

Workflow-Net: Toward Understanding Designer Workflows in Generative AI-Driven Systems through Comparing Node-and Prompt-Based Interfaces

Tae Hee Jo^{1,4}, Jiin Choi^{1,4}, Semin Jin^{1,4}, Seung Won Lee^{1,4}, Yugyeong Jang^{2,4}, Sang Woon Park⁵, Seonghoon Ban⁵, Kyung Hoon Hyun^{3,4*}

¹Department of Interior Architecture Design, Ph.D. Student, Hanyang University, Seoul, Korea

²Department of Interior Architecture Design, Master's Student, Hanyang University, Seoul, Korea

³Department of Interior Architecture Design, Professor, Hanyang University, Seoul, Korea

⁴Human-Centered AI Design Institute, Hanyang University, Seoul, Korea

⁵RECON Labs Inc., Seoul, Korea

Abstract

Background Generative artificial intelligence (AI) is increasingly integrated into design practice, yet how these generative AI-driven design support system interface frameworks shape the designers' workflow remains underexplored. To address this gap, we formalized and compared three generative AI-driven interfaces: Agentic Prompt-Based (AB), Programming Node-Based (PB), and Creative Node-Based (CB). Studying their influence requires methods that capture workflow dynamics beyond micro-level actions. Existing approaches such as linkography or workflow graphs (W-graphs) focus on words, concepts, or artifacts, limiting the analysis of high-level actions and cross-user patterns.

Methods This study introduces Workflow-Net, a novel evaluation method that uses large language models (LLMs) to cluster structured protocol data based on semantic intent to map high-level design actions and to aggregate individual workflows into a comprehensive, weighted directed graph. A within-subject user study was conducted with nine designers, where participants performed three distinct design tasks across all three interfaces to capture multi-user, cross-interface behavioral patterns.

Results Findings show that interface frameworks do not merely support design but strongly influence designers' behavior by structuring the cognitive arc. AB supported initial ideation but limited refinement due to text abstraction gaps and a lack of iterative detail control. PB offered precision and granular control but enforced rigid linearity, extreme convergence, and the highest cognitive load. In contrast, CB best supported the creative process by enhancing designer agency and satisfaction, effectively balancing exploration with refinement through automated process traceability.

Conclusions This study establishes that the interface framework is a structural determinant of the creative workflow. While AB and PB interfaces impose significant tradeoffs through abstraction gaps or high cognitive load, the CB interface emerged as a balanced model that fosters higher designer agency. Beyond evaluation, the Workflow-Net methodology offers a foundation for developing future hybrid, agent-assisted co-creative partners that adapt to the fluid dynamics of the designer's workflow.

Keywords Design Research, Designer Workflow, Design Process Analysis, Interface Framework, Node-based Interface, Generative AI

Tae Hee Jo and Jiin Choi contributed equally to this work.

This work was supported by the Technology Innovation Program (RS-2025-02317326, Development of AI-Driven Design Generation Technology Based on Designer Intent) funded by the Ministry of Trade, Industry & Energy(MOTIE, Korea), and supported by the research fund of Hanyang University(HY-20250000003838).

*Corresponding author: Kyung Hoon Hyun (hoonhello@hanyang.ac.kr)

Citation: Jo, T. H., Choi, J., Jin, S., Lee, S. W., Jang, Y., Park, S. W., Ban, S., & Hyun, K. H. (2026). Workflow-Net: Toward Understanding Designer Workflows in Generative AI-Driven Systems through Comparing Node-and Prompt-Based Interfaces. *Archives of Design Research*, 39(2), 7-34.

<http://dx.doi.org/10.15187/adr.2026.05.39.2.7>

Received : Mar. 18. 2026 ; **Reviewed :** Apr. 13. 2026 ; **Accepted :** Apr. 22. 2026
pISSN 1226-8046
eISSN 2288-2987

Copyright : This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>), which permits unrestricted educational and non-commercial use, provided the original work is properly cited.

1. Introduction

Designers' creative design process is a complex and dynamic behavioral workflow, characterized by continuous iteration between exploration (divergence) and refinement (convergence) (Choi et al., 2025; You et al., 2025). These sequences of design moves reveal not just the output, but the underlying intent behind their design action decisions and inspiration sources (Goldschmidt, 2014; Kwon et al., 2023). While Generative AI integration in design systems offers powerful capabilities (D. C. E. Lin et al., 2025; Duan et al., 2025; Lawton et al., 2023), human-computer interaction (HCI) research has largely focused on the tools themselves (Lee et al., 2024; Shen et al., 2025; J. Zhou et al., 2024) rather than the interface frameworks (e.g. conversational versus canvas-based) through which they are accessed. Consequently, how these fundamental structures shape the creative design process workflows remains largely underexplored.

Current analysis methods face distinct limitations in capturing these dynamics. Linkography (Goldschmidt, 2014; Hatcher et al., 2018) focuses on micro-level actions (e.g. tracing links between individual words and sentences), making it difficult to aggregate high-level patterns across users. Conversely, computational methods such as W-graphs (Chang et al., 2020) cluster workflow states based on geometric similarity. This artifact-centric approach fails to capture designers' action intent; distinguishing for instance, whether a user intends to "refine a design feature" or "branch into varying designs". Thus, current methods lack a systematic method to compare how different interfaces facilitate or constrain the creative design process workflows by identifying cross-user behavioral patterns or interface-specific tendencies.

To bridge this gap, the study addresses the following question: How do distinct interface frameworks shape designers' creative design process workflows? To answer this, we introduce Workflow-Net, a novel evaluation method that maps designers' meaningful high-level action-centric behaviors into a comprehensive workflow graph, enabling robust multi-user cross-interface analysis. Therefore, this study involves the following tasks: 1) formalizing and modeling three distinct generative AI-driven interface frameworks (AB, PB, and CB); 2) implementing Workflow-Net, a novel evaluation method combining protocol analysis with Large Language Models (LLMs) to cluster and map high-level design behaviors; 3) constructing aggregated workflow graphs to visualize cross-user interface-specific patterns; 4) conducting a comparative user study to capture workflow dynamics; and 5) analysis to empirically compare the impact of each interface on the creative design process.

2. Related Works

2.1. Interface Frameworks for Generative AI-Driven Design Support Systems

The interface is the primary space where designers communicate their intent with generative AI. While recent HCI surveys broadly categorize these interface frameworks into Conversational and Canvas User Interfaces (UIs) (Luera et al., 2024), distinct sub-framework

structures these interactions in fundamentally different ways, significantly shaping the designer’s workflow and sense of agency. We conceptualized the two UIs as three distinct frameworks—an Agentic Prompt-based (AB) interface, a Programming Node-based (PB) interface, and a Creative Node-based (CB) interface. Our rationale for this classification is as follows.

2. 1. 1. Conversational (prompt-based) UIs

Characterized by natural language interaction, the AB interface framework positions the generative AI as a dialogue partner. This structure is classified as a “Conversational UI”, where most interactions occur in a prompt box as part of a turn-based exchange (Luera et al., 2024). Commercial systems such as Lovart AI (Resonate International INC., 2025) exemplify this approach, featuring a central chat panel where users direct an AI agent to generate and modify content within a creative workspace. Such interface frameworks facilitate iterative feedback as users articulate goals and collaboratively shape outcomes with AI.

While accessible, relying solely on conversation presents significant usability challenges. The difficulty of crafting effective prompts is a well-documented issue in HCI, often described as a trial-and-error process where users struggle to discover the specific phrasing a model will understand (Dang et al., 2022; Drosos et al., 2025; Mondal et al., 2024). This creates an abstraction gap between a user’s high-level intent and the concrete, detailed language the model requires to produce a satisfactory result (Liu et al., 2023). These challenges specifically include:

- **Difficulty Specifying Intent:** Language is often ambiguous for visual tasks, leading users to struggle with precisely referring to objects or locations. This results in verbose prompts and a frustrating trial-and-error process (Caetano et al., 2025; Masson et al., 2024) rather than effective design exploration.
- **Lack of Granular Control:** A single prompt often modifies the entire output, making localized edits difficult without unintended side effects. Users report a lack of control and find it hard to verify that only the desired changes were made (Chung & Adar, 2023; Masson et al., 2024).
- **Poor Reusability:** Prompts are typically designed for single interactions. Reusing a prompt for a new object requires manual text editing, which is inefficient (Desmond & Brachman, 2024; Masson et al., 2024), especially for iterative design workflows.

To mitigate these issues, HCI research has explored blending conversational interfaces with direct manipulation. For instance, DirectGPT (Masson et al., 2024) is a UI layer that allows users to select objects on a canvas to localize a prompt’s effect or drag-and-drop objects directly into a prompt to create unambiguous references. These developments demonstrate a clear need to move beyond purely conversational models to support the nuanced, granular control required in complex design tasks.

2. 1. 2. Canvas (node-based) UIs

In contrast to conversational UIs, node-based interfaces fall under the broader category of “Canvas UI”, characterized by a central canvas area that houses the primary content, with generative and secondary tools often located in the periphery (Luera et al., 2024). These

node-based interfaces offer a more visual and structured approach to managing generative AI-driven design process workflows (You et al., 2025) by representing operations and assets as nodes on an infinite canvas. Within this category, we distinguish between two fundamentally different approaches: Programming Node-Based and Creative Node-Based interfaces.

PB interfaces are modeled after visual scripting tools like ComfyUI (Comfy Org, 2024), offering users granular and procedural control over generative AI processes. In this interface framework, users manually construct a complete computational workflow pipeline, often structured as a Directed Acyclic Graph (DAG) (Xue et al., 2025), by connecting operational nodes and configuring their internal parameters. This approach has been widely adopted across professional domains—such as computational modeling with Grasshopper (Robert McNeel & Associates, n.d.), multimedia with Max/MSP (Cycling '74, n.d.), and game logic with Unreal Blueprints (Epic Games, n.d.)—allowing for the creation of complex pipelines without deep programming knowledge.

This structure offers precision and repeatability, making it powerful for tasks requiring detailed adjustments and multi-step generative processes (You et al., 2025). A representative example is ComfyUI (Comfy Org, 2024), which leverages Stable Diffusion (Zhang et al., 2023) for image generation by allowing designers to orchestrate specific functions such as VAE for encoding (Han et al., 2019), CLIP for textual conditioning (Ramesh et al., 2022), and ControlNet (Zhang et al., 2023) for fine-grained control. However, the strength of this interface framework introduces significant challenges in creative contexts:

- **High Entry Barrier and Cognitive Load:** Achieving desired results requires substantial knowledge of node functions and parameter configurations. The need to manually construct workflows from a blank canvas is laborious and cognitively demanding (Jiang, 2023; Xue et al., 2025), prompting research into AI assistance like InstructPipe (Z. Zhou et al., 2024) to generate initial pipelines from natural language to lower this barrier.
- **Interaction Cost:** Achieving complex wiring can be taxing, leading to explorations of alternative interaction models such as “positional control” (Jiang, 2023), where a node’s spatial position explicitly determines its execution order to reduce interaction costs.
- **Fixation and Translation Overhead:** Designers often spend excessive time translating expressive semantic intents into rigid programming structures (You et al., 2025) rather than pursuing emergent creative shifts. Because these workflows are often task-specific, creating new variations for broad exploration remains a time-consuming obstacle.

In contrast to the procedural nature of PB interfaces, CB interfaces offer a fluid, content-centric framework focused on exploration and process traceability. In these systems, nodes represent self-contained design assets or creative states, such as images, videos, ideas, or code sketches arranged on an infinite canvas (Luera et al., 2024; You et al., 2025). The connections between nodes are often generated automatically as a result of user actions, creating a visual history of the workflow process (Chen et al., 2025).

This structure inherently supports the non-linear workflows essential for creativity, allowing designers to easily branch, backtrack, and compare parallel explorations without the high overhead of manual pipeline construction (Zhong et al., 2024). Commercial tools like Flora AI (FLORA, 2025) and GENPRESSO AI (Recon Labs Inc., 2025) exemplify this approach, enabling users to generate new content by linking or acting upon existing asset nodes. HCI research has further explored the potential of this interface framework:

- **Idea Tracking:** IdeationWeb (Shen et al., 2025) uses a CB interface to help users track the evolution of structured design ideas, reducing the cognitive load of linear, chat-based brainstorming.
- **Intuitive Control:** Spellburst (Angert et al., 2023) functions as an intuitive version control system where each node is a complete sketch modified through prompts, semantic sliders, or direct code editing.
- **Agent-Powered Assistance:** DesignManager (You et al., 2025) employs a CB interface as the foundation for an agent-powered copilot, where the visual workflow provides context for an AI agent to offer process-aware assistance across multi-tool design workflows.

In this interface framework, the canvas functions as a critical communicative space where inputs and outputs are visualized to facilitate clearer interaction. By structuring both inputs and outputs as nodes, CB interface frameworks support intuitive editing and allow users to explicitly visualize the generative history, helping them trace and regulate their design processes over time (Chen et al., 2025).

2. 2. Creative Design Process Workflow Analysis

To understand how these interface frameworks influence behavior, we must look beyond static outputs and analyze the design process workflow itself. While existing methods are valuable, they present distinct limitations in capturing high-level design action intent.

Micro-Level vs. Macro-Level Analysis. Linkography (Goldschmidt, 2014; Hatcher et al., 2018) is a foundational method for analyzing moment-to-moment evolution of the thought process, mapping the links between design moves. However, its strength is also its limitation; it focuses on tracing micro-level connections between the individual words, sentences, or concepts (Hatcher et al., 2018). While effective for isolated design case studies, this granularity makes it difficult to systematically capture the high-level actions that represent designers' underlying intent, such as "verifying a design's feasibility", "refining a design feature form", or "branching into varying designs". Furthermore, this approach makes it challenging to aggregate workflows across multiple users or identify common behavioral or interface-specific patterns (Smith et al., 2025).

Artifact-Centric vs. Action-Centric Analysis. To address aggregation, W-graphs (Chang et al., 2020; Hyun & Jang, 2025; Jang & Hyun, 2024) captures the overall sequence and collective patterns across multiple designers' design process by clustering intermediate states based on the geometric similarity of the 3D output. While effective for revealing 'what was made', this artifact-centric approach fails to capture and distinguish the context of the user's action intents and behavioral patterns regarding 'why it was done that way'. For instance, "refining a curve" and "correcting an error" might produce similar geometric changes but represent fundamentally different creative behaviors.

Bridging the Gap with LLMs. To overcome these limitations, recent studies demonstrate that LLMs can effectively group detailed work data (e.g., event logs) into semantic units and label them automatically (Fani Sani et al., 2023; Vaarandi & Bahşi, 2025). These studies are significant in that they can translate raw records into contextually understandable explanations for humans. Moreover, recent research has also proposed methods for semantic-based clustering using LLMs on various data (e.g., news articles, short texts, conversation logs) (Miller & Alexander, 2025; Tarekegn, 2024), demonstrating the possibility to derive common patterns that reflect contextual meaning, rather than just listing data.

Building on this, the current study proposes an action-centric approach. By leveraging LLMs to cluster structured protocols based on semantic intent rather than just geometric output or keywords, we map meaningful, high-level design action behaviors. Validated by Clustering Stability (CLS) metrics (von Luxburg, 2010), this method constructs a comprehensive multi-user workflow graph that reveals how different interface frameworks systematically shape the creative design process beyond the constraints of artifact-centric and individual, micro-level analyses.

3. Method

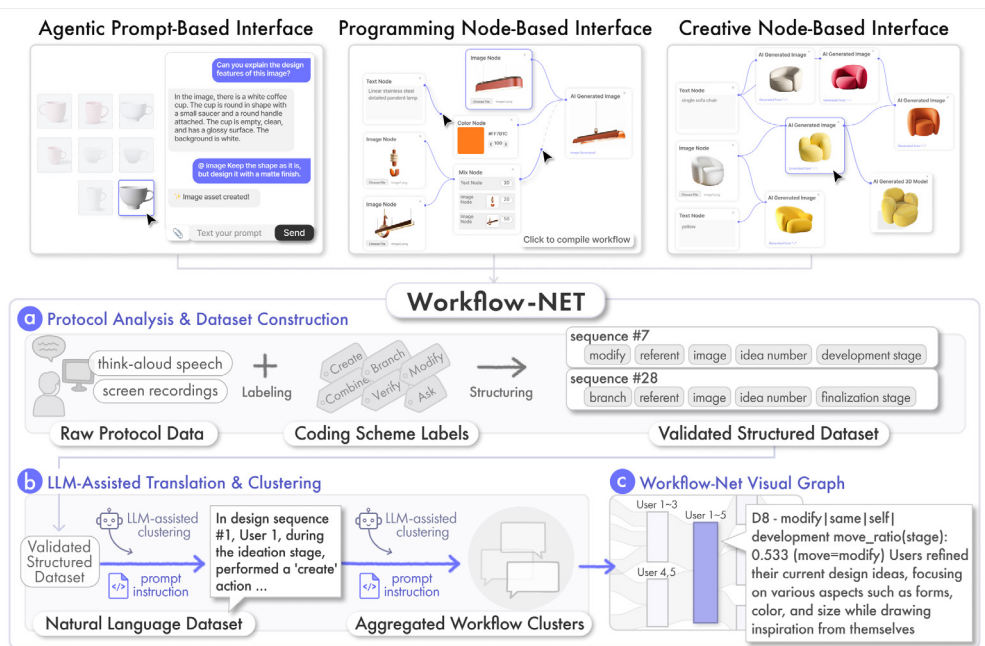


Figure 1 Overview of our Workflow-Net framework

3. 1. Overview

The proposed method aims to empirically compare how distinct generative AI-driven interface frameworks shape the behaviors of the creative design process workflows. The research methodology undergoes the following stages: 1) Modeling Interface Frameworks: Development of the distinct AB, PB, and CB interaction models to serve as the comparative apparatus. 2) Workflow-Net Analysis: Our evaluation pipeline that captures high-level design intent through deductive protocol analysis approach (Ericsson, 2017) followed by LLM-assisted in-context clustering (Fu et al., 2025) to aggregate multi-user data. 3) Graph Construction: The transformation of multi-user sequences into a comprehensive, weighted directed graph to visualize collective behavioral patterns and state transitions. 4) Quantitative Evaluation: The derivation of behavioral metrics from the analysis to empirically measure the impact of interface frameworks on the creative design process workflows.

3. 2. Modeling Generative AI-Driven Interface Frameworks

To investigate how interaction modalities shape design workflows, we developed three web-based generative AI-driven design support interfaces connected to a unified backend architecture (Figure 5). To ensure a rigorous, controlled comparison, this backend exposes a consistent suite of generative agents to all three frontends: *text-to-text*, *image-to-text*, *text-to-image*, *image-to-image*, *text-to-3D*, and *image-to-3D*. As detailed in Section 3.2.2, this parity ensures that while the interaction framework (AB, PB, and CB) vary, the underlying generative power remains constant, allowing us to isolate the impact of the interface itself on the creative process.

3. 2. 1. Frontend User Interface Implementations

AB interface: Modeled after conversational UIs, this interface features a chat panel where users input natural language prompts and optional attachments (Figure 2). The system parses text input and places automatically generated assets as cards onto the canvas, prioritizing dialogue over direct manipulation to articulate design goals. The goal of this interface is to allow users a fluid, iterative workflow based on natural language, minimizing the need to learn complex controls:

- The user types a command into the chat panel such as '@image a vintage wooden chair' or '@3d minimalist urban mug cup'.
- The user can optionally attach images to their chat prompts or select existing asset cards on the canvas to provide multimodal context.
- The agent parses the text for commands and context, packages the multimodal inputs, and executes the relevant generative task.
- Upon completion, the newly generated asset (image or 3d model) is automatically placed as cards onto the canvas space, allowing for immediate user-led manipulation and spatial organization.

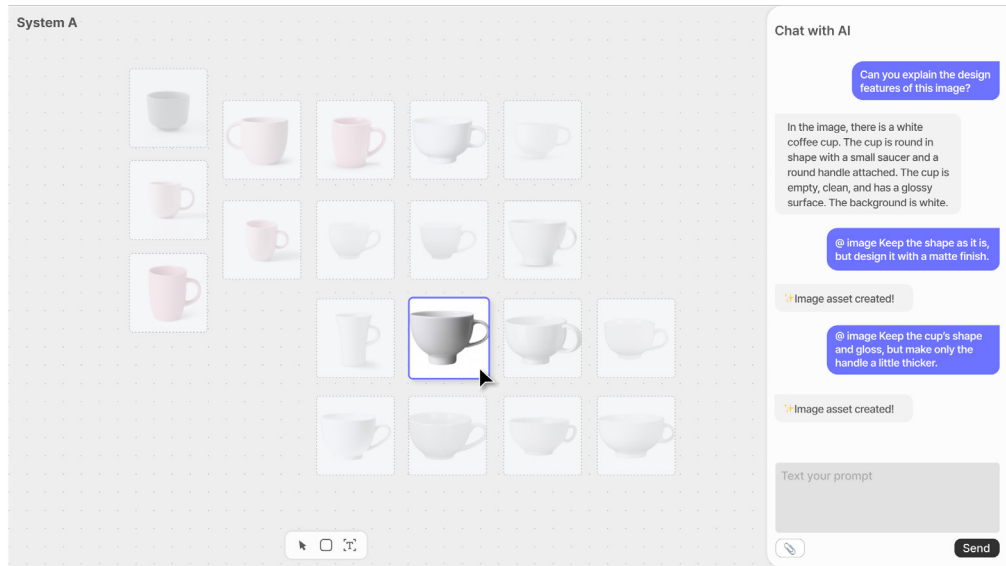


Figure 2 Frontend UI of our developed Agentic Prompt-based (AB) Interface – Sample of P5’s cup design task

PB interface: Modeled after visual scripting environments, this interface requires users to manually construct DAGs (Figure 3) (Xue et al., 2025) by connecting logic nodes (inputs, parameters, outputs). This interface framework grants the designer explicit, granular control over the entire generative workflow by requiring them to construct a complete workflow of operations. The goal of this interface is to allow users precision, repeatability, and a clear visualization of the data flow in the generative process. The workflow is structured around the construction of DAG, which visually represents the entire computational pipeline:

- The user populates the canvas with various nodes from a sidebar representing specific operations.
- They configure each node’s internal parameters and draw explicit connections between the output and input sockets to define the data flow.
- A global ‘*compile*’ button triggers the execution of the workflow when clicked; the system traverses the completed workflow, from the terminal output node backwards to assemble and run the full chain of operations.

As noted in prior work, users must manually create these workflows, which can be laborious and time-consuming (Xue et al., 2025).

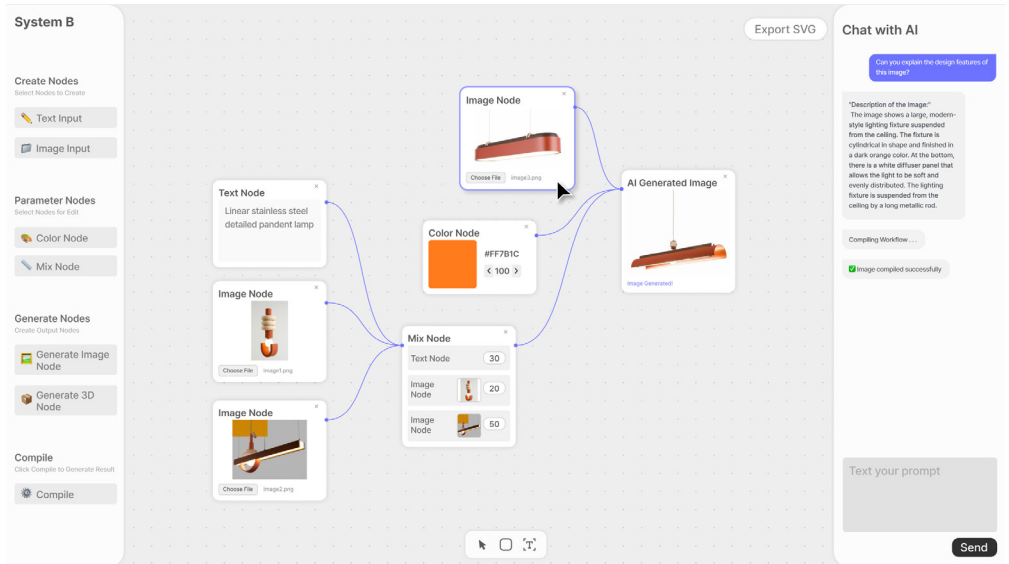


Figure 3 Frontend UI of our developed Programming Node-based (PB) Interface – Sample of P3’s ceiling lamp design task

CB interface: This interface streamlines interaction by automating connections through more direct, selection-based interaction (Figure 4), lowering the cognitive load of visual programming. When a user acts on a content node, the system automatically generates a new linked node, creating a visual history tree without requiring manual logic wiring:

- The user populates the canvas with self-contained content nodes (e.g., text, image).
- To generate new content, the user selects one or more existing nodes that serve as inputs.
- They then click a direct action button (e.g., *Generate Image*, *Generate 3D Model*), which the system immediately creates a new output node on the canvas and automatically draws connections back to the input nodes that were used for its content generation.

This interface, by automating the creation of connections, visually documents the designer’s complete generative design process history, enabling process traceability.

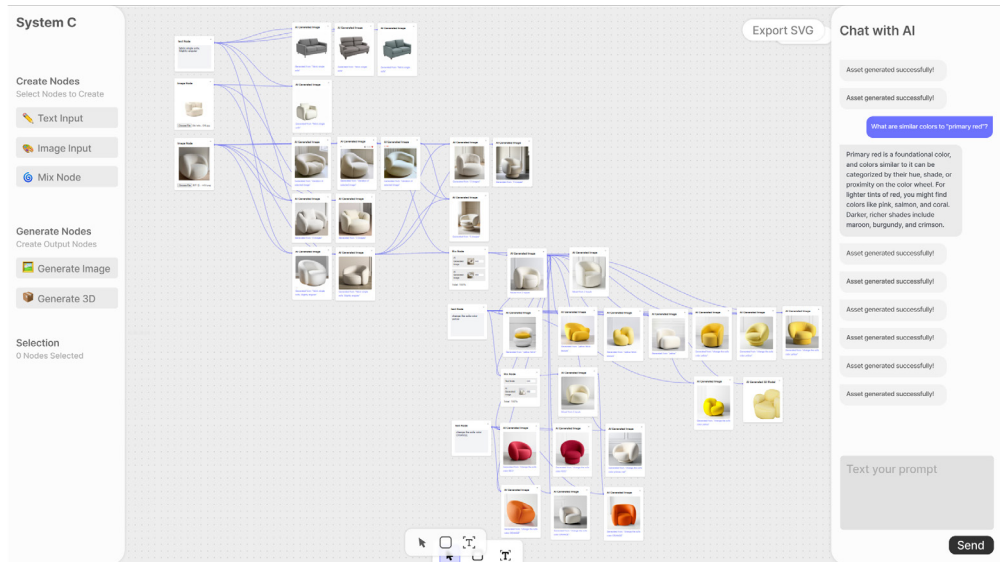


Figure 4 Frontend UI of our developed Creative Node-based (CB) Interface – Sample of P2’s chair design task

3. 2. 2. Backend Technical Implementations

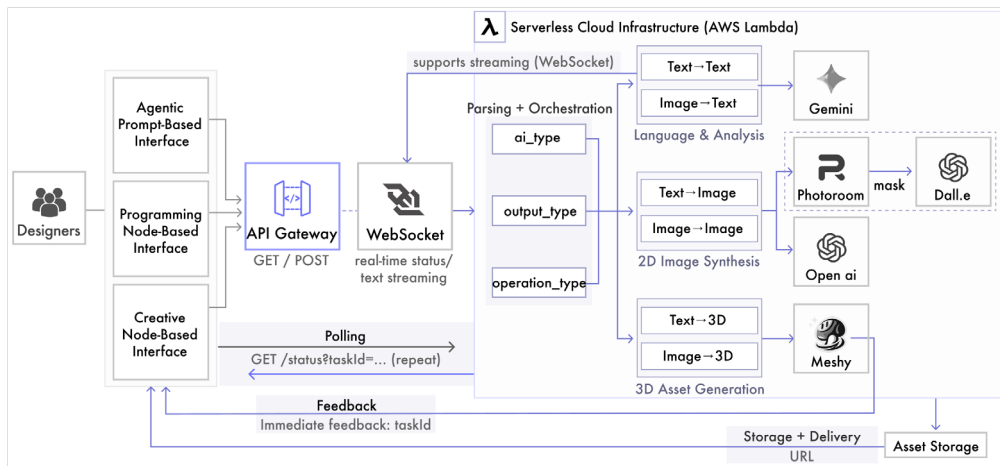


Figure 5 Frontend to backend architecture framework for the generative AI capabilities

To ensure a controlled comparison, all three interfaces were connected to a single backend system built on a scalable serverless architecture + that functions as an orchestration layer, utilizing a unified API gateway to receive and process requests from the frontend interfaces (Figure 5). The gateway parses a set of parameters (Table 1) to dynamically route tasks to the appropriate generative agent.

Table 1 List of key parameters and descriptions required by the backend endpoint

| Parameter | Parameter Labels | Description |
|-----------------------|---|--|
| <i>ai_type</i> | <i>openai, gemini, meshy, photoroom</i> | Specifies the AI provider to use. |
| <i>output_type</i> | <i>text_only, image, model3d</i> | Defines the desired output format. |
| <i>operation_type</i> | <i>generate, edit, analyze, variation</i> | Details the specific design action to perform. |

Generative AI Functionalities. The backend architecture was engineered to provide six distinct generative AI functions, which were made equally available to all our three interfaces through the unified API endpoint. The logic for invoking these functions followed a consistent pattern of parameterization and data handling defined by the backend’s orchestration layer:

1. *Text-to-Text* Generation: this facilitates general-purpose textual queries. When a request is made with an *output_type* of *text_only* and *operation_type* of *generate*, the backend routes the user’s prompt to Google’s Gemini (Google, n.d.). The interfaces are equipped to handle both standard JSON responses and real-time streaming of text, allowing for immediate feedback during longer text generations.
2. *Text-to-Image* Generation: this is responsible for creating images from textual descriptions. The interface sends a request with an *output_type* of *image* and an *operation_type* of *generate*. The backend orchestrator forwards the prompt to a text-to-image model. The interface utilizes OpenAI’s DALL-E 3 model (OpenAI, 2025) for high-quality generation, with Google’s Gemini model (Google, n.d.) as an alternative. To ensure stylistic consistency, the interface’s logic was designed to append a string to user prompts (e.g., “Please generate a professional product photograph of a single product with a clean white background. The product should be positioned at a three-quarter view. Ensure the lighting is even and the focus is sharp.”) before sending the request. The backend returns the final image as a *base64 string*.
3. *Text-to-3D* Generation: this exclusively utilizes the Meshy API (Meshy, n.d.) (*ai_type*: ‘*meshy*’) with an *output_type* of *model3d* and *operation_type* of *generate*. Due to the extended processing time, it relies on the aforementioned backend’s asynchronous workflow. The interface sends the user’s prompt along with numerous Meshy-specific parameters (e.g., *art_style, enable_pbr, target_polycount*). The backend initiates the task with Meshy and immediately returns a *taskId*. The interface then periodically polls the API using this *taskId* to fetch status updates until the generation is complete and download links for the 3D model files (.glb, .fbx) are available.
4. *Image-to-Text* Analysis: to enable the analysis of visual content, this is invoked when the *output_type* is *text_only* and the *operation_type* is *analyze*. The interface transmits a *POST* request containing the image as an *image_base64 string* in the request body. The backend then passes this image data, along with an optional guiding prompt (provided by the user via the chat panel or a connected text node), to Google’s Gemini model (Google, n.d.). If no text is provided, the model utilizes its native vision capabilities to return a general textual description; otherwise, it returns an answer to a specific query about its visual content.

5. *Image-to-image* Generation: this encompasses more complex visual transformations and was implemented through a multi-step process orchestrated by the interface system logic:
 - Variation: for generating a variation of a single source image, the *operation_type* is set to *variation*. The interface ensures the input image is converted to a compatible RGBA PNG format before sending it to OpenAI's DALL-E model (OpenAI, 2025). This specific operation is text-free; the model utilizes the input image's visual embedding as the sole conditioning signal to generate a stylistically and structurally similar output.
 - Edit: when a user provides a source image and a textual modification prompt (e.g., "*add a handle*"), the interface initiates a three-step pipeline:
 - A. Background removal: the original image is first sent to the PhotoRoom (Photoroom, n.d.). This isolates the main subject of the image.
 - B. Mask creation: the interface then processes the background-removed image from PhotoRoom to create a precise, opaque mask (*mask_base64*). This mask defines the exact area of the original image that DALL-E is allowed to modify.
 - C. Image Generation: Finally, the interface sends the original source image (converted to RGBA format), the newly created mask, and the user's textual prompt to OpenAI's DALL-E (OpenAI, 2025) editing endpoint. This ensures that modifications are applied only to the intended areas, preserving the rest of the image.
6. *Image-to-3D* Generation: Similar to *text-to-3D*, this process is powered by the Meshy API (Meshy, n.d.), with the interface configured to use the Meshy-5 model (*ai_model: 'meshy-5'*) by default. The function is invoked with an *output_type* of *model3d* and an *operation_type* of *image_to_3d*, requiring an *image_base64* payload. Similar to the variation operation, this process does not require a text prompt; the Meshy-5 model infers the 3D geometry and texture directly from the provided 2D visual data. It follows the same asynchronous *taskId*-based workflow, where the interface tracks the task's status and retrieves the final 3D model files upon completion.

3. 3. Workflow-Net: Multi-user Action-centric Analysis

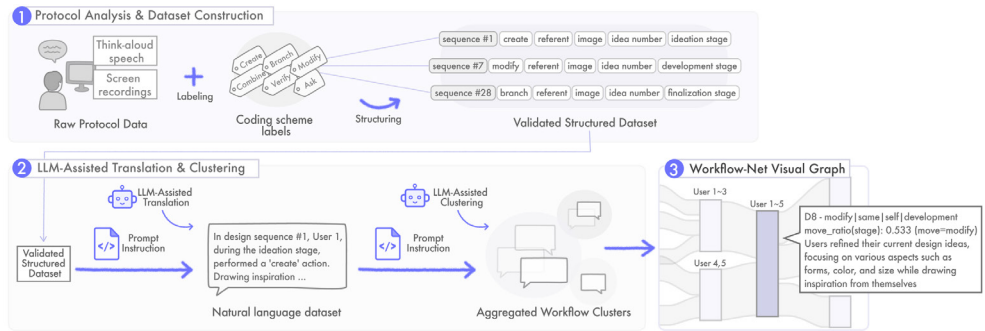


Figure 6 Illustration of our multi-step Workflow-Net framework: 1) Protocol Analysis & Dataset Construction, 2) LLM-Assisted Translation & Clustering, 3) Workflow-Net Graph Visualization.

We developed a multi-step methodology (Figure 6) to evaluate and quantify designers’ behaviors by mapping high-level design action data into a comprehensive graph.

Step 1. Protocol Analysis and Dataset Construction. To capture the high-level intent behind design moves, we employed deductive protocol analysis (Ericsson, 2017; Ericsson & Simon, 1980). Researchers systematically coded designers’ think-aloud verbalizations and observations against a predefined coding schema containing six high-level *design action moves* (*create, branch, modify, combine, verify, ask*) and eight *contextual parameters* (e.g., *User, Referent, Referent property, Inspiration Source, Design Idea*).

Our predefined coded scheme labels were developed by synthesizing and adapting established frameworks from foundational design research. The main category, ‘*design action move*’ is justified by its functional role in the creative process as identified in prior HCI and design studies (Lee et al., 2024; Shen et al., 2025; You et al., 2025). Furthermore, to provide a richer understanding of the designer’s intent, each design action move is contextualized by several other observable parameters, including the *Referent, Referent Property, Inspiration Source* (self-creativity vs. AI-inspiration) (X. Lin et al., 2025), *Design Idea Input Number* and *Output Number*, and *Design Stage*. While these parameter categories were predetermined by our schema, the specific labels assigned to them were dynamically extracted based on the unique, real-time context of each user’s actions. As a result, our protocol coding scheme enables the sequential mapping of each designer’s intended actions into a structured dataset (Figure 6-1). Unlike purely inductive methods, this ensures actions are mapped to functional roles grounded in design theory.

This coding scheme was applied by the experiment moderators who systematically reviewed each participant’s think-aloud transcriptions and screen recordings. During this manual process, moderators extracted the relevant labels for every observed action. Critically, this phase also involved inferring the user’s underlying motivation (Action Motive) for each design move sequence, a judgment grounded in direct evidence from the qualitative data. This procedure served as our primary human validation stage, ensuring the final structured dataset was an accurate representation of the design process and formed a reliable foundation for the subsequent analysis.

Step 2. LLM-Assisted Translation and Clustering. To aggregate individual workflows, we developed a pipeline (Figure 6-2) to convert these labeled codes into a unified workflow.

While our protocol analysis produced a validated, structured dataset, the coded labels are not inherently suitable for nuanced, LLM-driven workflow analysis. To address this, we developed a process to translate the structured data into rich, natural language descriptions that preserve the original, human-validated context.

We evaluated two distinct prompt engineering strategies to perform this translation. The first was an unconstrained approach, where the LLM was instructed to generate a natural language summary from the data points. The second was a template-based approach, which required the model to populate a predefined sentence structure. To validate the output of both methods, we implemented two verification processes. We conducted a rule-based automated check on the entire translated dataset. This sanity check systematically verified that all key data parameters from each source row were present in the corresponding output sentence. For any translations that failed this check, moderators then performed manual spot-checks to diagnose those errors.

Our evaluation revealed that the unconstrained approach consistently failed both the automated and manual checks, frequently omitting critical labels from the final translation. In contrast, the template-based approach achieved 100% accuracy on both the automated validation and manual spot-checks, successfully including all labels in every instance. Therefore, we exclusively adopted the template-based method for our final data translation. The selected technique involved systematically presenting each row of the coded dataset to the GPT-4o model (OpenAI, n.d.). The model was instructed to synthesize the discrete data points into a coherent narrative sentence based on a predefined template.

Following the high-fidelity translation of coded labels into natural language, we then employed LLM-assisted in-context clustering (Fu et al., 2025) to group design actions from multiple users based on semantic intent. This aggregation step is designed to overcome the limitations of prior methods by enabling the systematic comparison of cross-user behavioral patterns based on high-level design action moves. The clustering logic prioritized design moves and design idea relations while enforcing hard constraints against mixing different design stages. To ensure the method's clustered outputs were not arbitrary but consistently reproducible, we conducted a rigorous stability analysis.

The validation process involved setting the model's decoding temperature to 0.5, a value chosen to balance determinism with semantic flexibility, and executing the clustering process five times on each identical input datasets. The consistency of the five resulting partitions per interface (AB, PB, CB) was then assessed using the Clustering Stability (CLS) method (von Luxburg, 2010), with detailed results presented in Table 2.

Table 2 Clustering Stability Scores for AB, PB, and CB Interfaces

| Interface | T1 | T2 | T3 | T4 | T5 |
|-----------|------------------------------------|--------|-------------------------|--------|--------|
| AB | Clustering Stability Matrix | | | | |
| T1 | 1.0000 | 0.8869 | 0.7882 | 0.8383 | 0.7964 |
| T2 | | 1.0000 | 0.7383 | 0.7872 | 0.7193 |
| T3 | | | 1.0000 | 0.8449 | 0.8161 |
| T4 | | | | 1.0000 | 0.8952 |
| T5 | | | | | 1.0000 |
| AB Stats | Mean: 0.8111 | | SD (σ): 0.0546 | | |
| PB | Clustering Stability Matrix | | | | |
| T1 | 1.0000 | 0.9148 | 0.9237 | 0.8230 | 0.7863 |
| T2 | | 1.0000 | 0.9268 | 0.7799 | 0.7408 |
| T3 | | | 1.0000 | 0.7799 | 0.7906 |
| T4 | | | | 1.0000 | 0.7243 |
| T5 | | | | | 1.0000 |
| PB Stats | Mean: 0.8190 | | SD (σ): 0.0720 | | |
| CB | Clustering Stability Matrix | | | | |
| T1 | 1.0000 | 0.9642 | 0.8294 | 0.8089 | 0.7949 |
| T2 | | 1.0000 | 0.8617 | 0.8404 | 0.8266 |
| T3 | | | 1.0000 | 0.9756 | 0.9593 |
| T4 | | | | 1.0000 | 0.9551 |
| T5 | | | | | 1.0000 |
| CB Stats | Mean: 0.8816 | | SD (σ): 0.0691 | | |

The analysis demonstrated a high degree of stability across all three interfaces (AB=0.81(σ =0.05), PB=0.82(σ =0.07), CB=0.88(σ =0.07)). This outcome validates our methodology as a robust tool for this analytical task, confirming our methodology's reliability for cross-user analysis.

Step 3. Graph Construction. To model the sequential dynamics, we utilized NetworkX (Hagberg et al., 2007) to construct a directed graph of the aggregated workflow of multiple users, where nodes represent semantic action clusters and edges represent sequential transitions (Figures 10 to 12). The weight $w(u, v)$ of an edge is calculated as the total frequency of transitions from state u to an action state v across all designer sessions D , formally expressed as:

$$w(u, v) = \sum_{L \in D} \sum_{l=1}^{|L|-1} I(s(a_l) = u \wedge s(a_{l+1}) = v)$$

where L is a single designer's sequence, $s(a_i)$ maps an action to its corresponding state, and $I(\cdot)$ is the indicator function. This topology reveals the mapping of the dominant behavioral patterns and cognitive flow across interfaces.

3. 4. Quantifying Behavioral Metrics using Workflow-Net

To empirically compare the interfaces, we derived four metrics from the graph topology, targeting specific dimensions of the creative process:

1. **Design Action Move Variety:** Serves as a quantitative proxy for the breadth of exploration within the solution space. The variety of design action moves measures the diversity of cognitive and operational strategies employed during ideation (Shah et al., 2003). High variety suggests a comprehensive use of cognitive and operational strategies, while low variety indicates a constrained process dominated by a single action type (Wadinambiarachchi et al., 2024). This is quantified by calculating the ratio of each unique action label (e.g., Create, Modify, Branch) relative to the total number of actions performed.
2. **Majority Agreement:** An inverse measure of behavioral variety that quantifies consensus among designers at any given sequence in the workflow. High agreement signals convergence on a shared strategy, while low agreement indicates divergent activity. It is calculated as the ratio of the most frequent action to the total actions at each discrete sequence step, averaged across design stages.
3. **Branching Rate:** Characterizes the complexity and degree of divergence in the creative process (Chang et al., 2020; Liu et al., 2017). A high branching rate indicates the active generation of parallel conceptual pathways. This is operationalized as the percentage of all actions labeled as a “branch” move within each design stage.
4. **Idea Switch Ratio:** Evaluates the balance between deep, iterative elaboration on a single idea versus broad exploration across multiple concepts (Dow et al., 2010). It quantifies the design idea continuity by calculating the ratio of moves labeled as a “switch” to a new design idea versus those continuing the “same” design idea. A high switch ratio is a behavioral signature of parallel prototyping, which has been shown to improve design outcomes (Dow et al., 2011).

Collectively, these four metrics transform the sequential behavioral data captured by Workflow-Net into a robust quantitative framework. This enables the nuanced, action-centric cross-interface comparisons essential for understanding the efficacy of different design support systems.

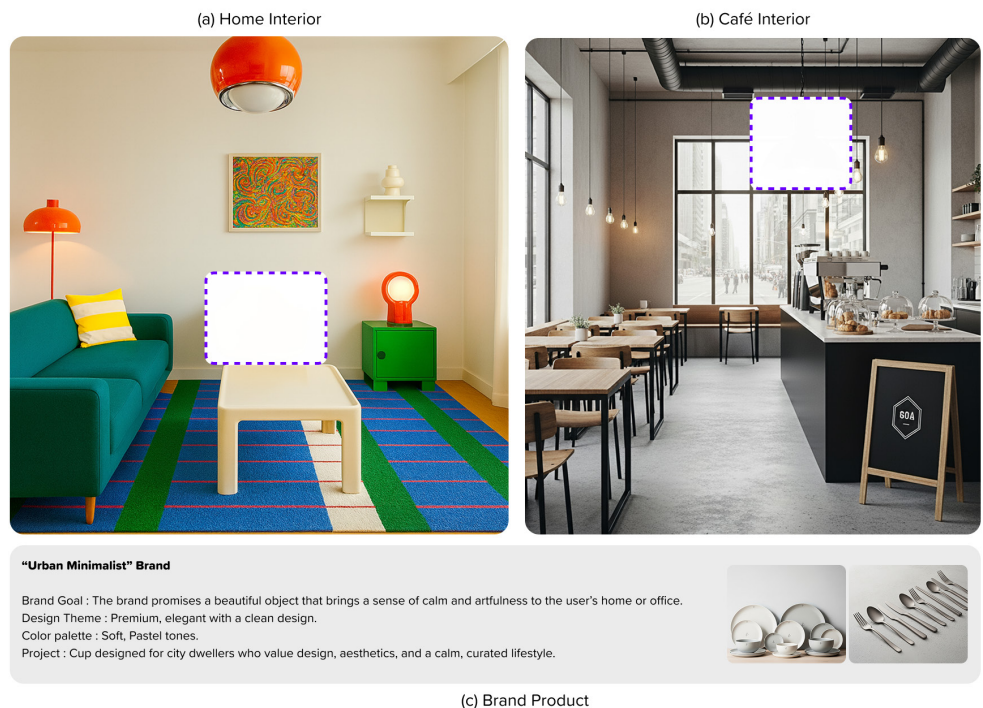
4. Experimental Design

We conducted a within-subject user study with nine expert designers (Table 3). The procedure began with an introduction, consent, and tutorial (20 min), followed by three distinct design tasks (Chair, Ceiling Lamp, Cup) as shown in Figure 7. Participants performed these tasks across all three interfaces, with the order counterbalanced to minimize carryover effects (Lazar et al., 2017). Each design task session lasted 45 minutes, during which we collected concurrent think-aloud protocols and screen recordings. Post-task evaluations included standardized questionnaires (Creativity Support Index (CSI) (Cherry & Latulipe, 2014), System Usability Scale (SUS) (Lewis, 2018), Task Load Index (TLX) (Hart & Staveland, 1988) and custom satisfaction (Lee et al., 2024) and agency (Chan et al., 2022; Shin et

al., 2025) scales, concluding with a 20-minute in-depth interview to assess the designers' experience.

Table 3 Participant Demographics

| Participant | Age | Gender | Experience (years) | Domain of Expertise |
|-------------|-----|--------|--------------------|---|
| P1 | 29 | Male | 5 | Interior Design, Interior Construction |
| P2 | 27 | Female | 4 | Architectural Planning, Furniture Design, Interior Design |
| P3 | 28 | Male | 3 | Lighting Design, Interior Design |
| P4 | 28 | Male | 2 | Visual Merchandising, Interior Design |
| P5 | 26 | Female | 5 | Interior Design |
| P6 | 29 | Male | 5 | Interior Design |
| P7 | 26 | Female | 3 | Interior Design |
| P8 | 29 | Female | 4 | Interior Design |
| P9 | 29 | Female | 5 | Interior Design |



(c) Brand Product

Figure 7 Two spatial contexts (a, b) and one brand product design (c) tasks used in the study's design brief

5. Results and Discussion

Our proposed action-centric approach demonstrated significant efficacy in quantifying high-level behavioral dynamics often obscured by prior methods. By successfully aggregating diverse individual workflows into a unified workflow, the methodology achieved high robustness across all datasets, validating Workflow-Net as a reliable analytical tool. Crucially, the resulting graph topology revealed that interface frameworks do not merely support design; rather, they fundamentally structure the cognitive arc. Each interface framework enforced distinct boundaries on exploration and refinement, significantly shaping the creative process.

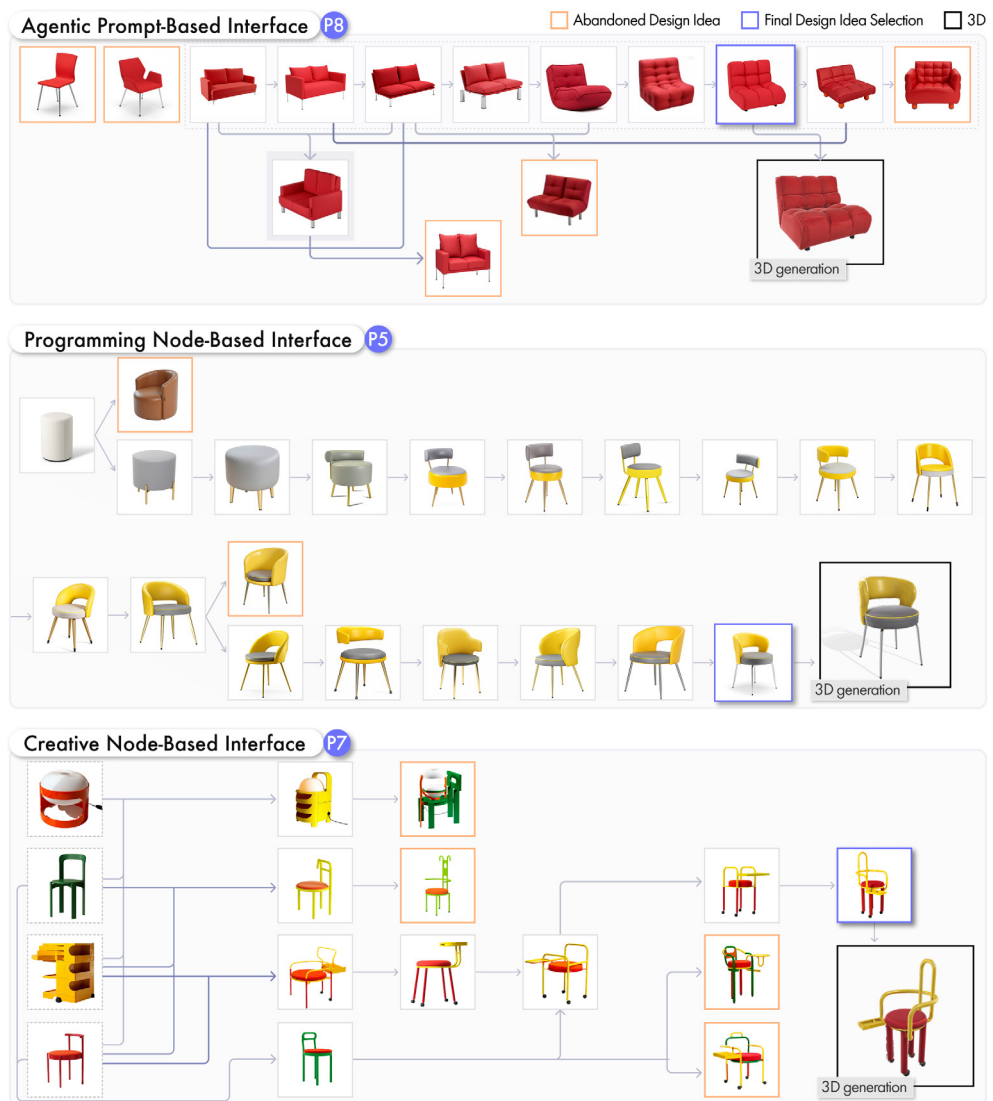


Figure 8 This figure illustrates three distinct behavioral patterns observed across the design workflows of three different designers performing the same chair design task, each utilizing a different generative AI-driven interface framework: P8 using AB (top), P5 using PB (middle), and P7 using CB (bottom)

Table 4 Workflow-Net Behavioral Analysis Metrics: Design Action Move Variety

| Design Move | AB | | PB | | CB | |
|-------------|------------|-------------|------------|-------------|------------|-------------|
| | N | Ratio | N | Ratio | N | Ratio |
| Create | 30 | 15% | 16 | 8% | 29 | 15% |
| Branch | 16 | 8% | 9 | 5% | 27 | 14% |
| Modify | 95 | 47% | 134 | 71% | 77 | 40% |
| Combine | 27 | 13% | 4 | 2% | 29 | 15% |
| Verify | 8 | 4% | 6 | 3% | 9 | 5% |
| Ask | 19 | 9% | 12 | 6% | 11 | 6% |
| Select | 9 | 4% | 9 | 5% | 9 | 5% |
| Sum | 204 | 100% | 190 | 100% | 191 | 100% |

Table 5 Workflow-Net Behavioral Analysis Metrics: Majority Agreement, Idea Switch Ratio, and Branch Rate

| Design Stage | Majority Agreement | Swith Ratio | Branch Rate |
|--------------|--------------------|---------------|---------------|
| AB | Ideation | 61% | 15% |
| | Development | 53% | 27% |
| | Finalization | 59% | 26% |
| | TOTAL AVG. | 57.67% | 22.67% |
| PB | Ideation | 75% | 13% |
| | Development | 84% | 6% |
| | Finalization | 81% | 0% |
| | TOTAL AVG. | 80.00% | 6.33% |
| CB | Ideation | 50% | 34% |
| | Development | 53% | 37% |
| | Finalization | 60% | 27% |
| | TOTAL AVG. | 54.33% | 32.67% |

Table 6 Subjective Evaluation Metrics: Creativity Support Index (CSI), System Usability Scale (SUS), and NASA-TLX

| Metric | AB (M, SD) | PB (M, SD) | CB (M, SD) |
|----------|---------------|---------------|---------------|
| CSI | 48.77 (24.40) | 65.68 (12.93) | 76.54 (21.09) |
| SUS | 67.50 (22.47) | 69.17 (14.95) | 86.94 (13.85) |
| NASA-TLX | 36.11 (21.87) | 37.35 (12.66) | 21.30 (14.89) |

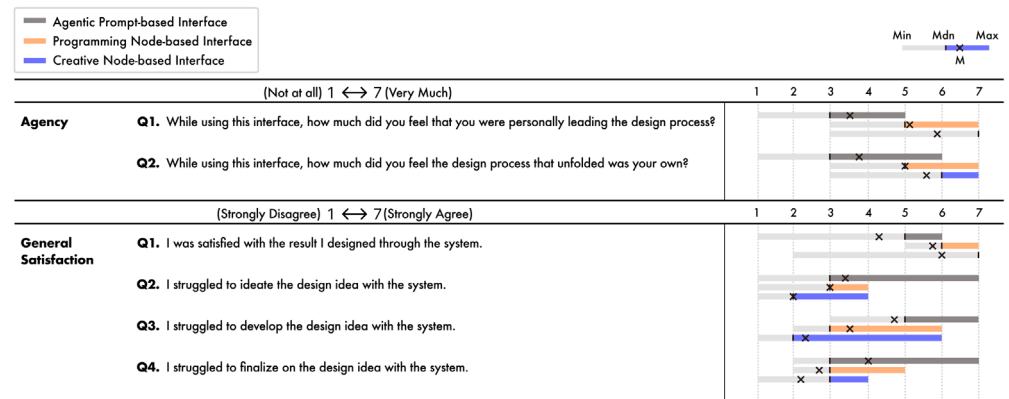


Figure 9 Subjective Evaluation Metrics: Agency and General Satisfaction

5. 1. AB Interface as a Design Support Tool for Open-ended Initial Design Stage Exploration

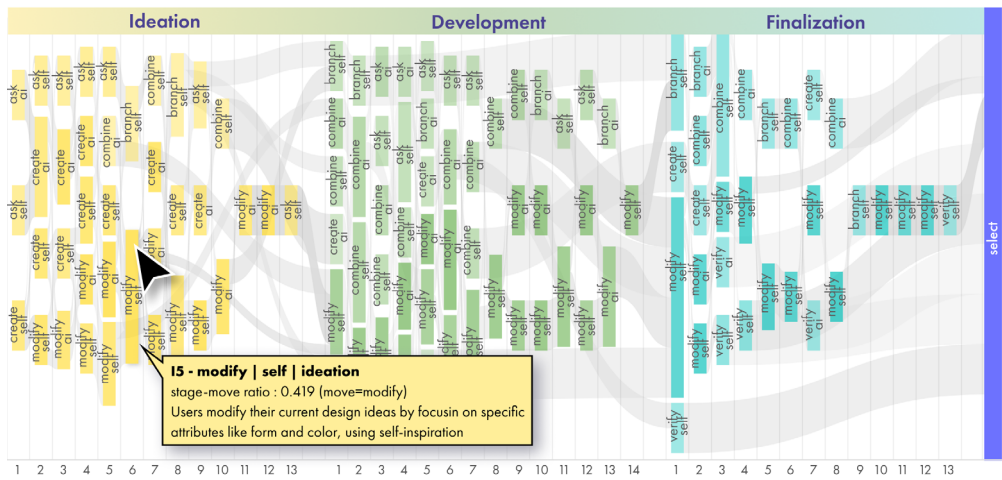


Figure 10 Aggregated workflow graph visualization of AB interface framework

The utility of the AB interface was largely confined to the open-ended, initial stages of the design process. Our Workflow-Net analysis revealed a process that lacked both strong divergence for exploration and effective convergence for refinement. While “*modify*” was the most frequent action, it constituted only 47% of all moves which was significantly lower than the focused activity observed in the PB interface (71%), indicating an inability to effectively iterate on design details (Table 4). Consequently, the AB interface received the lowest scores for creativity support (CSI: $M=48.77$, $SD=24.40$) and usability (SUS: $M=67.50$, $SD=22.47$), alongside a moderately high workload (NASA-TLX: $M=36.11$, $SD=21.87$) (Table 6).

This limitation became pronounced as designers transitioned into the design development stage, where the AB interface induced significantly more struggle compared to the other interface frameworks. General satisfaction ratings reflected this friction (Figure 9), with participants reporting the highest levels of struggle in developing (Q3: $M=4.78$, $SD=1.30$) and finalizing (Q4: $M=4.00$, $SD=1.66$) their concepts. Designers found text-only interaction to be a bottleneck for conveying complex visual concepts, resulting in consistently low ratings for user Agency (Q1: $M=3.56$, $SD=1.33$) (Figure 9). Qualitative feedback illuminated a profound sense of disconnect; as ideas became more concrete, the interface framework felt “*uncontrollable*” (P9), forcing designers to abandon their design ideas. Participants noted the frustration of a system that “*wouldn’t listen*” (P9), with P4 stating, “*I gave up on my ideas... it felt like the AI was the designer and I was the assistant*”.

These findings align with existing literature regarding the abstraction gap of text-only interaction (Dang et al., 2022; Liu et al., 2023; Masson et al., 2024), where users struggle to articulate nuanced visual intent through language alone. Ultimately, the AB interface framework proved effective only for simple, open-ended ideation, as user agency proved fragile once the design process demanded precision and refinement.

5. 2. PB Interface as a High-control Tool for Convergent Refinement Process

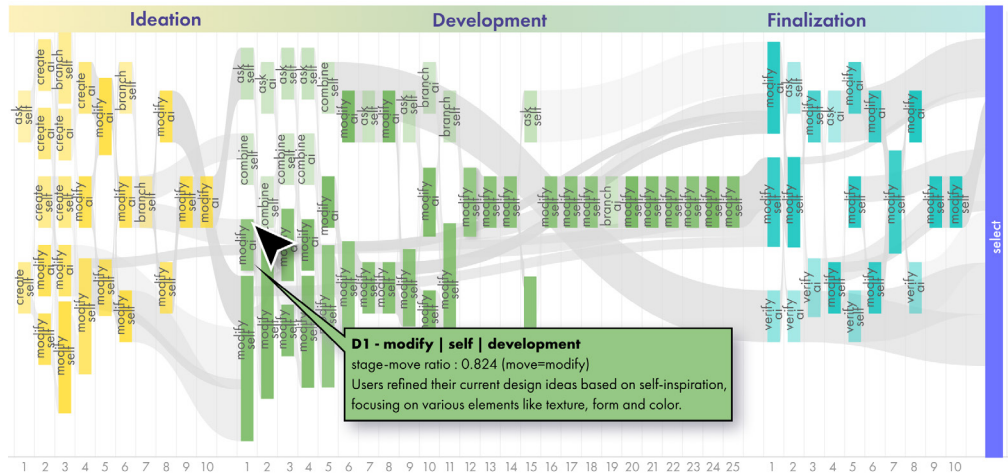


Figure 11 Aggregated workflow graph visualization of PB interface framework

The PB interface enforced a highly convergent and linear workflow centered on iterative refinement. Workflow-Net analysis revealed an overwhelming dominance of the “*modify*” action, which accounted for 71% of all moves (Table 4). This rigid focus led designers to elaborate deeply on a single concept rather than exploring alternatives, as evidenced by idea switch ratio decreasing from 13% during ideation to 0% by the finalization stage (Table 5). Moreover, divergent thinking was inhibited as the process progressed; the branching rate plummeted from 11% during ideation to 0% in the final stage. This linear trajectory resulted in highly similar behavior across participants, with the PB interface yielding the highest Majority Agreement scores, peaking at 84% during the design idea development stage.

While designers valued the granular control which was reflected in moderate CSI ($M=65.68$, $SD=12.93$) and SUS scores ($M=69.17$, $SD=14.95$) (Table 6), this precision came at a significant cost. Despite the high degree of procedural control, the interface yielded only a moderate sense of Agency (Q1: $M=5.11$, $SD=1.10$) (Figure 9), suggesting that manual technical management does not linearly translate to creative empowerment. Quantitative measures of struggle were lower than AB but remained present (Figure 9), particularly in the development phase (Q3: $M=3.56$, $SD=1.67$), though participants found it easier to finalize designs (Q4: $M=2.78$, $SD=0.97$) once the pipeline was established. As shown in Table 6, PB demonstrated the highest overall workload (NASA-TLX: $M=37.35$, $SD=12.66$), confirming the “*stressful and slow*” nature of manual pipeline construction. Specifically, results showed that the PB interface imposed a higher physical demand than the CB interface framework. Participant 6 captured this trade-off, noting that while “*connecting the nodes was really annoying,*” it remained “*effective for refinement*”.

These findings align with literature identifying manual pipeline construction as a notable usability barrier (Jiang, 2023). Furthermore, the observed paradox of control where rigid mechanics hinder creative exploration demonstrates how such interfaces force designers to

expend significant effort translating emergent ideas into structured workflows (Angert et al., 2023). This highlights that high procedural control can actually diminish creative agency when the interface's logic restricts the natural, fluid progression of the design.

5. 3. CB Interface as a Fluidic Design Partner Balancing Exploration and Refinement

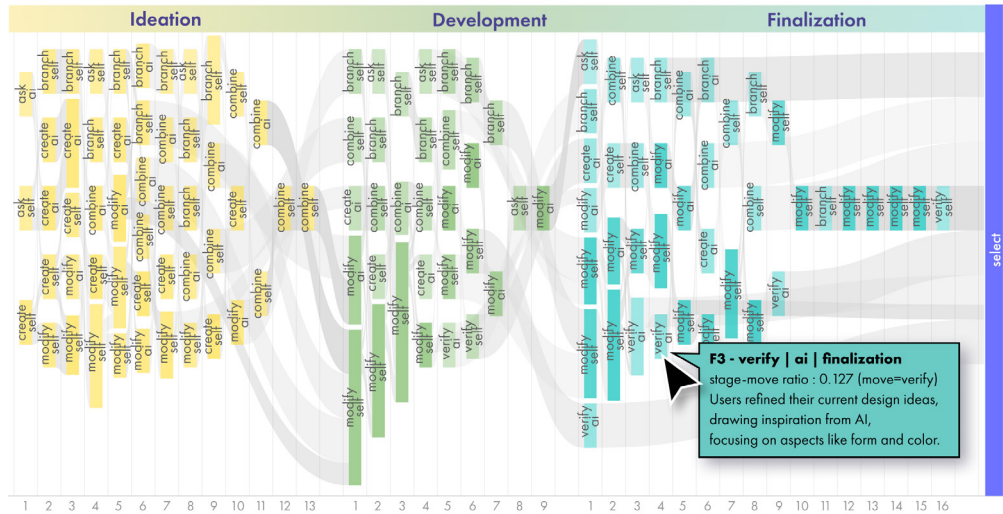


Figure 12 Aggregated workflow graph visualization of CB interface framework

In contrast to the other interface frameworks, the CB interface emerged as the most effective model, successfully balancing broad exploration with intuitive refinement. Our Workflow-Net analysis revealed a uniquely divergent and non-linear process, characterized by the highest variety of *design action moves*. This versatility was reflected in the lowest average Majority Agreement (53%), indicating a more personalized design process for each participant (Table 5). The interface actively encouraged divergent thinking, sustaining the highest Branching Rate (18% in ideation, 16% in development) and a design Idea Switch Ratio (32.67%) more than five times higher than the PB interface (Table 5).

This fluid, parallel workflow translated into superior subjective ratings across all primary metrics (Table 6). The CB interface achieved the highest creativity support (CSI: $M=76.54$, $SD=21.09$) and usability (SUS: $M=86.94$, $SD=13.85$) scores while maintaining the lowest workload (NASA-TLX: $M=21.30$, $SD=14.89$). General Satisfaction scores (Figure 9) across all questions were the most favorable in this condition, with the lowest reported struggle in creation (Q2: $M=2.00$, $SD=1.00$), development (Q3: $M=2.33$, $SD=1.58$), and finalization (Q4: $M=2.22$, $SD=1.20$). Qualitative feedback revealed that this success was rooted in the system's ability to act as a creative stimulant; P7 noted that the system allowed for more "*design experiments and tests*" than anticipated, generating results that "*exceeded expectations*".

However, the powerful generative capacity of CB interface presented a minor challenge; P9 found the sheer volume of information in the canvas space "*a bit chaotic*," highlighting a need for future mechanisms to prevent choice overload. Despite this, the CB interface, by

empowering designers with a strong sense of agency without imposing a heavy workload, stood out as the most holistic and successful design partner among the three interface frameworks.

Crucially, this exploratory power did not hinder refinement. By automatically visualizing workflow history, the CB interface framework provided essential process traceability, allowing designers to “*see their thought process*” at a glance (P3) without the manual labor required by the PB interface. This seamless integration created the highest sense of Agency (Q1: $M=5.89$, $SD=1.54$) (Figure 9), with participants feeling they were leading the process. P3 described “*the AI not as a director, but as a supportive assistant providing a guideline to efficiently produce 60–70% of the design for the designer to finalize*”.

6. Design Implications for Generative AI-Driven Design Support Systems

The distinct performance profiles of the AB, PB, and CB interface frameworks offer clear directives for designing future AI-driven creative design partners. Our findings suggest that future AI design tools should move toward hybrid, agentic interfaces that bridge the gap between abstract intent and procedural precision through the following three principles:

6. 1. Balancing Fluidity and Precision through Hybrid Architectures

While the AB interface framework is effective for initial ideation, its abstraction gap makes it a bottleneck for refinement. Conversely, the PB interface framework’s procedural control is valuable for detail, but its rigidity inhibits the fluid exploration essential to creativity. Future systems should synthesize these strengths by utilizing an infinite canvas as the primary workspace for non-linear exploration, while integrating procedural nodes as an optional, on-demand layer. Tools should allow designers to transition seamlessly from low-fidelity conversational ideation (AB) to high-fidelity procedural tuning (PB) within a single canvas-based environment. This allows the system to support the entire design lifecycle—from broad divergence to structured convergence—without forcing the user to switch tools or sacrifice creative flow.

6. 2. Mitigating Cognitive Load via Agent-Assisted Automation

A primary barrier observed in the PB interface was the usability wall created by manual pipeline construction. To lower this barrier, future systems must move away from manual wiring toward intent-based automation. Instead of requiring users to connect every node, generative AI should act as a technical intermediary that intelligently suggests design actions or auto-generates procedural pipelines based on high-level designer intent. Implementing an intelligent agent that can suggest next design moves or an automated workflow pipeline construction features enables reproducibility and granular control without the physical and cognitive overhead of manual node management. By automating the labor of the design process, the interface allows the designer to remain focused in the strategy phase of the workflow.

6. 3. Fostering Agency through Process Traceability and Prompt Archiving

Our results indicate that user agency is fragile and closely tied to a designer's ability to visually trace their thought process. The success of the CB interface framework was largely due to its inherent process traceability. Future tools should treat the workflow history as a primary design asset. Systems should implement 'Prompt Archives' that visualize the linguistic and procedural evolution of a design. By preserving the exchange of interactions (e.g., conversation prompts) between the designer and AI, the interface supports reflection and iterative reuse.

Real-time node traceability ensures that users can backtrack, branch, and experiment without manual overhead, reinforcing the role of the AI as a supportive assistant rather than a "black-box" director. This shift ensures the system remains designer-led, upholding creative agency even as the AI's generative power increases.

7. Conclusion

This study establishes that the interface framework is not merely a medium for Generative AI, but a structural determinant of the designers' creative workflow. To evaluate this, we introduced Workflow-Net, a novel method combining protocol analysis with LLM-assisted clustering to map high-level design action behaviors across multiple users. Through this lens, we revealed fundamental differences in how interaction modalities shape the creative design process. While AB interfaces support broad ideation and PB interfaces enable precise control, both impose significant tradeoffs either through abstraction gaps or high cognitive load. In contrast, the CB interface emerged as a balanced model; by automating process traceability and visualizing the workflow history, it successfully bridged divergent exploration with convergent refinement, fostering the highest level of designer agency.

Beyond evaluation, the Workflow-Net methodology offers a new foundation for agent enhancement. The quantitative metrics generated by Workflow-Net such as branching rates and action frequencies can serve as a real-time signal for context-aware AI agents. By monitoring these behavioral signatures, an intelligent system could infer the designer's current phase within the workflow. By monitoring the data, an intelligent agent could infer the designers' current state in the design workflow. This situational awareness would allow the agent to dynamically adapt its behavior: providing creative stimulants and diverse alternatives during exploratory phases, or switching to granular, logic-based assistance and part-level editing (such as inpainting) during refinement.

Furthermore, it was not within the scope of this study to statistically analyze the impact of individual designers' capabilities on the generative outcome. Rather, the behavioral patterns and average results observed in this study reflect the specific demographic context of our participants (refer to Table 3). Future research should gather additional data to investigate how varying levels of design expertise (e.g., novices compared to professionals) might alter the way these distinct interface frameworks shape the creative design process workflows.

This work provides a robust framework for developing future generative AI-driven designer-led systems. By shifting the role of AI from a “black-box” director to a responsive, co-creative partner that adapts to the fluid dynamics of the creative process, we move closer to tools that do not just generate content but effectively empower the human designer.

References

1. Angert, T., Suzara, M., Han, J., Pondoc, C., & Subramonyam, H. (2023, October). Spellburst: A node-based interface for exploratory creative coding with natural language prompts. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (pp. 1–22). <https://doi.org/10.1145/3586183.3606719>.
2. Caetano, A., Verma, K., Taheri, A., Kumaran, R., Chen, Z., Chen, J., ... & Sra, M. (2025). Agentic workflows for conversational human-ai interaction design. *arXiv preprint arXiv:2501.18002*. <https://doi.org/10.48550/arXiv.2501.18002>.
3. Chan, L., Liao, Y. C., Mo, G. B., Dudley, J. J., Cheng, C. L., Kristensson, P. O., & Oulasvirta, A. (2022, April). Investigating positive and negative qualities of human-in-the-loop optimization for designing interaction techniques. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (pp. 1–14). <https://doi.org/10.1145/3491102.3501850>.
4. Chang, M., Lafreniere, B., Kim, J., Fitzmaurice, G., & Grossman, T. (2020). Workflow graphs: A computational model of collective task strategies for 3D design software. In *Graphics Interface 2020*.
5. Chen, P., Yao, J., Cheng, Z., Cai, Y., Li, J., You, W., & Sun, L. (2025, April). Coexplored: framing and advancing collaborative design space exploration between human and AI. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (pp. 1–20). <https://doi.org/10.1145/3706598.3713869>.
6. Cherry, E., & Latulipe, C. (2014). Quantifying the creativity support of digital tools through the creativity support index. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(4), 1–25. <https://doi.org/10.1145/2617588>.
7. Choi, J., Lee, S. W., & Hyun, K. H. (2025, April). GenPara: Enhancing the 3D Design Editing Process by Inferring Users' Regions of Interest with Text-Conditional Shape Parameters. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (pp. 1–21). <https://doi.org/10.1145/3706598.3713502>.
8. Chung, J. J. Y., & Adar, E. (2023, October). Promptpaint: Steering text-to-image generation through paint medium-like interactions. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (pp. 1–17). <https://doi.org/10.1145/3586183.3606777>.
9. Comfy Org. (2024). *ComfyUI*.
10. Cycling '74. (n.d.). *Cycling '74 documentation*.
11. Dang, H., Mecke, L., Lehmann, F., Goller, S., & Buschek, D. (2022). How to prompt? Opportunities and challenges of zero-and few-shot learning for human-AI interaction in creative applications of generative models. *arXiv preprint arXiv:2209.01390*. <https://doi.org/10.48550/arXiv.2209.01390>.
12. Desmond, M., & Brachman, M. (2024). Exploring prompt engineering practices in the enterprise. *arXiv preprint arXiv:2403.08950*. <https://doi.org/10.48550/arXiv.2403.08950>.
13. Dow, S. P., Glassco, A., Kass, J., Schwarz, M., Schwartz, D. L., & Klemmer, S. R. (2010). Parallel prototyping leads to better design results, more divergence, and increased self-efficacy. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 17(4), 1–24. <https://doi.org/10.1145/1879831.1879836>.
14. Dow, S., Fortuna, J., Schwartz, D., Altringer, B., Schwartz, D., & Klemmer, S. (2011, May). Prototyping dynamics: sharing multiple designs improves exploration, group rapport, and results. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 2807–2816). <https://doi.org/10.1145/1978942.1979359>.

15. Drosos, I., Williams, J., Sarkar, A., Wilson, N., Rintel, S., & Panda, P. (2025, June). Dynamic prompt middleware: Contextual prompt refinement controls for comprehension tasks. In *Proceedings of the 4th Annual Symposium on Human-Computer Interaction for Work* (pp. 1–23). <https://doi.org/10.1145/3729176.3729203>.
16. Duan, R., Zhu, C., Chen, Y., Hu, Y., Shi, J., & Ramani, K. (2025, July). DesignFromX: Empowering consumer-driven design space exploration through feature composition of referenced products. In *Proceedings of the 2025 ACM Designing Interactive Systems Conference* (pp. 1040–1060). <https://doi.org/10.1145/3715336.3735824>.
17. Epic Games. (n.d.). *Blueprints visual scripting in Unreal Engine*.
18. Ericsson, K. A. (2017). Protocol analysis. *A Companion to Cognitive Science*, 425–432. <https://doi.org/10.1002/9781405164535.ch33>.
19. Ericsson, K. A., & Simon, H. A. (1980). Verbal reports as data. *Psychological Review*, 87(3), 215. <https://doi.org/10.1037/0033-295X.87.3.215>.
20. Fani Sani, M., Sroka, M., & Burattin, A. (2023, October). LLMs and process mining: Challenges in rpa: Task grouping, labelling and connector recommendation. In *International Conference on Process Mining* (pp. 379–391). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-56107-8_29.
21. FLORA. (2025). Flora.ai.
22. Fu, J., Tang, H., Khan, A., Mehrotra, S., Ke, X., & Gao, Y. (2025). In-context clustering-based entity resolution with large language models: A design space exploration. *Proceedings of the ACM on Management of Data*, 3(4), 1–28. <https://doi.org/10.1145/3749170>.
23. Goldschmidt, G. (2014). *Linkography: unfolding the design process*. MIT Press. <https://doi.org/10.7551/mitpress/9455.001.0001>.
24. Google. (n.d.). *Gemini API: Models*.
25. Hagberg, A., Swart, P. J., & Schult, D. A. (2007). *Exploring network structure, dynamics, and function using NetworkX* (No. LA-UR-08-05495; LA-UR-08-5495). Los Alamos National Laboratory (LANL).
26. Han, K., Wen, H., Shi, J., Lu, K. H., Zhang, Y., Fu, D., & Liu, Z. (2019). Variational autoencoder: An unsupervised model for encoding and decoding fMRI activity in visual cortex. *NeuroImage*, 198, 125–136. <https://doi.org/10.1016/j.neuroimage.2019.05.039>.
27. Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In *Advances in psychology* (Vol. 52, pp. 139–183). North-holland. [https://doi.org/10.1016/S0166-4115\(08\)62386-9](https://doi.org/10.1016/S0166-4115(08)62386-9).
28. Hatcher, G., Ion, W., Maclachlan, R., Marlow, M., Simpson, B., Wilson, N., & Wodehouse, A. (2018). Using linkography to compare creative methods for group ideation. *Design Studies*, 58, 127–152. <https://doi.org/10.1016/j.destud.2018.05.002>.
29. Hyun, K. H., & Jang, Y. (2025). BIGcad: Assisting 3D CAD Modeling with Workflow Graph-Driven Bayesian Command Inferences. *Archives of Design Research*, 38(2), 179–199. <https://doi.org/10.15187/adr.2025.05.38.2.179>.
30. Jang, Y., & Hyun, K. H. (2024, May). Advancing 3D CAD with workflow graph-driven bayesian command inferences. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems* (pp. 1–6). <https://doi.org/10.1145/3613905.3650895>.
31. Jiang, P. (2023, April). Positional Control in Node-Based Programming. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems* (pp. 1–7). <https://doi.org/10.1145/3544549.3585878>.
32. Kwon, E., Rao, V., & Goucher-Lambert, K. (2023). Understanding inspiration: Insights into how designers discover inspirational stimuli using an AI-enabled platform. *Design Studies*, 88, 101202. <https://doi.org/10.1016/j.destud.2023.101202>.
33. Lawton, T., Grace, K., & Ibarrola, F. J. (2023, July). When is a tool a tool? user perceptions of system agency in human-ai co-creative drawing. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference* (pp. 1978–1996). <https://doi.org/10.1145/3563657.3595977>.

34. Lazar, J., Feng, J. H., & Hochheiser, H. (2017). *Research methods in human-computer interaction*. Morgan Kaufmann.
35. Lee, S. W., Jo, T. H., Jin, S., Choi, J., Yun, K., Bromberg, S., ... & Hyun, K. H. (2024, May). The impact of sketch-guided vs. prompt-guided 3D generative AIs on the design exploration process. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (pp. 1–18). <https://doi.org/10.1145/3613904.3642218>.
36. Lewis, J. R. (2018). The system usability scale: past, present, and future. *International Journal of Human-Computer Interaction*, 34(7), 577–590. <https://doi.org/10.1080/10447318.2018.1455307>.
37. Lin, D. C. E., Kang, H. B., Martelaro, N., Kittur, A., Chen, Y. Y., & Hong, M. K. (2025, April). Inkspire: Supporting design exploration with generative ai through analogical sketching. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (pp. 1–18). <https://doi.org/10.1145/3706598.3713397>.
38. Lin, X., Huang, H., Huang, K., Shu, X., & Vines, J. (2025, April). Seeking inspiration through human-llm interaction. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (pp. 1–17). <https://doi.org/10.1145/3706598.3713259>.
39. Liu, M. X., Sarkar, A., Negreanu, C., Zorn, B., Williams, J., Toronto, N., & Gordon, A. D. (2023, April). "What it wants me to say": Bridging the abstraction gap between end-user programmers and code-generating large language models. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (pp. 1–31). <https://doi.org/10.1145/3544548.3580817>.
40. Liu, Z., Kerr, B., Dontcheva, M., Grover, J., Hoffman, M., & Wilson, A. (2017, June). Coreflow: Extracting and visualizing branching patterns from event sequences. In *Computer Graphics Forum* (Vol. 36, No. 3, pp. 527–538). <https://doi.org/10.1111/cgf.13208>.
41. Luera, R., Rossi, R. A., Siu, A., Dernoncourt, F., Yu, T., Kim, S., ... & Lipka, N. (2024). Survey of User Interface Design and Interaction Techniques in Generative AI Applications. *arXiv preprint arXiv:2410.22370*, 1–42. <https://doi.org/10.48550/arXiv.2410.22370>.
42. Masson, D., Malacria, S., Casiez, G., & Vogel, D. (2024, May). Directgpt: A direct manipulation interface to interact with large language models. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (pp. 1–16). <https://doi.org/10.1145/3613904.3642462>.
43. Meshy. (n.d.). Meshy documentation.
44. Miller, J. K., & Alexander, T. J. (2025). Human-interpretable clustering of short text using large language models. *Royal Society Open Science*, 12(1). <https://doi.org/10.1098/rsos.241692>.
45. Mondal, S., Bappon, S. D., & Roy, C. K. (2024, April). Enhancing user interaction in ChatGPT: Characterizing and consolidating multiple prompts for issue resolution. In *Proceedings of the 21st International Conference on Mining Software Repositories* (pp. 222–226). <https://doi.org/10.1145/3643991.3645085>.
46. OpenAI. (2025). *Images (Vision) guide*.
47. OpenAI. (n.d.). *GPT-4o*.
48. Photoroom. (n.d.). *Photoroom documentation*.
49. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., & Chen, M. (2022). Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2), 3. <https://doi.org/10.48550/arXiv.2204.06125>.
50. Recon Labs Inc. (2025). *GENPRESSO*.
51. Resonate International INC. (2025). *Lovart – The creative AI platform*.
52. Robert McNeel & Associates. (n.d.). *Grasshopper*.
53. Shah, J. J., Smith, S. M., & Vargas-Hernandez, N. (2003). Metrics for measuring ideation effectiveness. *Design studies*, 24(2), 111–134. [https://doi.org/10.1016/S0142-694X\(02\)00034-0](https://doi.org/10.1016/S0142-694X(02)00034-0).
54. Shen, H., Shen, L., Wu, W., & Zhang, K. (2025, April). Ideationweb: Tracking the evolution of design ideas in human-ai co-creation. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (pp. 1–19). <https://doi.org/10.1145/3706598.3713375>.
55. Shin, J., Polyanskaya, A., Lucero, A., & Oulasvirta, A. (2025, April). No Evidence for LLMs Being Useful in Problem Reframing. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (pp. 1–25). <https://doi.org/10.1145/3706598.3713273>.

56. Smith, A., Anderson, B. R., Otto, J. T., Karth, I., Sun, Y., Joon Young Chung, J., ... & Kreminski, M. (2025, June). Fuzzy Linkography: Automatic Graphical Summarization of Creative Activity Traces. In *Proceedings of the 2025 Conference on Creativity and Cognition* (pp. 637–650). <https://doi.org/10.1145/3698061.3726915>.
57. Tarekegn, A. N. (2024). Large language model enhanced clustering for news event detection. *arXiv preprint arXiv:2406.10552*. <https://doi.org/10.48550/arXiv.2406.10552>.
58. Vaarandi, R., & Bahşi, H. (2025). Using large language models for template detection from security event logs. *International Journal of Information Security*, 24(3), 104. <https://doi.org/10.1007/s10207-025-01018-y>.
59. von Luxburg, U. (2010). Clustering stability: an overview. *Found Trends Mach Learn* 2(3), 235–274.
60. Wadinambarachchi, S., Kelly, R. M., Pareek, S., Zhou, Q., & Velloso, E. (2024, May). The effects of generative AI on design fixation and divergent thinking. In *Proceedings of the 2024 CHI conference on human factors in computing systems* (pp. 1–18). <https://doi.org/10.1145/3613904.3642919>.
61. Xue, X., Lu, Z., Huang, D., Wang, Z., Ouyang, W., & Bai, L. (2025). Comfybench: Benchmarking llm-based agents in comfyui for autonomously designing collaborative ai systems. In *Proceedings of the computer vision and pattern recognition conference* (pp. 24614–24624). <https://doi.org/10.1109/cvpr52734.2025.02292>.
62. You, W., Lu, Y., Ma, Z., Li, N., Zhou, M., Zhao, X., ... & Sun, L. (2025). DesignManager: An Agent-Powered Copilot for Designers to Integrate AI Design Tools into Creative Workflows. *ACM Transactions on Graphics (TOG)*, 44(4), 1–26. <https://doi.org/10.1145/3730919>.
63. Zhang, L., Rao, A., & Agrawala, M. (2023). Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 3836–3847). <https://doi.org/10.1109/iccv51070.2023.00355>.
64. Zhong, R., Shin, D., Meza, R., Klasnja, P., Colusso, L., & Hsieh, G. (2024, May). AI-assisted causal pathway diagram for human-centered design. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (pp. 1–19). <https://doi.org/10.1145/3613904.3642179>.
65. Zhou, J., Li, R., Tang, J., Tang, T., Li, H., Cui, W., & Wu, Y. (2024, May). Understanding nonlinear collaboration between human and AI agents: A co-design framework for creative design. In *Proceedings of the 2024 CHI conference on human factors in computing systems* (pp. 1–16). <https://doi.org/10.1145/3613904.3642812>.
66. Zhou, Z., Jin, J., Phadnis, V., Yuan, X., Jiang, J., Qian, X., ... & Du, R. (2024, May). Experiencing InstructPipe: Building multi-modal AI pipelines via prompting LLMs and visual programming. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems* (pp. 1–5). <https://doi.org/10.1145/3613905.3648656>.