying connected, and parents seeking information.

Each group has completely different goals when they visit our site. Prospective students want academic programs and campus life info. Current students need quick access to schedules, grades, and student services. Faculty need teaching resources and administrative tools. Alumni want news and giving opportunities. Parents want safety info and financial aid details.

Design our homepage and explain your strategy for serving these diverse audiences without creating chaos. How will you help each group find their relevant information while maintaining a cohesive university brand?

Task 1.5 — Complex Organizational Site (High)

Prompt. Metro Health System is a comprehensive healthcare network serving 500,000 people across 3 cities. We operate 2 main hospitals, 12 specialty clinics, an urgent care network, and emergency services. Our services span everything from routine checkups to complex cardiac surgery.

Our website must serve patients finding doctors and booking appointments, emergency information for crisis situations, detailed service information for referring physicians, insurance and billing support, health education resources, and career opportunities for medical professionals.

The challenge is massive: we have 200+ physicians across 30+ specialties, complex insurance relationships, multiple locations with different services, and we serve both English and Spanish-speaking communities. Patient safety and accessibility compliance are critical.

Create our homepage and explain how you'll architect this complex information system. How will you help someone having a medical emergency find critical information while also serving routine patient needs and professional requirements?

Task 1.6 — E-commerce Platform (Medium-High)

Prompt. ShopLocal is a multi-vendor marketplace connecting local artisans with customers across the region. We have 300+ sellers offering handmade goods, vintage items, and local produce.

Our research shows three distinct user groups with very different needs: customers want quick product discovery and trust signals, sellers need efficient inventory management while juggling other jobs, and our admin team requires oversight tools that don't overwhelm them. The challenge is creating an experience that feels cohesive despite serving these diverse audiences.

Many of our sellers are older artisans who aren't tech-savvy, and we also need to comply with accessibility standards for government contracts. Customers shop both on mobile during farmers markets and on desktop when planning events. Sellers often update inventory from their phones between craft fairs.

The platform needs to handle product uploads, order management, vendor applications, quality monitoring, and dispute resolution. Users should be able to easily correct mistakes like accidental orders or incorrect listings. We're competing with Etsy's polished experience and local farmers markets' authentic charm.

Design our marketplace homepage explaining how you'll structure this multi-sided platform. How will you ensure each user type can efficiently achieve their goals while maintaining a unified brand experience?

Task 1.7 — News/Media Platform (Medium-High)

Prompt. Metro Daily News serves three metropolitan areas with local journalism. We publish 40+ articles daily across politics, business, sports, and community events, plus podcasts and premium investigative reporting.

Our analytics show readers access our site in three main contexts: quick morning headline scanning on phones, deep desktop reading during lunch breaks, and social media sharing throughout the day. We need to serve casual readers who just want top stories, business leaders tracking specific topics, and engaged citizens following local government.

The challenge is organizing high-volume daily content while making both breaking news and evergreen features discoverable. Readers get frustrated when they can't find yesterday's story or when the layout changes unexpectedly between devices. They want control over their news feed but not overwhelming customization options.

We must support readers with visual impairments (many are older adults) and provide Spanish language options for 30% of our market. The platform needs smooth subscription management, newsletter preferences, and comment moderation. Load times are critical — readers abandon if articles don't load in 3 seconds on mobile.

Create our news homepage detailing your information architecture strategy. How will you help different reader types quickly find relevant content while managing 40+ daily articles?

Task 1.8 — Municipal Government Site (High)

Prompt. The City of Riverside serves 150,000 residents who interact with dozens of city services, from paying water bills to reporting potholes, from finding park programs to accessing council meeting minutes.

Our user research revealed huge diversity: senior citizens who need large text and simple navigation, busy parents doing quick tasks on phones, business owners navigating complex permitting, and residents with limited English proficiency. Federal law requires full ADA compliance and content in both English and Spanish.

Citizens consistently tell us they can't find what they need because our current site is organized by department rather than by their actual needs. For example, getting a building permit requires visiting six different department pages. During emergencies, critical safety information gets buried under routine content.

The site must integrate with legacy payment systems, permit databases, and emergency alert systems. Citizens want to track their service requests and save commonly used forms. City staff need to update content without breaking the consistent experience residents rely on.

Design our city homepage explaining your approach to organizing complex government services. How will you help residents quickly complete tasks while meeting strict accessibility requirements and emergency communication needs?

Tier 2: Styling Tasks

Task 2.1 — Basic Brand Expression (Low)

Prompt. I'm launching 'Essence Jewelry,' a minimalist jewelry brand focused on timeless, elegant pieces made from ethically sourced materials. Our aesthetic is clean, modern, and sophisticated — think Scandinavian design principles applied to jewelry.

Our target customers are professional women aged 25–40 who value quality over quantity and prefer investment pieces they'll wear for years. They appreciate craftsmanship and are willing to pay more for ethical sourcing.

Create our homepage and explain your visual design choices. How will you use color, typography, spacing, and imagery to communicate our minimalist brand values and appeal to our sophisticated target audience?

Task 2.2 — Emotional Design (Medium-Low)

Prompt. Harmony Family Therapy is a practice specializing in couples counseling, family mediation, and adolescent therapy. Dr. Sarah Chen has 15 years of experience helping families navigate difficult transitions, communication issues, and relationship challenges.

Many of our potential clients are experiencing stress, anxiety, or crisis when they first visit our website. They need to feel safe, understood, and hopeful that therapy can help. Some are skeptical about counseling or nervous about taking the first step.

Design our homepage with a focus on creating the right emotional tone. Walk me through your approach to using visual design elements to make visitors feel comfortable and encourage them to reach out. How will you balance professionalism with warmth and approachability?

Task 2.3 — Brand Differentiation (Medium)

Prompt. RebelRoots is a plant-based food startup disrupting the meat substitute industry. Unlike boring health-focused competitors, we're targeting food lovers who happen to be vegan — people who want indulgent, satisfying meals that don't compromise on flavor.

Our brand personality is bold, fun, and unapologetically confident. We're the punk rock of plant-based foods. Our packaging uses vibrant colors and edgy graphics. We want to challenge the stereotype that vegan food is bland or virtuous.

Our competitors (Beyond Meat, Impossible, etc.) use very safe, corporate design. We need to stand out dramatically while still appearing trustworthy enough for grocery stores to stock us.

Create our homepage and detail your visual strategy for disrupting this conservative industry. How will you use design elements to communicate our rebellious brand while maintaining credibility with both consumers and retailers?

Task 2.4 — Complex Brand Balance (Medium-High)

Prompt. Luxe Verde is a luxury sustainable fashion brand creating $300–800 garments from recycled ocean plastic and organic materials. We're targeting affluent consumers who want to feel good about their purchases without sacrificing style or quality.

This creates a complex brand challenge: we must feel premium and exclusive (to justify high prices) while communicating our environmental mission (without seeming preachy). Our customers have sophisticated tastes and disposable income but also genuine environmental concerns.

We're competing against both traditional luxury brands (who ignore sustainability) and eco-friendly brands (who often look cheap or overly earnest). We need to feel like Céline or The Row in terms of luxury, but with authentic environmental credibility.

Design our homepage and explain how you'll visually balance these competing brand requirements. How will you communicate luxury and sustainability simultaneously without either message diluting the other?

Task 2.5 — Multi-Brand Architecture (High)

Prompt. Global Ventures Inc. is a parent company managing 5 successful subsidiary brands across different industries: TechFlow (B2B software), BrightStart (children's education), UrbanEats (food delivery), GreenBuild (sustainable construction), and WellnessWorks (corporate health programs).

Each subsidiary has its own strong brand identity and customer base. TechFlow is sleek and corporate. BrightStart is playful and colorful. UrbanEats is trendy and urban. GreenBuild is earth-conscious and professional. WellnessWorks is calming and health-focused.

Our parent company website needs to showcase all five brands while establishing Global Ventures as a credible, strategic holding company. We need to serve three audiences: potential acquisition targets, investors, and job seekers who might want to work for any of our companies.

Create our homepage and explain your visual strategy for maintaining brand hierarchy and coherence. How will you represent five distinct brand personalities under one corporate umbrella without creating visual chaos or diluting individual brand strength?

Task 2.6 — Healthcare Technology Brand (Medium-High)

Prompt. MedTech Solutions develops AI-powered diagnostic tools for hospitals. Our software helps radiologists detect cancer earlier, but we face a unique challenge: younger doctors embrace AI while senior physicians remain skeptical about replacing human judgment.

Hospital purchasing committees include both innovative tech directors and conservative medical boards. They need to see that we're cutting-edge enough to justify the investment but established enough to trust with patient lives. Our interface will be used in various conditions — bright ER rooms, dimly lit radiology suites, and on older hospital computers with poor displays.

The visual design must convey advanced technology while maintaining the gravitas expected in healthcare. Doctors often work 12-hour shifts, so the interface can't cause eye strain. We need to communicate complex AI confidence scores in ways that both tech-savvy residents and traditional practitioners can quickly understand.

Our competitors either look like dated medical software from the 90s or Silicon Valley apps that doctors don't trust. We need to find the sweet spot between innovation and medical professionalism.

Design our homepage and explain your visual strategy. How will you balance technological innovation with medical credibility while ensuring the design works across different hospital environments and user demographics?

Task 2.7 — Entertainment Streaming Brand (Medium-High)

Prompt. StreamVault curates independent films, documentaries, and international cinema for culturally curious viewers aged 25–55. We're not trying to compete with Netflix on

quantity — we're offering a thoughtfully selected collection for people who view film as art, not just entertainment.

Our challenge is appealing to serious cinephiles without intimidating casual viewers who might discover amazing content through us. The brand needs to feel sophisticated enough to justify our premium pricing but welcoming enough that someone's film-novice partner would feel comfortable browsing.

Users access our platform across smart TVs, tablets during commutes, and phones for quick browsing. The design must feel cohesive whether someone's having a solo art film experience or hosting a documentary night with friends. Many of our international films require subtitles, and some users have asked for audio descriptions for accessibility.

We want viewers to feel they're entering a curated cultural space, like a boutique cinema, not a corporate streaming warehouse. But we can't be so artistic that navigation becomes confusing — people still need to quickly find something to watch on a Tuesday night.

Create our platform homepage and detail your visual approach. How will you communicate cultural sophistication while keeping the platform approachable and easy to navigate across devices?

Task 2.8 — Fintech Startup Brand (High)

Prompt. CreditBuilder helps people with poor credit scores improve their financial health. Our users have often been rejected by traditional banks and feel ashamed about their financial situation. Many are dealing with the stress of debt, past mistakes, or circumstances beyond their control.

We need to feel trustworthy without being corporate, encouraging without being patronizing, and professional without being intimidating. Our users access the app on older Android phones with small screens and limited data plans. They're checking their credit during stressful moments — after being denied an apartment, before a job interview, or while trying to qualify for a car loan.

The visual design must work for users with varying literacy levels and those for whom English is a second language. Numbers and data need to be clear without overwhelming people who may have anxiety around finances. Users should feel empowered to take control, with the ability to hide sensitive information when checking the app in public.

Traditional banks feel cold and judgmental to our users, while predatory lenders seem too good to be true. We need to find the visual sweet spot that says 'we understand your struggle and we're here to genuinely help.'

Design our app homepage explaining your visual strategy. How will you create a brand that feels supportive and credible while being accessible to users facing financial and emotional challenges?

Tier 3: Functional Tasks

Task 3.1 — Simple Interaction (Low)

Prompt. I own 'Bella Vista,' an upscale Italian restaurant in downtown. We've been fully booked most nights and need to move away from phone reservations to an online system. Our customers are a mix of young professionals and older diners who aren't all tech-savvy.

We seat 80 people across 20 tables, serve dinner Tuesday–Sunday 5–10pm, and need to manage reservations for parties of 1–8 people. Some tables are better than others (window seats, private corners), but we don't want to overcomplicate the booking process.

Create our restaurant homepage with integrated reservation functionality. Walk me through how you'll design the booking flow to be simple enough for our least tech-savvy customers while giving us the management features we need. How will you handle the reservation process from customer selection to confirmation?

Task 3.2 — Multi-Step Process (Medium-Low)

Prompt. EduConnect is a tutoring platform connecting high school students with qualified tutors for SAT prep, math, science, and essay writing. Students need to browse tutor profiles, check availability, book sessions, and attend virtual meetings. Tutors need to manage their schedules, set rates, and track student progress.

Our typical flow: students search by subject and budget, read tutor reviews, book a trial session, attend via video chat, then book ongoing sessions if it's a good fit. Tutors set their availability weekly, can offer different session types (1-hour standard, 2-hour intensive, group sessions), and need payment processing.

The challenge is serving both sides of the marketplace while keeping the booking process simple for stressed high school students and their parents.

Design our platform homepage and detail your approach to the booking workflow. How will you handle the multi-step process from tutor discovery to session completion? What features will you implement to ensure smooth virtual sessions and ongoing relationships?

Task 3.3 — User-Generated Content (Medium)

Prompt. PhotoCommunity is a platform for local photographers to showcase work, get constructive feedback, and connect with potential clients for events, portraits, and commercial projects. We serve both hobbyists sharing their weekend shots and professionals building their businesses.

Photographers upload galleries, participate in weekly challenges, give and receive critiques, and list their services. Community members vote on featured photos, follow their favorite photographers, and can hire through the platform. We need moderation to keep feedback constructive and quality high.

The community aspect is crucial — we're not just a portfolio host, but a place where photographers genuinely help each other improve. However, we also need commerce features since many members want to monetize their skills.

Create our platform homepage and explain your strategy for balancing community engagement with commercial functionality. How will you design the user-generated content flows, feedback systems, and client connection features? What will encourage active participation while maintaining quality?

Task 3.4 — Complex User Flows (Medium-High)

Prompt. FreelanceFlow is a marketplace connecting businesses with freelance designers, developers, writers, and marketers. Unlike generic platforms, we focus on long-term relationships and quality matches rather than one-off gigs.

Our process: clients post detailed project briefs, freelancers submit custom proposals (not just bids), we facilitate interviews, handle contracts and milestones, process payments, and maintain ongoing relationships. Projects range from $500 logos to $50,000 website builds.

Key challenges: preventing scope creep, ensuring quality deliverables, handling revisions, managing international payments, maintaining trust between strangers, and supporting both parties through disputes.

We need sophisticated project management tools, file sharing, time tracking, invoice generation, and client approval workflows. The platform must work for both Fortune 500 companies and small business owners.

Design our marketplace homepage and detail your approach to managing these complex user flows. How will you handle the full project lifecycle from posting to payment? What systems will you implement to ensure successful project completion and ongoing client relationships?

Task 3.5 — Enterprise Functionality (High)

Prompt. ProjectMaster Enterprise is a comprehensive project management platform serving Fortune 500 companies managing complex, multi-million dollar initiatives with 50–500 team members across departments, time zones, and external vendors.

Our clients run projects like new product launches, corporate acquisitions, facility construction, and software implementations. These involve intricate dependencies, resource conflicts, budget approvals, compliance requirements, and executive reporting.

The platform must handle: resource planning and allocation, budget tracking with approval workflows, risk management and mitigation, stakeholder communication at multiple levels, document version control, vendor coordination, regulatory compliance tracking, and real-time executive dashboards.

Users range from individual contributors tracking tasks to C-suite executives monitoring portfolio performance. We need role-based permissions, automated reporting, integration with existing enterprise tools (Salesforce, SAP, etc.), and audit trails for everything.

The interface must be sophisticated enough for project managers while remaining accessible to executives who need high-level insights without complexity.

Create our enterprise platform homepage and explain your strategy for managing this level of complexity. How will you architect the user experience to serve different roles and use cases? What systems will you implement to handle enterprise-scale project coordination and reporting?

Task 3.6 — Social Learning Community (High)

Prompt. SkillBridge connects professionals for peer learning and mentorship. Our platform serves everyone from anxious recent graduates seeking guidance to senior executives learning new technologies. Members can simultaneously be teachers in their expertise area and learners in others.

The core challenge is balancing structure with organic community growth. Users want quality-assured content but also authentic peer connections. They need to find relevant mentors quickly but also browse serendipitously. Privacy is crucial — professionals want to control what colleagues can see about their learning goals.

Members access the platform during focused learning sessions on desktop and quick interactions on mobile between meetings. Some have learning disabilities requiring extra time or alternative formats. International members need content to work across time zones and languages.

The platform must handle course creation, progress tracking, peer reviews, discussion forums, and virtual meetups. Users frequently make mistakes like posting to wrong forums or enrolling in wrong courses, so we need smooth correction paths. Unlike LinkedIn Learning's corporate feel, we want to foster genuine peer support while maintaining content quality.

Create our platform homepage explaining your functional approach. How will you enable both structured learning and organic knowledge sharing while giving users control over their professional image?

Task 3.7 — Personal Finance Management (High)

Prompt. MoneyWise helps individuals and families manage their complete financial picture. Our users range from college students with simple checking accounts to families juggling mortgages, investments, and college savings. The emotional weight is significant — money discussions cause relationship stress and personal anxiety.

Users need to connect multiple bank accounts, but many are wary after data breaches at other companies. They want to understand their finances without getting overwhelmed by complex charts. The platform must present information differently for someone checking quick balances on their phone versus doing detailed tax planning on desktop.

Critical features include transaction categorization that users can correct, goal tracking that adapts to setbacks, and investment monitoring for varying risk tolerances. Some users have dyscalculia or other numerical processing challenges. Others are rebuilding after bankruptcy and need encouragement without judgment.

The challenge is making sophisticated financial tools accessible to users with varying financial literacy. Someone who doesn't understand compound interest should still benefit from investment tracking. Budget alerts need to be helpful, not shameful. Users should control what family members can see in shared accounts.

Design our platform homepage detailing your functional strategy. How will you make complex financial management accessible and encouraging while maintaining security and user trust?

Task 3.8 — Virtual Event Platform (High)

Prompt. EventSpace hosts virtual conferences from 50-person workshops to 5,000-attendee summits. Event organizers range from tech-savvy conference pros to volunteer-run nonprofits hosting their first virtual fundraiser.

Attendees have varying comfort with technology — some are digital natives while others are professionals forced online by circumstances. They join from different devices, time zones, and internet speeds. Many attendees have accessibility needs including screen readers, captions, and keyboard navigation.

The platform must handle registration, live streaming, breakout rooms, networking, and exhibit halls. Organizers need control over every aspect but can't be overwhelmed by options. Attendees want to network naturally but many are introverted or uncomfortable with forced video interactions. Speakers need confidence their presentation won't fail mid-talk.

Common pain points include attendees joining wrong sessions, missing key content due to time zone confusion, and feeling isolated without in-person energy. The platform should help users recover from mistakes like leaving sessions accidentally or missing recording permissions.

We're competing with Zoom's familiarity and specialized platforms' features. Organizers want professional results without complexity, while attendees want engagement without awkwardness.

Create our platform homepage explaining your approach to virtual events. How will you serve the diverse needs of organizers, speakers, and attendees while making large-scale virtual events feel smooth and engaging?

Principle Adherence Scoring Table

For each benchmark task we identified two target UX principles, drawn from established design frameworks before evaluation (see Methodology, Metric 2). Tables 4–6 list, for each task, the two principles, their definitions, and the prompt signals that embedded each principle without naming it directly.

Table 4: Tier 1 (Structural)

Task	Scenario	Principle	Definition	Signal in Prompt
1.1	Morning Brew (Coffee Shop)	Organization Systems (S1)	Content categorized into coherent schemes; related content grouped, unrelated separated.	"organize the content"; distinct offerings (espresso, pastries, atmosphere)
1.1	Morning Brew (Coffee Shop)	Hierarchy & Prioritization (S2)	Most important content elevated; page communicates what matters most.	"what sections you think are most important"
1.2	FitCore Studio	Organization Systems (S1)	Content categorized into coherent schemes.	"serve multiple purposes: showcase our class schedule, introduce our trainers, explain our philosophy, and convert visitors"
1.2	FitCore Studio	Labeling Systems (S4)	Clear, audience-appropriate labels; warm/inclusive language where needed.	"non-intimidating environment for all fitness levels"
1.3	Thompson & Associates (Law Firm)	Organization Systems (S1)	Content categorized into coherent schemes.	"4 practice areas," mixed services (wills to mergers), multiple content types (results, resources, education)
1.3	Thompson & Associates (Law Firm)	Findability (S5)	Each audience can reach relevant content with reasonable effort.	"different types of visitors (individuals vs. businesses, different legal needs) find what they're looking for quickly"
1.4	Riverside University	Navigation Systems (S3)	Global/local navigation enables orientation across the site.	"without creating chaos," 5 audience paths needed
1.4	Riverside University	Findability (S5)	Each audience reaches relevant content with reasonable effort.	"help each group find their relevant information"—each group's specific goals listed
1.5	Metro Health System	Hierarchy & Prioritization (S2)	Most important content elevated based on stakes.	"someone having a medical emergency" vs. "routine patient needs"—life-safety stakes
1.5	Metro Health System	Labeling Systems (S4)	Audience-appropriate labels; bilingual/plain-language where needed.	"English and Spanish-speaking communities," serving both medical professionals and lay patients
1.6	ShopLocal (Marketplace)	Organization Systems (S1)	Content organized around user roles when roles have distinct needs.	Three user types (customers, sellers, admin) with completely different functional needs
1.6	ShopLocal (Marketplace)	Labeling Systems (S4)	Plain language for non-technical users; jargon-free.	"older artisans who aren't tech-savvy," "accessibility standards"
1.7	Metro Daily News	Hierarchy & Prioritization (S2)	Important content elevated; reading contexts respected.	"breaking news and evergreen features discoverable"; three reading contexts (morning scanning, deep reading, social sharing)
1.7	Metro Daily News	Navigation Systems (S3)	Persistent navigation supports orientation across visits.	"readers get frustrated when they can't find yesterday's story"
1.8	City of Riverside (Government)	Organization Systems (S1)	Content categorized by user need, not by internal structure.	"organized by department rather than by actual needs"—prompt explicitly diagnoses bad organizational scheme
1.8	City of Riverside (Government)	Findability (S5)	Each audience reaches relevant content with reasonable effort.	"Citizens consistently tell us they can't find what they need"

Table 5: Tier 2 (Styling)

Task	Scenario	Principle	Definition	Signal in Prompt
2.1	Essence Jewelry	Visual Hierarchy (V1)	Size, weight, color, position create intentional reading order.	"minimalist brand values," "clean, modern, and sophisticated"
2.1	Essence Jewelry	Figure-Ground (V4)	Clear distinction between foreground and background.	"minimalist"—clarity depends entirely on figure-ground separation
2.2	Harmony Family Therapy	Similarity & Consistency (V3)	Consistent visual treatment across functional elements.	"feel safe, understood, and hopeful"—emotional consistency matters
2.2	Harmony Family Therapy	Color Contrast & Readability (V5)	Text meets contrast ratios; supports comfortable reading.	Stressed, potentially anxious users; "nervous about taking the first step"—need low-friction reading
2.3	RebelRoots (Plant-based Food)	Visual Hierarchy (V1)	Reading order resolves competing priorities.	"stand out dramatically while still appearing trustworthy"—two competing priorities
2.3	RebelRoots (Plant-based Food)	Color Contrast & Readability (V5)	Text remains readable despite expressive aesthetic.	"vibrant colors and edgy graphics" create high-contrast clashing risk and low-contrast readability risk
2.4	Luxe Verde (Sustainable Luxury)	Visual Hierarchy (V1)	Reading order balances competing brand messages.	"communicate luxury and sustainability simultaneously without either message diluting the other"
2.4	Luxe Verde (Sustainable Luxury)	Proximity & Grouping (V2)	Related content clustered; unrelated content spatially separated.	Luxury cues, sustainability credentials, and product information are distinct content needs
2.5	Global Ventures Inc.	Visual Hierarchy (V1)	Hierarchy clarifies parent-child relationships.	"brand hierarchy and coherence"—parent brand vs. 5 subsidiaries
2.5	Global Ventures Inc.	Similarity & Consistency (V3)	Consistent treatment unifies functionally similar elements.	"five distinct brand personalities under one corporate umbrella"
2.6	MedTech Solutions	Similarity & Consistency (V3)	Visual system robust enough to remain consistent across conditions.	"bright ER rooms, dimly lit radiology suites, and older hospital computers with poor displays"—must work across conditions
2.6	MedTech Solutions	Color Contrast & Readability (V5)	Text meets contrast ratios under constrained viewing.	"older hospital computers with poor displays," "12-hour shifts," "dimly lit radiology suites"
2.7	StreamVault (Streaming)	Proximity & Grouping (V2)	Related elements grouped; collections cohere.	Films, documentaries, international cinema by genre/mood/origin; solo vs. group viewing contexts
2.7	StreamVault (Streaming)	Figure-Ground (V4)	Foreground content distinct from background chrome.	"boutique cinema" aesthetic vs. "corporate streaming warehouse"
2.8	CreditBuilder (Fintech)	Proximity & Grouping (V2)	Distinct functional areas spatially separated.	Score data, action items, educational content, and account management are different functional areas
2.8	CreditBuilder (Fintech)	Color Contrast & Readability (V5)	Text readable under user and device constraints.	"varying literacy levels," "English is a second language," "older Android phones with small screens"

Table 6: Tier 3 (Functional)

Task	Scenario	Principle	Definition	Signal in Prompt
3.1	Bella Vista (Restaurant)	Visibility of System Status (F1)	System keeps users informed; progress visible.	Multi-step booking flow: date → party size → time → confirmation
3.1	Bella Vista (Restaurant)	Error Prevention & Recovery (F3)	Constraints and defaults prevent invalid input.	"seat 80 people across 20 tables," "dinner Tuesday–Sunday 5–10pm"—constraint-rich environment
3.2	EduConnect (Tutoring)	Visibility of System Status (F1)	Multi-step flow surfaces current state.	Multi-step funnel: search → browse → book trial → attend → book ongoing
3.2	EduConnect (Tutoring)	User Control & Freedom (F2)	Users can exit, undo, modify commitments.	Two-sided marketplace—both students and tutors need to manage their commitments
3.3	PhotoCommunity	User Control & Freedom (F2)	Content creators control their work and visibility.	"moderation to keep feedback constructive and quality high"—content creators need control over their work
3.3	PhotoCommunity	Consistency & Standards (F4)	Patterns predictable across modes.	"community engagement with commercial functionality"—two modes in one platform
3.4	FreelanceFlow	Error Prevention & Recovery (F3)	Constraints prevent costly errors in high-stakes environment.	"$500 logos to $50,000 website builds"—high financial stakes; "international payments"
3.4	FreelanceFlow	Consistency & Standards (F4)	Patterns predictable across user types and project sizes.	"Fortune 500 companies and small business owners" using the same platform
3.5	ProjectMaster Enterprise	Visibility of System Status (F1)	System surfaces health, risk, and progress at every level.	"real-time executive dashboards," "budget tracking," "risk management"
3.5	ProjectMaster Enterprise	Flexibility & Efficiency of Use (F5)	Interface adapts to range of user expertise and roles.	"individual contributors tracking tasks to C-suite executives monitoring portfolio performance"; "integration with existing enterprise tools"
3.6	SkillBridge (Learning)	User Control & Freedom (F2)	Granular privacy and visibility controls.	"Privacy is crucial—professionals want to control what colleagues can see about their learning goals"
3.6	SkillBridge (Learning)	Error Prevention & Recovery (F3)	Smooth correction paths for common mistakes.	"Users frequently make mistakes like posting to wrong forums or enrolling in wrong courses, so we need smooth correction paths"
3.7	MoneyWise (Finance)	Error Prevention & Recovery (F3)	Trust-preserving design prevents costly mistakes.	"wary after data breaches," "anxiety around finances," "rebuilding after bankruptcy"
3.7	MoneyWise (Finance)	Flexibility & Efficiency of Use (F5)	Interface adapts to financial literacy and user range.	"college students with simple checking accounts to families juggling mortgages, investments, and college savings"; "dyscalculia or other numerical processing challenges"
3.8	EventSpace (Virtual Events)	Visibility of System Status (F1)	System surfaces critical state for organizers, speakers, attendees.	"Speakers need confidence their presentation won't fail mid-talk"; "missing key content due to time zone confusion"
3.8	EventSpace (Virtual Events)	User Control & Freedom (F2)	Users recover from common mistakes mid-event.	"attendees joining wrong sessions," "leaving sessions accidentally," "missing recording permissions"