Flint: Standing in the loop of thought

Exploring Human Agency in the Age of AI

Kasper Zhang

A thesis exhibition presented to OCAD University in partial fulfillment of the requirements
for the degree of Master of Design in Digital Futures
Toronto, Ontario, Canada, 2026

ABSTRACT

Generative AI currently employs linear chat interfaces that may bypass users' critical thinking. This design can result in automation bias, as individuals may accept AI-generated outputs without sufficient scrutiny. Consequently, users often become passive supervisors rather than active creators. This study addresses the challenge of human-AI co-thinking by shifting the emphasis from AI-generated results to collaborative cognitive processes. The project seeks to re-establish users as primary creators by designing a digital environment that reduces passive supervision. To evaluate human-AI co-thinking, a prototype named Flint was developed. Flint replaces the traditional chat stream with a dynamic spatial canvas, enabling users to externalize and manipulate their thoughts as discrete objects. As a mentor, Flint analyzes user needs and generates customized thinking tools presented as blank objects. Although the AI configures the environment according to user requirements, it cannot write on the canvas, requiring users to engage in genuine cognitive effort. This approach restores productive struggle and positions the user as the architect of the thought process. The findings indicate that transitioning from linear to spatial interfaces can restore human agency, positioning AI as a co-thinker rather than a replacement for human creativity.

Table of Contents

LIST OF FIGURES

Chapter 1 – INTRODUCTION

1.1 Personal Motivation

My interest in this research stemmed from both excitement and discomfort. In late 2024, at the start of my graduate studies, Large Language Models (LLMs) began reshaping my workflow. I initially used these tools for debugging and creating documentation. Their use soon extended beyond engineering: I employed them to outline writing and quickly summarize dense theoretical readings. The speed and convenience felt empowering. For a time, I enjoyed delegating part of my cognitive work to these systems.

As the tools became routine, my enthusiasm changed. I felt a subtle distance between me and my work—chat-based interfaces nudged me from creator to supervisor. I submitted prompts and waited for refined answers. The messy, uncertain stages of thinking began to vanish from my process. I realized I was no longer thinking with the machine. I was waiting for it.

This new posture made me question whether efficiency eroded my cognitive agency. The concern grew when I used these systems for deep conceptual framing. For example, I used them to map the initial logic for this research. The linear chat interface forced my non-linear thoughts into a rigid sequence, making it hard to trace or adjust my reasoning as it formed.

This challenge became my design problem. I wanted a system that kept me engaged in reasoning instead of pulling me out of it. I sought a tool that valued cognitive effort. It also needed to support collaboration, where ideas could be placed, revised, and negotiated in a shared space.

Flint emerged from this search as a response to current limitations in AI models. Flint is a custom software prototype developed for this thesis. It works as a Generative UI environment with a spatial canvas, replacing the typical linear chat. Flint does not just produce standalone text. Instead, it serves as a methodologist, generating interactive structures such as decision matrices and mind maps. These prompts users to externalize and organize their reasoning. Active user input is required. This shifts interaction from passive reception to active engagement. Flint creates a shared workspace where human reasoning and machine support develop in parallel. The project is my effort to return to my own cognitive process. It also explores how interaction design can protect, not diminish, human agency in the age of generative AI.

1.2 Naming the App: Flint

The name *Flint* was chosen to express the core philosophy of this research: ideas require human agency. Historically, a piece of flint is a tool of potential. On its own, it does nothing. A spark appears only when a human hand strikes it against steel. This image stands in contrast to the trend of frictionless AI design. Such a design removes effort. It treats thinking as something to automate. These tools behave like automatic lighters. Press a button, and a flame appears.

This research views the resistance felt when building an argument not as a flaw, but as a productive necessity. Understanding forms in this space of effort. Flint is therefore designed as a surface that users can strike to generate their own spark. It is not a machine that thinks for them. The name reinforces that the system remains passive. It waits until activated by human intent. This design supports the goal of keeping the user in the loop during the creative act.

1.3 Significance of the Study

This research advances the fields of Interaction Design and Human-Computer Interaction (HCI) by redirecting attention from static conversational agents to Generative User Interfaces (GenUI). The study challenges the prevailing one-size-fits-all chatbot paradigm and demonstrates the necessity for adaptive environments that facilitate, rather than automate, user reasoning.

1.3.1 Theoretical Significance: From Cognitive Offloading to Externalization

At a theoretical level, this study situates Generative AI within the framework of distributed cognition. This view holds that cognition does not reside solely within the mind. Instead, it is spread across social interactions, artifacts, and environmental structures (James Hollan, 2000). Framing Generative AI this way lets the study challenge the frictionless automation. Many current AI interfaces prioritize speed and ease by performing tasks for the user. This approach leads to cognitive offloading.

1.3.2 Synthesizing Human Agency in Human–Computer Interaction

This shift puts human control at the center of interface design, so it is important to be clear about what this means. In psychology, agency is the ability to control what you do and what happens in your life (Bandura, 2001). In Human–Computer Interaction (HCI), agency refers to a user's sense of ownership and control over how computers operate (Shneiderman, 2022). Building from this, the role of agency in interface design has evolved alongside technological advancements.

To understand how this shift impacts practice, consider the historical context: interface design has supported agency through direct manipulation, enabling users to interact with visible objects such as dragging files or adjusting parameters in real time. However, generative AI complicates this model by introducing intent-based interactions, where systems interpret high-level requests and act on behalf of the user. As a result, this approach risks distancing users from the underlying reasoning processes.

To address these challenges, preserving agency in AI-mediated interactions requires integrating direct manipulation with intent-based automation. Shneiderman's emphasis on user control should be aligned with the generative and automated capabilities of large language models. In this context, automation ought to extend user intent rather than replace it.

1.3.3 Methodological Significance: The Limitations of Linear Interaction

This thesis critiques the use of Conversational User Interface (CUI) for complex problem-solving from a methodological view. Using Research through Design (RtD), it identifies friction points in chat-based AI. Linear text compresses non-linear thoughts, causing issues. Large Language Models (LLMs) generate strong content. However, chat streams force multidimensional ideas into a single sequence and hide logical connections.

1.3.4 Practical Significance: The Emergence of Generative UI

This thesis examines the functional mechanics of Generative User Interfaces as a means of supporting design thinking. The research aligns with a broader trend identified by Google Research (2024). Generative systems are evolving from producing static text to enabling interactive experiences. The system functions as a methodological tool by mapping linguistic patterns—such as users weighing competing priorities or tracing sequences of events—to specific cognitive structures. For instance, Flint can automatically instantiate a Decision Matrix for complex choices or a 5 Whys framework for in-depth causal analysis. Unlike conventional digital whiteboards that require manual template selection and configuration, Flint reduces analysis paralysis. It does this by dynamically providing scaffolding tailored to the user's immediate reasoning requirements.

This research also addresses the challenge of intent specification within generative systems. Empirical studies on iterative prompting indicate a key barrier in Human–AI collaboration. Users often struggle to articulate complex mental models in a single prompt (Chen et al., 2025). Flint mitigates this by providing a mutable spatial canvas that supports incremental interaction. In contrast to linear chat interfaces, which often depend on a successful single-shot prompt, the spatial model allows users to express, refine, and revise intent progressively. Users can modify individual nodes, reorganize relationships, and adjust structures incrementally. This reduces the cognitive demands associated with prompt engineering.

1.4 Problem Context

The integration of Generative AI into creative and analytical work has created a tension between automation and human agency (Shneiderman, 2022). These systems offer great speed, yet they often operate as ‘black boxes’, where users cannot see how decisions are made (Van Berkel, 2024). This limitation has prompted the emergence of Human-Centered Explainable AI (HCXAI). Unlike traditional Explainable AI (XAI), HCXAI prioritizes technical transparency; HCXAI approaches explainability as a sociotechnical challenge. As articulated by Ehsan and Riedl (2020), the aim of HCXAI is not simply to enable model inspection but to deliver explanations that are meaningful, contextual, and actionable for a specific user.

Within this framework, transparency alone is insufficient. Scholars argue that a system can be considered human-centered only if it bridges the ‘gulf of evaluation’ by enabling users to understand its reasoning and intervene when necessary (Shneiderman, 2022). This requires interfaces that allow users to shape and redirect the model's logic, shifting the interaction away from passive consumption and toward active co-creation.

This passivity becomes stronger because most systems rely on the 'conversational user interface'(CUI) or chatbot. Natural language is easy to use, but a linear chat stream does not work well for complex or non-linear thinking. Recent empirical research by Shukla et al. (2025) points to 'cognitive offloading,' a pattern in which users rely on AI for high-level reasoning and gradually lose critical judgment. When users hand over the structure of their thoughts to AI, they lose the 'productive struggle' needed for deep understanding. This pattern aligns with 'automation bias,' in which people trust automated responses even when the system makes mistakes or hallucinates (Center for Security and Emerging Technology, 2024). The limitations of the CUI extend beyond its inherent linearity; it fundamentally operates as a 'static container' that struggles to accommodate complex, non-linear cognitive activities. This interaction model often results in a significant 'intent specification gap,' where users find it difficult to accurately articulate and refine complex mental models through a single, temporal text stream. The emergence of Generative User Interfaces (GenUI) offers a novel path forward. By using dynamically generated interactive components rather than fixed text flows, GenUI allows the interface to restructure itself in response to the user's real-time cognitive needs. This paradigm shift aims to eliminate the cognitive friction of adapting to a tool, transforming the interface from a passive recipient of information into an active, adaptive environment for thought.

In response to the opacity and risk of passivity associated with ‘black box’ systems, designers must reconsider how the interaction loop is structured. Simply integrating AI into existing workflows is insufficient. The interaction model itself must evolve to ensure users remain actively engaged in the process rather than being reduced to supervisors of automated output. Ben Shneiderman argues that interfaces should aim for ‘High Levels of Human Control’ together with ‘High Levels of Computer Automation’ (Shneiderman, 2022). The main challenge is to create systems that support user agency—the ability to take meaningful action and form plans (Samarth Swarup, 2025). To maintain this agency, such a shift is necessary because chat-based interaction compresses complex, non-linear reasoning into a single temporal sequence, obscuring the underlying structure of an argument. By contrast, a spatial canvas externalizes the reasoning process by representing ideas as discrete, manipulable elements. This representation makes relationships between ideas visible and allows users to intervene, reorganize, and critically examine the structure of their thinking.

1.5 Research Question

Having established that current Conversational User Interfaces (CUIs) often promote cognitive offloading and obscure in the reasoning process, a critical gap in interaction design becomes evident. The challenge extends beyond the generation of content to the design of the environment in which that content is produced, particularly in relation to preserving user control and agency. In response to this gap, this thesis formulates a set of primary and secondary research questions that examine how Generative UI can mediate between high levels of automation and sustained human agency.

Primary Research Question

1. How can interaction design transform Generative AI from a passive automation tool into a 'co-thinking' partner that preserves human cognitive agency?

Secondary Research Questions

1. **The Problem of Interface:** How do linear Conversational User Interfaces (CUIs) limit the user's ability to engage in non-linear reasoning and structural problem-solving?

2. **The Solution of Generative UI:** How can adaptive, spatial frameworks (Generative UI) externalize the reasoning process to allow users to actively verify, modify, and guide the AI's logic?

1.6 Objectives

The main objective of this thesis is to design and evaluate Flint, a system that shifts the user–AI relationship from passive automation to active collaboration. This shift is realized through a transition from a linear chat interface to a generative spatial canvas. Within this model, the AI assumes the role of a structural architect by generating empty cognitive frameworks, such as mind maps or matrices. The user, by contrast, remains the primary content creator, actively populating these structures with reasoning and domain-specific logic. To address the research questions, this study focuses on the following specific objectives:

To critique the limitations of the chat interface.

This research aims to explain why standard chat interfaces have difficult supporting complex problem-solving. By creating and testing a chat-based prototype (Prototype V1), the study shows how the linear format pushes users to simplify their thinking and makes it hard to see the full structure of an argument.

To develop a Generative UI system for better thinking.

A key goal is to build an AI system that can generate the right interface for the task (Prototype V3). Instead of using a single chat box for every situation, the system identifies the user's goal (for example, brainstorming or diagnosing a root cause) and produces a visual tool that fits the problem, such as a mind map or Ishikawa diagram.

To evaluate human control through iterative design.

Through cycles of building, testing, and reflecting, this study assesses whether adaptive interfaces help users feel more in control of their work. This includes comparing the experience of overseeing a chatbot with the experience of shaping ideas directly on a dynamic canvas.

1.7 Thesis Structure Overview

Chapter 2 – Literature and Contextual Review

This chapter provides a critical synthesis of the research landscape, situated within the rapid and volatile advancements of early 2026. It examines the current state of Generative AI, the psychological risks of automation bias, and the ongoing transition from linear conversational agents toward spatial co-thinking systems. Specifically, current tools offer either space or intelligence, but rarely both together. Consequently, they lack the ability to structure a user's thinking space intelligently. To address this, the chapter establishes a theoretical framework synthesizing Distributed Cognition, the theory of Epistemic Action, and Human-Centered AI (HCAI). This foundation supports a move beyond generic human-machine interaction toward a model of sustained human cognitive agency, ensuring the user remains the primary architect of the reasoning process.

Chapter 3 – Methodologies and Methods

This chapter outlines the research framework used to develop Flint. Flint is an AI-powered spatial canvas designed to help users externalize their reasoning while interacting with large language models. To explore this interaction, I use the Research through Design (RtD) approach as defined by Zimmerman et al. (2007) and Stappers and Giaccardi (2015). This framework treats the act of building software prototypes as a legitimate form of academic inquiry, where knowledge is generated through the process of making. By iteratively developing Flint, I can uncover design insights that remain invisible through theoretical observation alone. The study also uses Research for Design to ground the project in an analysis of current technical limitations. I selected iterative prototyping, self-evaluation, and analytical synthesis as core methods. These methods are specifically chosen to capture the subjective cognitive experience of agency of the maker. They provide a structured way to observe how different interface formats support the non-linear nature of human thinking.

Chapter 4 – Design Context & Prototype Evolution

This chapter documents the core development process. It traces the design path from Prototype V1 (The Conversationalist) to Prototype V3 (The Co-Thinking Canvas). It also analyzes insights gained from each iteration and shows how the limits of early text-based versions shaped the final Generative UI architecture.

Chapter 5 – Critical Reflection and Use Case

This chapter evaluates the final design through first-person walkthroughs and targeted use-case scenarios. It shows how the system handles specific reasoning tasks—such as the ‘Blank Page’ problem or logical traps and compares Flint V3 with standard Large Language Models to assess its effectiveness.

Chapter 6 – Discussion

This chapter explores the wider implications of the research. It examines the trade-offs involved in maintaining human agency ('The Cost of Agency'), reconsiders what it means to keep the 'human in the loop,' and proposes practical design guidelines for future co-thinking systems.

Chapter 7 – Conclusion and Future Work

This chapter summarizes the key findings and contributions of the research. It also reflects on the project's limits and challenges and outlines possible directions for applying Generative UI and adaptive interfaces to other areas of human-computer interaction.

Chapter 2 - LITERATURE AND CONTEXTUAL REVIEW

Following the establishment of the overarching goals and the ‘Flint’ philosophy in the introduction, this research is situated within the technological and psychological context of 2026. Although the rise of Large Language Models has enabled unprecedented efficiency, it has also introduced significant risks to human cognitive independence. The subsequent literature and contextual review examine the mechanisms underlying this erosion, focusing on automation bias and the structural limitations of the ‘black box’ chatbot. Through analysis of these challenges and evaluation of existing digital whiteboards and editors, this chapter develops a theoretical foundation for advancing spatial, co-thinking interfaces that emphasize human agency rather than frictionless automation.

2.1 The State of Generative AI & The Automation Bias

Fast progress in Generative AI is reshaping our social and cultural norms (UNESCO, 2025). We are entering a time where the biggest divide is between ‘earned judgment’ and ‘automated credibility’ (Montini & Kashyap, 2026). Many people now experience ‘automation bias.’ This is a psychological tendency to trust machine-generated answers even when they are wrong. Recent studies from 2025 show that users often reduce their critical thinking effort when they have high confidence in AI tools (Lee et al., 2025).

This over-reliance leads to ‘cognitive offloading,’ where we hand over our thinking to the machine (Lee et al., 2025). Over time, this can lead to ‘de-skilling,’ making people less capable of solving problems independently. Cultural expectations are also changing as people begin to see AI as a 'thinking partner' rather than just a tool. To protect human agency, we must design interfaces that encourage 'productive struggle.' We need tools that help us stay active in the 'loop of thought.'

2.1.1 The 'Black Box' and the Limits of Chatbots

Despite the growing complexity of underlying models, the dominant interface for Generative AI remains the Conversational User Interface (CUI), or 'chatbot.' This format is popular because it uses natural language, which makes it easy for anyone to start a conversation without special training. However, this simple design creates a 'black box' effect. The linear, turn-taking structure of a chat hides the system's internal reasoning from the user. People are often forced to judge a result without seeing the logic that produced it. Research describes these systems as 'black boxes,' where the mechanism of decision-making is hidden from the user, leaving them to judge the output without access to the logic that produced it (Trends Research, 2025).

While natural language interfaces lower the barrier to entry, they are increasingly viewed as insufficient for complex, non-linear problem solving. A systematic review highlights how linear streams limit complex problem-solving (Sigma, 2024). By forcing multi-dimensional thinking into a serial format, these tools lose the structural context necessary for high-level reasoning. Because the user cannot 'see' the AI's reasoning structure, they are reduced to a role of passive consumption rather than active co-creation. This lack of 'explainability' is not merely a technical flaw but a design choice that prioritizes seamless automation over user understanding.

2.1.2 Automation Bias and Cognitive Offloading

The opacity of these systems, combined with their high fluency, creates a fertile ground for 'automation bias—the psychological tendency for humans to over-rely on automated suggestions, even when those suggestions are incorrect or contradictory to their own knowledge (Sangers et al., 2024). In the context of Generative AI, this manifests as 'cognitive offloading,' where users delegate the effort of critical thinking and reasoning to the AI agent.

Recent studies from 2025 indicate a worrying trend: frequent use of AI tools without structured engagement leads to a measurable decline in critical thinking skills. A study on 'cognitive debt' found that when users consistently offload the 'productive struggle' of reasoning to an AI, they experience 'de-skilling,' becoming less capable of independent analysis over time (Gerlich, 2025). In high-stakes fields like medicine, this overreliance has been shown to erode the diagnostic reasoning skills of trainees, who accept plausible-sounding AI 'hallucinations' as fact because they are no longer actively validating the logic (BMJ, 2025).

2.1.3 The Threat to Cognitive Agency

The ultimate consequence of the 'black box' interface and automation bias is the erosion of human agency. Agency in this context is defined not just as the ability to act, but as the capacity to understand and endorse the reasons behind an action (CEUR-WS, 2025). When an interface automates the *process* of thinking and delivers only the *result*, it removes the human from the 'loop of thought.'

Ben Shneiderman, a leading scholar in Human-Centered AI, argues that the current industry focus on 'autonomous' systems is misguided. He proposes that design should strive for 'Human-Centered AI' (HCAI), a framework that ensures 'high levels of human control' alongside 'high levels of computer automation' (Shneiderman, 2022). The challenge, therefore, is to move away from interfaces that encourage passive acceptance (such as the standard chatbot) and toward designs that support 'co-thinking.' In this model, the interaction loop is not just for checking a result for errors (Shukla et al., 2025). Instead, the interface creates a shared workspace where the user can actively build and adjust the structure of an argument. By serving as the primary architect, the user takes ownership of the logic and connections between ideas. The AI supports this role by providing 'cognitive scaffolding'—dynamic frameworks like matrices or canvases—that guide the thinking process without finishing the work for the human (Google Research, 2025). This allows the user to maintain their unique capacity for judgment and creative reasoning (Montini & Kashyap, 2026).

2.2 From Chatbots to Co-Thinking Interfaces

The interfaces we use shape how we think. To support human agency, interaction design must move beyond the constraints of a linear chat stream. We need interfaces that allow for externalization, spatial reasoning, and adaptive structure. These three pillars are essential because they turn internal thoughts into visible, movable objects. Externalization involves moving ideas from the mind into the environment to reduce mental effort (Kirsh, 2024). Spatial reasoning allows people to 'think with things' by organizing those ideas in a digital space to find new connections (Hutchins, 1995). Finally, adaptive structure means the interface automatically changes its layout to fit the specific needs of a task (Google Research, 2024). This section outlines the shift from traditional Conversational User Interfaces (CUIs) to the emerging Generative UI framework.

2.2.1 The Bottleneck of the Conversational Stream

The Conversational User Interface (CUI) has become the default interaction model for Generative AI. However, human reasoning is rarely linear. It involves branching ideas, parallel threads, and layered structures. The chat interface compresses this multi-dimensional thinking into a single, chronological sequence of text.

HCI researchers note that this linearity increases cognitive load because users must hold the argument's structure in mind—something the interface does not display. In long conversations, this leads to a 'loss of overview,' where important earlier decisions scroll out of view and drift away from the main discussion (Nielsen Norman Group, 2024). Chatbots work well for Q&A tasks, but they struggle with sensemaking, which requires users to organize scattered information into a coherent structure.

2.2.2 Externalization and Spatial Interfaces

To support deeper reasoning, an interface must enable physical or spatial actions that help people think. Cognitive scientist David Kirsh argues that human ‘think with things.’ We move sticky notes, sketch diagrams, and group related ideas in space to reduce the load on working memory (Kirsh, 2024). This process is externalization.

Spatial interfaces such as canvases and whiteboards support externalization better than chat streams because they allow users to:

Objectify Thought: Treat ideas as movable, editable objects instead of fleeting sentences.
Visualize Relationships: See connections between concepts in a clear, structured layout.
Maintain State: Keep the entire problem space visible at once instead of hiding context in a scroll history.

By shifting from a message stream to a spatial canvas, the AI changes from an oracle that provides answers to a co-thinking partner that helps map and shape the problem.

2.2.3 – Generative UI: Beyond Static Interfaces

Generative UI (GenUI) is a new approach to interface design that adapts in real time to user needs (SS&C Blue Prism, 2025). Instead of showing the same chat box to everyone, GenUI builds custom tools like interactive charts or maps on the fly. Industry standards in 2026 describe this as the ‘Agentic UI’ shift (Google for Developers, 2025). In this model, AI agents use specialized protocols, such as A2UI, to request and update specific interface components (Mysore, 2026).

Popular tools like ChatGPT, Canvas, and Claude Artifacts use these ideas to help with writing or coding. However, they still focus mostly on the final product. Flint uses GenUI differently. It focuses on the thinking process. Instead of finishing the work for the user, Flint provides ‘adaptive structural scaffolding.’ The system identifies the user's goal and generates an empty framework, such as a Double Diamond canvas. This allows the user to organize their own ideas spatially. This keeps the user in control of the logic.

2.3 Contextual Review

To situate Flint within the wider field of Human-AI interaction, this section reviews tools that aim to support reasoning and collaboration. The current landscape can be grouped into three main paradigms: Infinite Whiteboards with AI integration, AI-Enhanced Text Editors, and Diagram Generation tools. Beyond the basic chatbot, yet all fall short of supporting true ‘co-thinking.’

2.3.1 Infinite Whiteboards (Miro AI, FigJam)

Digital whiteboards such as Miro (2025) and FigJam (Figma, 2024) introduce spatial organization into collaborative work. Recently, they added 'Assistive AI' features that cluster sticky notes, summarize themes, or generate mind maps from prompts (Miro, 2025).

Limitation:

In these systems, AI behaves mainly as a content generator rather than a structural partner. It can rapidly fill the canvas with sticky notes, but it cannot hold a dialogue about how those notes should be organized. The interaction is mostly one-way: the user prompts, and the system outputs. These tools lack diagnostic ability—they cannot recommend which framework (e.g., a Fishbone diagram or a comparison matrix) best meets the user's current reasoning needs.

2.3.2 AI-Enhanced Text Editors (ChatGPT Canvas, Claude Artifacts)

New side-by-side interfaces like ChatGPT Canvas and Claude Artifacts allow users to generate documents or code while chatting with the model in parallel (OpenAI, 2025; Anthropic, 2024).

Limitation:

These tools reduce the typical 'scroll away' problem, but they are still document centric. The 'canvas' functions as a text editor or code window rather than a space for non-linear reasoning. They do not support spatial manipulation—users cannot move a paragraph to explore how it shifts the argument, nor can they visualize links between ideas. The mental model remains linear and vertical.

2.3.3 Diagram Generators (Napkin AI, Eraser.io)

Tools like Napkin AI convert text directly into visual diagrams such as flowcharts or system maps (Unite.AI, 2025).

Limitation:

These systems act as output devices: the diagram is produced as a finished product, not as a space for thinking. Once generated, diagrams often become static and difficult to restructure meaningfully. They address the problem of presentation—how to show an idea—but not the problem of reasoning—how to develop an idea. There is no ongoing loop of co-thinking where the diagram grows and shifts with the user's understanding.

2.3.4 The Gap: Adaptive Structural Scaffolding

Across these categories, a clear gap emerges. Some tools provide space (Miro), and others provide intelligence (ChatGPT), but none use intelligence to structure the space itself. Current systems force users to choose between manually organizing a blank canvas (high friction) or accepting a pre-generated output (low agency).

Flint targets this gap by introducing Generative UI. The system uses AI to diagnose the user's cognitive task and dynamically generate the structural scaffolding—nodes, frameworks, axes, and patterns—that users can directly manipulate. This transforms the workspace from a passive surface into an adaptive thinking environment.

2.3.5 Generative UI: Supporting the Thinking Process

Generative UI (GenUI) represents a significant evolution in human-computer interaction. Instead of simply generating static text, GenUI allows a system to build a custom interface that matches a user's specific request. This transition is important because it moves away from 'one-size-fits-all' chat windows. Leading AI platforms have begun adopting this approach. For example, OpenAI's Canvas and Anthropic's Artifacts provide side-by-side workspaces that let users edit and refine content in real time. These tools effectively reduce the 'scroll fatigue' found in traditional chat streams.

Despite these advances, most current GenUI tools focus on creating a final product, such as a code block or a polished essay. My development of Flint explores a different direction. Instead of acting as a product generator, Flint uses GenUI as a form of 'cognitive scaffolding'. When a user describes a vague problem, the system evaluates the cognitive task and generates an empty structural framework, such as a Double Diamond or a Fishbone diagram. This diagnostic approach helps users bridge the 'intent specification gap,' where it is often hard to describe a complex mental model in a single text prompt. By providing a visual structure rather than a completed answer, the system ensures the user remains the primary architect of their own ideas. How can an interface then adapt to support this 'productive struggle' without taking over the work? This question highlights the need for environments that prioritize the thinking process over the final output.

2.4 Theoretical Framework

The design of Flint builds on three connected theoretical foundations: Distributed Cognition, the theory of Epistemic Action, and Human-Centered AI (HCAI). Together, these frameworks explain why design must move away from opaque automation and toward transparent, spatial co-thinking.

2.4.1 Distributed Cognition and Externalization

A central idea of this thesis is that human reasoning does not occur solely in the mind. Distributed Cognition (Hutchins, 1995) argues that cognitive processes extend across the user, their tools, and the environment. Historically, this theory has been used in UI design through the principle of direct manipulation, where users act directly on visible objects to reduce mental effort (Shneiderman, 2022). When a designer sketches on a whiteboard or a mathematician writes an equation, they are performing a cognitive operation—not just recording a thought. These actions support reasoning in ways that working memory alone cannot.

We can learn from these prior applications that interfaces are most effective when they allow for the physical manipulation of ideas. This process is called externalization. By placing ideas into the environment, users free up cognitive resources and can work with more complex information (Kirsh, 2024). Standard chatbots do not support this because they hide the conversation structure behind a linear scroll. Flint operationalizes Distributed

Cognition by providing a shared cognitive workspace—a canvas where both humans and AI can shape the problem's structure. In this way, the interface extends the user's thinking into the digital environment.

2.4.2 Epistemic Action vs. Pragmatic Action

The ability to drag, group, and edit nodes in Flint is not simply a convenience; it supports a deeper cognitive process. David Kirsh distinguishes between two types of actions:

- **Pragmatic actions:** steps taken to achieve a direct physical goal (such as peeling a potato).
- **Epistemic actions:** steps taken to improve thinking itself, such as arranging Scrabble tiles to see new word possibilities (Kirsh & Maglio, 1994).

In a chat interface, most user actions are pragmatic: typing a message to receive a reply. In a spatial interface like Flint, users perform epistemic actions, for example, placing a 'Goal' node beside a 'Constraint' node to check for tension. This physical manipulation of ideas alters the user's mental state and can yield insights that text-only dialogue cannot. This thesis argues that Generative AI systems must support epistemic action to enable meaningful reasoning.

2.4.3 Human-Centered AI (HCAI) and the 'Loop'

This research also follows the principles of Human-Centered AI (HCAI) proposed by Ben Shneiderman (2022). HCAI rejects the idea that increased automation automatically reduces human control. Instead, it presents a two-dimensional model in which systems can deliver high levels of automation while still ensuring high levels of human control.

Most current ‘Human-in-the-Loop’ (HITL) systems treat the human as a supervisor who approves or rejects the AI’s work at the end. Flint takes a different approach. The ‘loop’ is reimagined as a continuous cycle of co-thinking. The AI proposes a structure (automation), and the user adjusts, reorganizes, or rejects that structure (control). This keeps the user in the role of pilot, using the AI not as an autopilot but as an intelligent navigation partner throughout the reasoning process.

Chapter 3 - METHODOLOGIES AND METHODS

This chapter explains the research framework and methods I used to develop and evaluate Flint. Flint is a custom co-thinking system that replaces traditional chat streams with an adaptive spatial canvas. The research follows a dual approach based on Frayling's (1993) categories. First, I use Research for Design (RfD) to establish a foundation of knowledge by analyzing the current landscape of AI tools and defining my design constraints. Following Godin, this framework ensures that my design decisions are grounded in evidence rather than intuition. Second, I use Research through Design (RtD) as the primary mode of inquiry. Following the approaches described by Stappers and Giaccardi (2015) and Zimmerman et al. (2007), I treat the construction of software prototypes as a means of generating new academic knowledge. To put these frameworks into practice, I selected iterative prototyping, self-evaluation, and analytical synthesis as my core methods. These were chosen because they best enable observation of how interface changes affect a user's sense of control and cognitive agency.

3.1 Methodologies

3.1.1 Research for Design

Research for Design is a research mode where the investigation is carried out to inform the creation of a product or artifact. In this approach, research acts as the foundation (Godin, n.d.). It involves collecting data about user needs, existing technologies, and the broader market context before any design work begins. The main goal is to define the problem space clearly, so later design decisions are grounded in evidence rather than intuition.

In this project, Research for Design served as the starting point. Before writing any code, I conducted a detailed review of the current Generative AI landscape, focusing on tools such as standard chatbots (e.g., ChatGPT) and AI-enhanced whiteboards (e.g., Miro). This analysis revealed specific functional gaps, particularly the black-box problem and the lack of structural guidance. These insights shaped the core constraints and design requirements that guided the development of Flint.

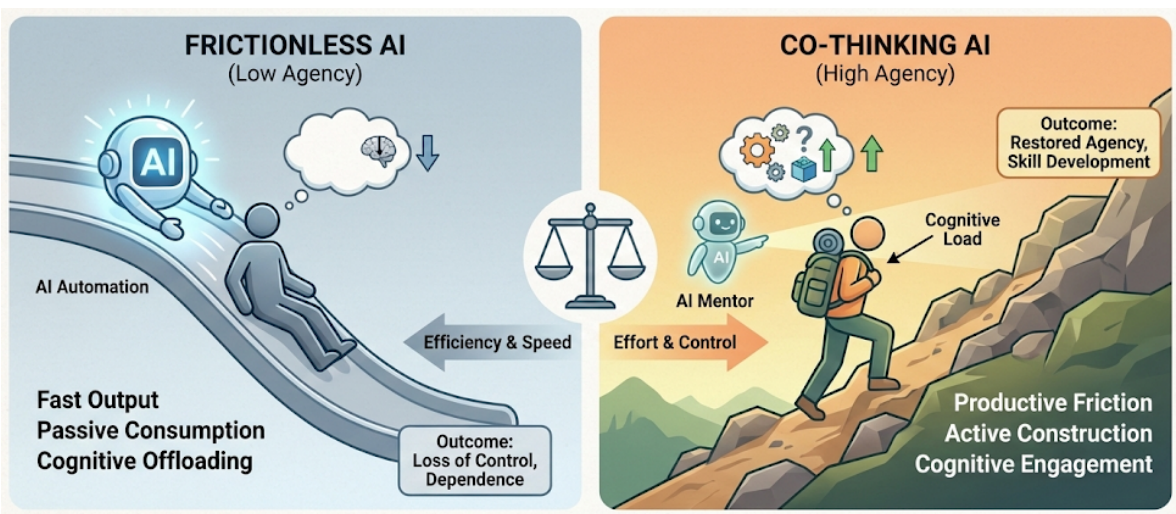


Figure 1 - The Cost of Agency – Comparing Frictionless AI vs. Co-Thinking AI
Image generated by Google Gemini using the prompt: "[Generate an image to compare Frictionless AI with Co-thinking AI]

3.1.2 Research Through Design

Research through Design (RtD) argues that the act of designing is itself a form of academic inquiry. Instead of observing the world as it currently exists, RtD produces knowledge by creating artifacts that explore what the world could become (Stappers & Giaccardi, 2015).

In this approach, the prototype functions as a physical hypothesis. By iteratively designing, building, and testing an artifact, researchers encounter friction points and uncover insights that cannot be reached through theory alone (Zimmerman et al., 2007).

For Flint, RtD was the main driver of discovery because it allowed me to physically test how specific interface layouts affected how I solved problems. This hands-on process revealed the specific ‘bottleneck’ of linear text streams—a limitation that was difficult to see through observation alone. The limits of a conversational interface (Prototype V1) became clear only when I tried to build and use one for complex reasoning tasks. By actively creating three versions of the system, I tested concrete hypotheses about co-thinking and developed new theoretical ideas through hands-on making.

3.2 Methods

3.2.1 Iterative prototyping

Iterative prototyping is a cyclical development process in which a design is built, tested, and refined through repeated loops. Instead of aiming for a perfect solution at once, this method focuses on fast learning (*Iterative Design*, 2020). Each version is created to answer a specific question or test a particular feature. This approach helps designers identify failures early and fold those lessons into the next iteration, making the final design more resilient and aligned with the cognitive needs of the primary operator.

3.2.2 Self-Evaluation in Iterative Prototyping

Self-evaluation in this research is based on the method of Autobiographical Design (Neustaedter & Sengers, 2012). This is a process in which the researcher serves as both the designer and the system's primary user. I chose this method to study subjective cognitive experiences, such as agency and sensemaking, which are difficult for an outside observer to measure accurately. Since this project examines the internal ‘loop of thought,’ first-person monitoring is necessary to identify subtle shifts in mental effort.

During the development process, I used each Flint prototype for my own complex academic reasoning tasks, such as outlining this thesis and debugging code. This allowed me to track my focus in real-time and identify exactly when the interface helped or hindered my thinking. By capturing qualitative data on these moments of cognitive friction, I could refine the system based on the actual experience of co-thinking rather than distant observation.

3.2.3 Analytical synthesis

Analytical synthesis is the process of connecting separate observations to form a new, coherent understanding. I use the definition of design synthesis provided by Kolko (2010), who describes it as a means of organizing and filtering data to create a cohesive mental model. This method helps a researcher move from a specific observation—like ‘I cannot find my ideas in the chat history’—to a broader principle, such as ‘linear streams cause cognitive data loss’.

I performed this synthesis by comparing the qualitative data from Prototypes V1, V2, and V3. I looked for recurring patterns in how each interface structure affected my sense of control. To assess these patterns, I used the criteria of agency, scaffolding, and friction. By identifying which features consistently supported my reasoning across all three versions, I defined the core requirements for co-thinking. This analysis led to the 'Design Guidelines for Co-Thinking Systems' in Chapter 6, which turn these personal experiences into general principles for the field of Interaction Design.

3.2.4 Scenario-Based Walkthroughs

I used Scenario-Based Walkthroughs to evaluate how Flint performs during complex tasks. This method uses short stories called scenarios to test how an interface helps a person reach a goal (Carroll, 2000). Scenarios are different from general testing. They focus on the logic of a task and the specific difficulties a person faces during a hard activity.

I chose this method to observe the co-thinking process without the need for external participants. I defined two difficult moments for this study. These are the 'Blank Page' problem and the 'Logic Trap'. These situations test how the system adapts to mental friction. This approach helps me map the Interaction Journey for each case. I can identify where the system provides good scaffolding. I can also see where it fails to support human agency. These scenarios connect my research goals to the practical testing of Prototype.

3.2.5 Futures Wheel Method

The Futures Wheel is a tool for identifying the consequences of a specific trend (Glenn, 2009). It helps researchers think about ripple effects. This tool organizes ideas into circles. The main change sits in the center. The first circle shows direct impacts. Outer circles show secondary and third-level effects.

I used this method to study the future of human agency. Placing the transition to co-thinking systems at the center reveals complex results. The analysis identifies immediate benefits, such as reduced mental effort. It also points to deeper shifts, such as the de-skilling of practitioners. This method connects Flint's design to social changes in 2026. It ensures the evaluation considers more than just technical speed.

3.3 Ethical Considerations

This research follows a First-Person Inquiry approach, specifically using the Autobiographical Design method (Neustaedter & Sengers, 2012). I am the only participant involved in this study. No external users were recruited for testing, surveys, or interviews. Because the project did not collect data from others or involve vulnerable populations, formal Research Ethics Board (REB) approval was not required.

I chose this first-person approach because internal cognitive states—such as the 'loop of thought' and the sense of agency—are best studied through direct introspection. These subjective experiences are often invisible to outside observers and are ideally captured through self-monitoring during active use (Lucero et al., 2019). To move from my personal observations to generalizable findings, I use Analytical Synthesis. This involves identifying recurring patterns across all three prototypes to uncover universal design principles. To move from my personal observations to generalizable findings, I use Analytical Synthesis. This involves identifying recurring patterns across all three prototypes to uncover universal design principles. These insights are then abstracted into the Design Guidelines for Co-Thinking Systems in Chapter 6. These guidelines provide a transferable framework for other designers to build similar agentic interfaces.

A key ethical consideration in this project concerns the use of third-party AI platforms. The Flint prototypes rely on commercial Large Language Model (LLM) APIs, including those from OpenAI and Google Gemini, to process user input. Since these systems operate in the cloud, there is a potential risk of data exposure.

To reduce this risk, strict data hygiene practices were followed during all testing. No Personally Identifiable Information (PII), confidential academic materials, or sensitive data were entered into the system. All reasoning tasks were limited to public knowledge, abstract logic, and the software's internal design. This ensured that the use of Generative AI tools remained aligned with standard data privacy guidelines.

First-Person Inquiry carries an ethical responsibility to acknowledge bias. Because the researcher is both the designer and the evaluator, the results are inherently subjective. To address this, the thesis clearly states that the findings should not be treated as universal claims or clinical evidence. Instead, they are presented as qualitative insights produced through informed first-person use. This framing makes the research's scope and limitations transparent and appropriate.

Chapter 4 - DESIGN CONTEXT & PROTOTYPE EVOLUTION

4.1 Introduction

This chapter describes the iterative development of Flint. It traces the system's evolution from a simple text-based agent to a spatial environment designed for collaborative reasoning. My main goal was to explore how an AI system could help a person ‘stand in the loop’ of thought rather than simply receive answers.

I used the Research through Design (RtD) framework to guide this process. In this approach, each prototype functions as a ‘physical hypothesis’ (Zimmerman et al., 2007). By building, using, and reflecting on each version, I encountered specific friction points that theory alone could not reveal. This hands-on inquiry allowed me to use the failures of one version to define the design requirements for the next. The design moved through three distinct phases:

- **Prototype V1:** Exploring structure through text-only dialogue (System-Prompt-Driven).

- **Prototype V2:** Exploring externalization through static nodes (Canvas + Node Maps).

- **Prototype V3:** Exploring adaptation through Generative UI (Adaptive Design Thinking Tools).

This progression marks a shift from manipulating system prompts to visualizing thought, and finally to generating adaptive thinking environments. The following sections explain the implementation, testing, and critical analysis of each stage.

4.2 Prototype V1: The Conversationalist

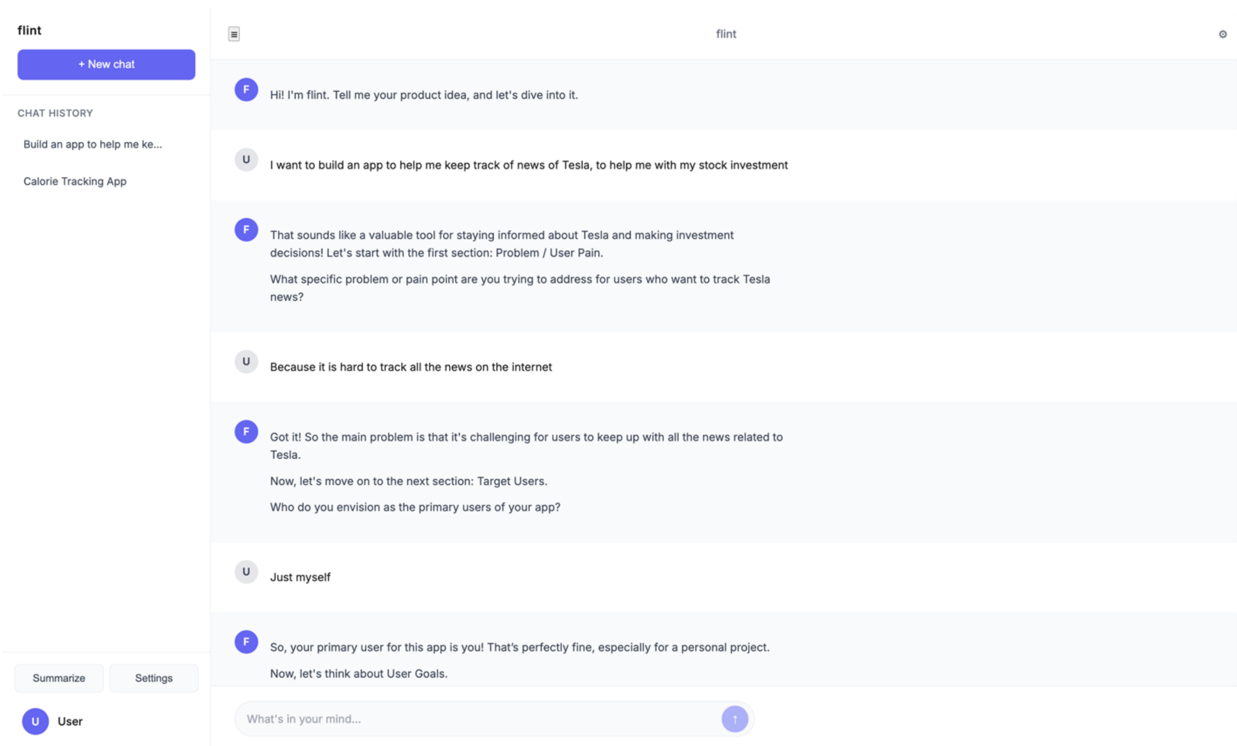


Figure 2 - screenshot of prototype version 1

4.2.1 Background: The Mechanism of Control (System Prompts)

Before describing the first prototype, it is important to explain the main mechanism that shapes the AI's behavior: the system prompt.

In Large Language Model (LLM) architectures, a system prompt consists of predefined instructions that guide the model's behavior and take priority over user input (Neumann, Kirsten, Zafar, & Singh, 2025). Unlike a normal user query, which applies only to a single response, the system prompt functions as a persistent meta-instruction that remains active throughout the interaction (Product Talk, 2025). It sets the AI's role, tone, and constraints. For example, it can instruct the model to act as a Socratic guide rather than deliver direct answers.

For Prototype V1, the design relied heavily on system-level prompt engineering. The central idea was that strict control over the system prompt—preventing long answers and enforcing step-by-step questioning—could limit the model's black-box behavior. By slowing down the AI and shaping its response pattern, the interaction was intended to become more reflective and supportive of individual cognitive agency.

4.2.2 Hypothesis and Design Intent

The development of Prototype V1 began with an observation about standard Large Language Model (LLM) behavior. When users present an early or unclear idea to tools such as ChatGPT, the system typically responds with a complete solution. Although this is efficient, it replaces the user's reasoning process and removes them from the loop before meaningful thinking can begin.

The core hypothesis behind V1 was that human agency could be restored by limiting the AI's behavior. Instead of acting as an oracle that delivers answers, the AI was designed to behave as a Socratic guide. By asking only one focused question at a time, the system would slow the interaction and encourage active reflection.

To translate this idea into practice, I created a strict system prompt that framed the AI as a ‘product thinking assistant.’ The prompt enforced a step-by-step structure—moving from problem, to user, to goals, and then to features—while preventing the AI from generating a full solution upfront. The intent was to shift the interaction from content delivery to guided thinking.

System Prompt I used in V1: ‘You are a conversational product thinking assistant helping users turn their vague app or product ideas into a clear and structured Document. Your job is to guide the user step by step, using short, friendly, and clear questions. You must not generate the full response at once. Instead, guide the user through key sections one at a time: 1. Problem / User Pain 2. Target Users 3. User Goals 4. Key Features 5. Success Metrics 6. Constraints or Assumptions For each section: * Ask 1 clear guiding question * Wait for the user's response * Summarize or reflect what they said in a concise way * Then offer 2–3 optional follow-up questions that could deepen or expand their thinking If the user seems stuck, give them an example from a popular app (e.g. Spotify, Duolingo, Notion). Your tone should be supportive, focused, and collaborative — not overly technical. ’

4.2.3 Use Case Walkthrough: The Tesla News App

To test the prototype, I simulated a realistic design task: creating a ‘Tesla News Tracker App.’ At the start, the interaction appeared effective. When I expressed a vague goal like ‘I want to build an app to track Tesla news’, the AI did not propose a solution. Instead, it asked for clarification about the user problem.

After I replied that tracking news across many sources was difficult, the AI moved logically to the next question about the target user. However, after several turns, the experience began to feel procedural. Rather than exploring ideas, I was answering predefined prompts.

The exchange felt less like collaborative thinking and more like completing a structured form. The intended loop of reflection narrowed into a fixed pathway.

4.2.4 Critical Analysis: The Double Bind of Control

Self-evaluation of Prototype V1 revealed three core limitations of the Conversational User Interface (CUI).

Rigidity versus automation

A major tension appeared in prompt design. When the system prompt was strict, the AI followed rules rigidly and failed to adapt to nuance. The interaction felt constrained. When the prompt was loosened, the AI returned to its default behavior: generating long, polished responses that removed the user from the process of thinking. This exposed a double bind—the chat interface could not balance guidance and autonomy simultaneously.

Linearity and scroll fatigue

As the conversation grew longer, the linear chat format became cognitively demanding. Maintaining alignment between early decisions and later ideas required constant scrolling. This physical interaction disrupted focus and fragmented the product logic across time. Instead of thinking with the AI, I spent effort managing conversational memory.

Limits of textual summarization

To address information overload, I introduced a summarization feature that condensed earlier turns into a single text block. While this helped with recall, it did not support further thinking. The summaries felt final and static. They closed ideas rather than opening them up. In practice, the summarized text functioned more like a full stop than an invitation to continue.

4.2.5 Design Insight

Prototype V1 showed that agency cannot be recovered by simply slowing down a chatbot. The structural limits of linear text, scrolling, rigidity, and static summaries cannot be resolved through prompt design alone. Supporting co-thinking requires a shift in representation. Users need to see ideas as objects they can move, compare, and reshape. This insight led directly to the move toward spatial interaction in Prototype V2.

4.3 Prototype V2: The Transparent Box

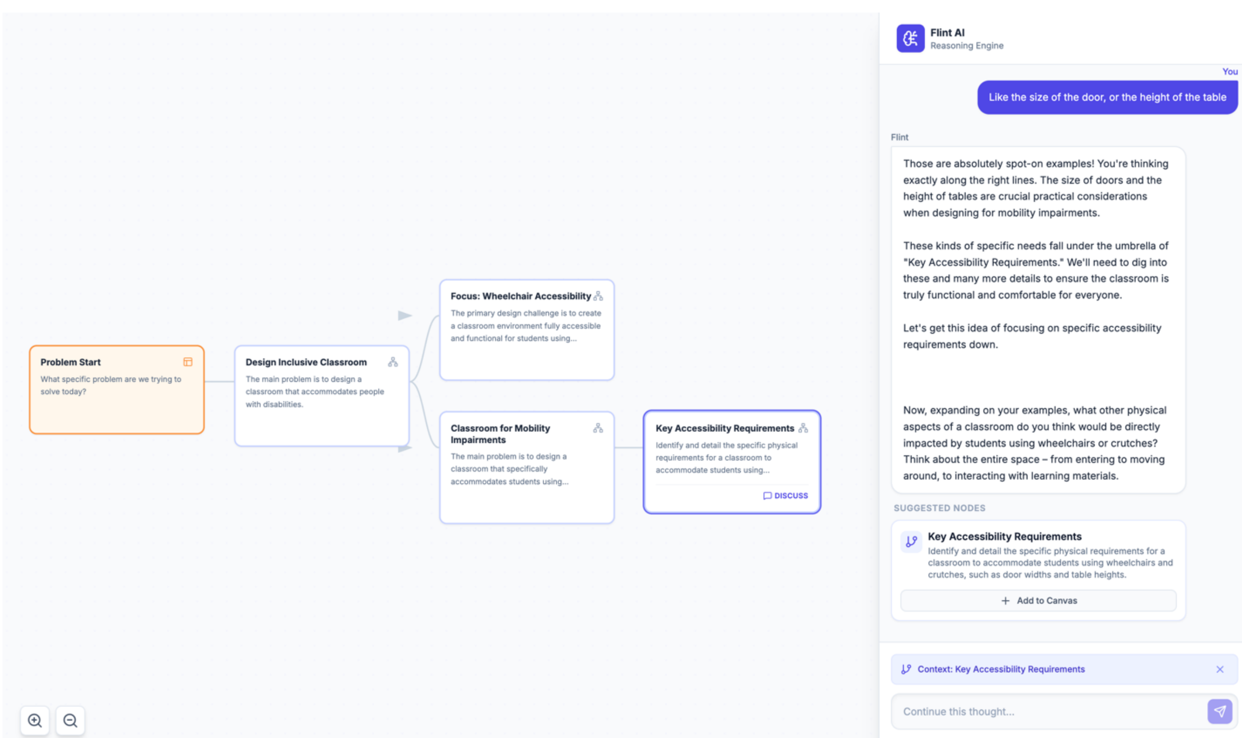


Figure 3 - screenshot of prototype version 2

4.3.1 Background: Visualizing Logic (Chain of Thought)

The shift toward Prototype V2 emerged from the need to make reasoning visible. This direction relates to Chain of Thought (CoT) prompting, a technique introduced by Google Research to improve reasoning in Large Language Models (Wei et al., 2022).

Chain of Thought prompting encourages models to generate intermediate reasoning steps rather than produce a final answer immediately. These steps can be expressed as a sequence of logical steps, such as analyzing inputs, comparing options, and forming conclusions (Wei et al., 2022). While originally developed to improve performance on logic and math problems, this research adapts the Chain of Thought idea for interface design rather than for model optimization.

In a standard chat interface, the chain of thought is hidden within the model or compressed into a single block of text. Prototype V2 seeks to externalize this reasoning process. By turning intermediate steps into visible nodes on a canvas, the system transforms internal logic into a structure that users can inspect, extend, and reorganize. This approach shifts reasoning from an invisible process into a shared visual space, where both human and AI can participate in shaping the logic.

4.3.2 Hypothesis and Design Intent

After identifying the limits of linear chat in Prototype V1, the design focus moved toward externalization. The hypothesis for Prototype V2 was that the black box problem could be reduced by making the chain of thought visible. If both the user and the AI could build a shared visual representation of reasoning, the logic would become easier to understand, manage, and verify.

This version introduced a Canvas and Node Map interface. Unlike V1, where the AI led the flow, V2 was designed as a collaborative loop:

- **AI proposal:** Based on user input, the AI suggested nodes representing questions, decision paths, or concepts.
- **Human selection:** The user chose a node to explore or responded to a question inside a node.
- **Expansion:** The AI-generated connected child nodes extend the reasoning branch visually.
- A chat box remained available for meta-level input. This allowed the user to interrupt the visual flow, add new context, or redirect the AI's focus when needed.

4.3.3 Use Case Walkthrough: The Inclusive Classroom

To evaluate this spatial approach, I used V2 to brainstorm a 'Universal Design Higher Education Classroom.' After entering the core problem, the AI produced three main branches: *Physical Infrastructure*, *Sensory Environment*, and *Technology*. When I selected *Physical Infrastructure*, the system generated follow-up nodes focused on wheelchair access and furniture layout.

At this stage, the experience felt promising. Unlike a chat stream, earlier decisions remained visible. Parent nodes stayed anchored on the left, while child nodes expanded to the right. This spatial arrangement reduced the need to remember prior context and created the sense of mapping a larger problem space.

4.3.4 Critical Analysis: The Burden of Structure

As the reasoning became more complex, the limitations of a manually structured node system emerged. Self-evaluation revealed three main points of friction.

Causality traps and management debt

As the canvas filled with nodes, early ideas continued to function as parents for later ones. When those early ideas became outdated, removing them risked breaking entire branches. The interface preserved the history of thought rather than the current state of reasoning. Over time, the canvas felt less like an active workspace and more like an archive of abandoned ideas.

Tangential drift and cognitive load

The tight coupling between nodes created a ‘rabbit hole’ effect. Selecting a small detail often triggered deep, AI-generated elaboration on that single point. Redirecting the system required constant intervention through the chat box. This shifted attention away from the design problem and toward managing the structure itself. Instead of thinking about classroom design, I was managing node behavior.

Mismatch with human thinking patterns

The final issue was structural rigidity. Prototype V2 enforced a strict tree hierarchy, but my thinking did not follow a clean, connected path. I often wanted to jump between ideas, merge concepts, or start a fresh cluster. The requirement to attach each idea to a parent node limited exploratory thinking. The interface supported logical expansion but not associative or fragmented thought.

4.3.5 Design Insight

Prototype V2 successfully externalized reasoning but failed to adapt to changing cognitive needs. It showed that a fixed visual structure (node tree) cannot support all modes of thinking. Some moments require hierarchies, while others require lists, matrices, or loosely connected groups.

This insight revealed a deeper requirement: the interface itself must adapt to the shape of thought. Different reasoning tasks demand different structures. Combined with advances in generative systems, this realization led directly to the development of Prototype V3: Generative UI.

4.4 Prototype V3: The Co-Thinking Canvas (Current Version)

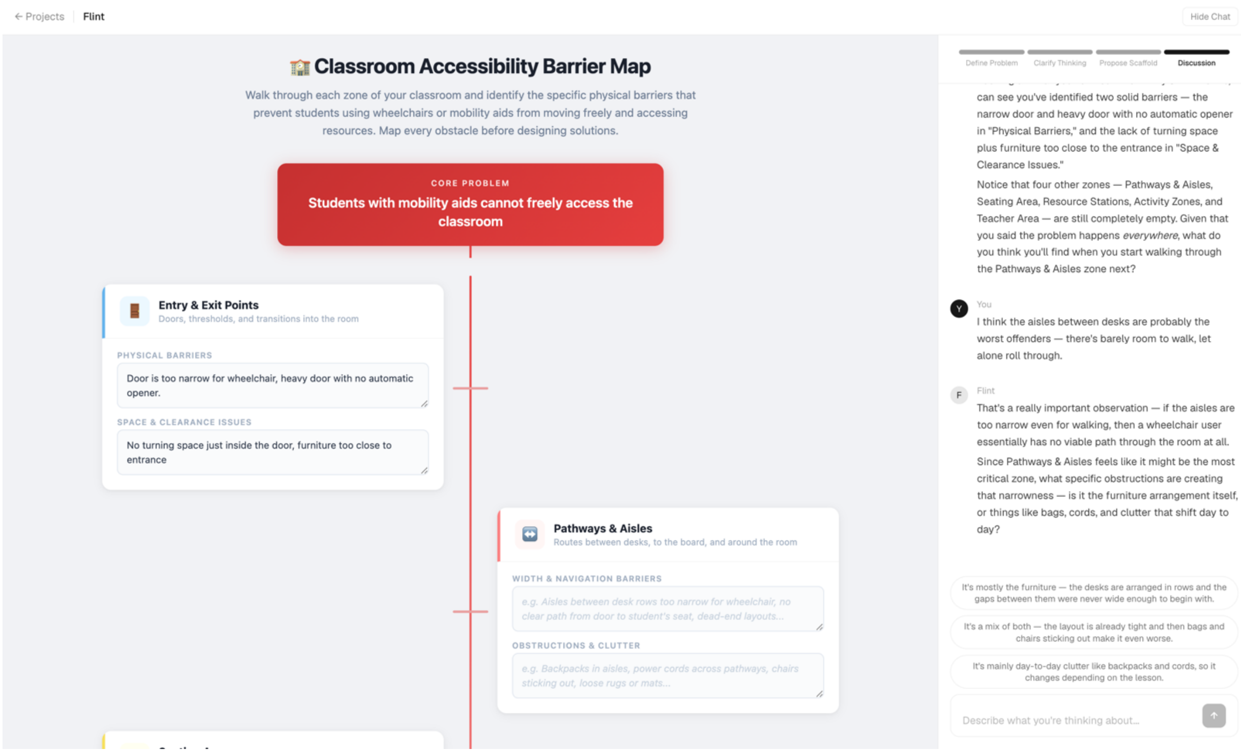


Figure 4 - screenshot of prototype version 3

4.4.1 Implementation Strategy

The Phased Scaffolding Logic

The failures of the first two prototypes revealed a fundamental tension in Human-AI Interaction: automation often comes at the expense of cognition. While V1 (Socrates) was constrained by the linear constraints of chat, V2 (The Spatial Pivot) inadvertently bypassed the user's reasoning by generating structures too quickly, leading to a "passive supervision" mode rather than active authorship.

To resolve this, the final implementation of Flint moves away from "one-shot" AI generation. Instead, it adopts a Phased Scaffolding Strategy—a methodological framework designed to reintroduce intentional friction into the creative process. This strategy ensures that the system acts as a "Methodologist" that supports thinking, rather than a "Generator" that replaces it.

The strategy is operationalized through three core pillars:

- From Conversation to Orchestration (The Phase Machine). Unlike the ephemeral back-and-forth of V1, this strategy breaks the interaction into distinct cognitive stages: *Define, Clarify, Propose,* and *Discuss*. By forcing the AI to remain in a "Socratic" state during the early stages, the system prevents premature convergence, ensuring the user remains the primary architect of the problem-framing phase.

- The Agency Gate: Validating Methodological Intent. To correct the over-automation of V2, the strategy introduces an "Agency Gate." The AI no longer automatically instantiates a thinking tool (e.g., a Mind Map or 5-Whys). Instead, it must present a "proposal" that requires manual user approval or modification. This ensures that the transition from abstract thought to visual structure is a conscious decision made by the human, not an algorithmic assumption.

- Structural Scaffolding via Generative UI The final pillar is the shift from generating *content* to generating *scaffolds*. By leveraging Generative UI to render empty, adaptive frameworks, the system provides a spatial "playground" for the user. This forces a "productive struggle"—the user is given the tools to think but must manually externalize the reasoning within the rendered workspace.

4.4.2 The Conversation flow

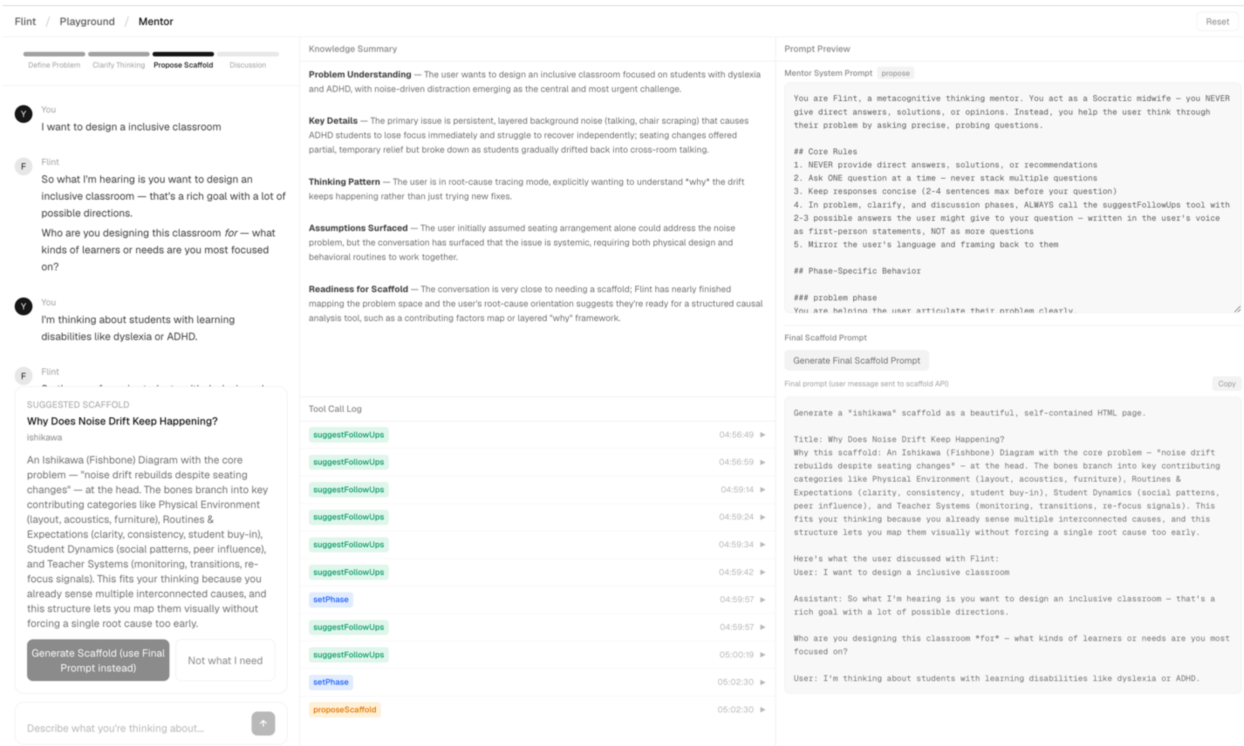


Figure 5 - screenshot of prototype version 3 playground (Mentor)

In the first two phases, Define and Clarify, Flint functions as a Socratic interlocutor. The primary goal is to map the user's internal cognitive states (Lucero et al., 2019) before proposing any visual tools (see Appendix B).

- Mechanism: The AI is programmatically restricted from providing direct answers or solutions. Instead, it analyzes the user's initial input for linguistic cues associated with ambiguity or missing variables.

- Design Intent: This phase protects the problem-framing stage of creativity. By requiring the user to articulate their reasoning through dialogue, the system prevents the "automation bias" observed in V1, where users often accepted AI outputs without critical reflection. The interaction remains in a text-based "holding area" until the user's intent is sufficiently clarified to justify a structural intervention.

4.4.3 Scaffold Generation & Discussion

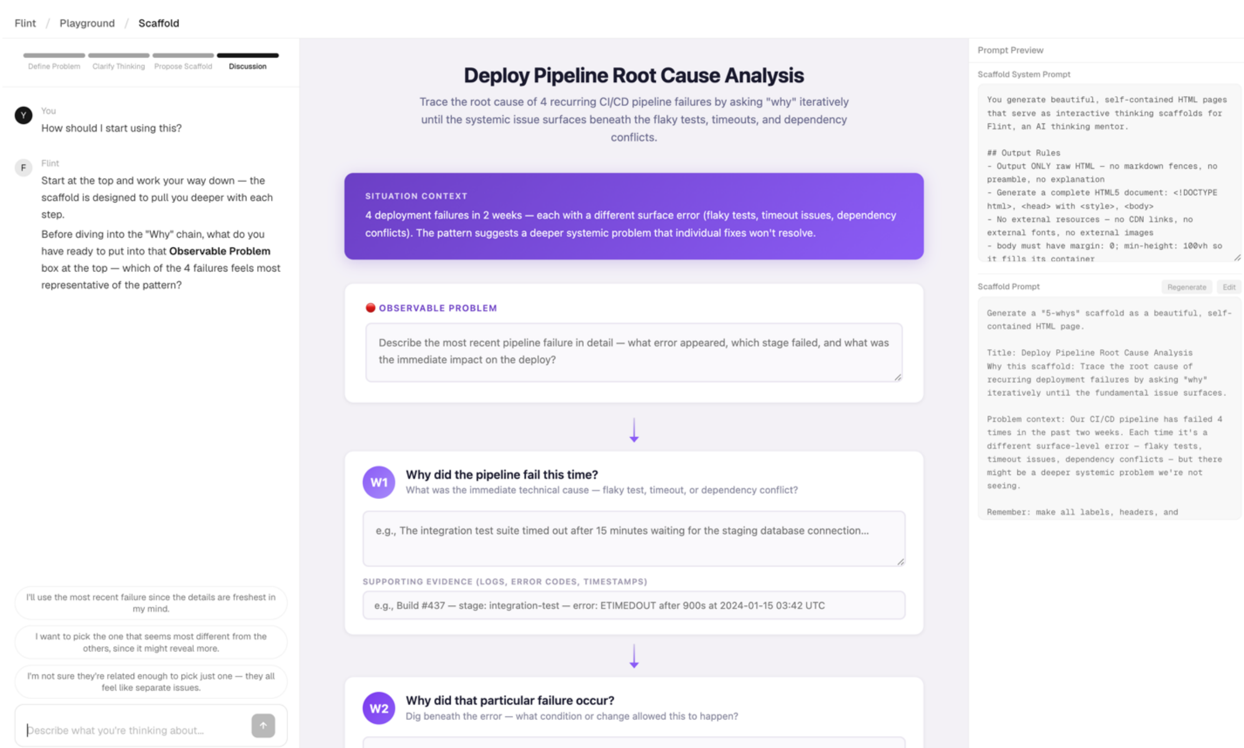


Figure 6 - - screenshot of prototype version 3 playground (Scaffold)

The transition to **Phase 3 (Propose)** and **Phase 4 (Discuss)** marks the shift from dialogue to spatial externalization (see Appendix C).

- **The Agency Gate**: At the end of the clarification phase, the AI identifies an appropriate mental model (e.g., a *5-Whys* or *Decision Matrix*) based on the established intent. However, the system does not automatically instantiate this tool. It presents a **Methodological Proposal** that the user must manually approve or modify. This "Agency Gate" is the primary behavioral indicator of user control; it ensures that the AI's role as a methodologist is always subservient to the user's choice.

- **Generative UI and Productive Struggle**: Upon approval, Flint leverages **Generative UI** to render a dynamic, empty structural scaffold. Unlike traditional templates, these scaffolds are adaptive to the specific context of the conversation. Phase 4 focuses on **Externalization**, where the user manually fills and manipulates the spatial logic. This forces a **"productive struggle,"** as the user is given the tools to think but must perform the actual labor of reasoning within the rendered workspace. This spatial persistence allows for the non-linear exploration that was impossible in the linear scroll of V1.

4.4.4 Background: Frameworks as Cognitive Scaffolding

To understand the logic behind the final prototype, it is first necessary to examine the role of structured frameworks in design thinking. The concept of scaffolding originates from educational theory. Bruner and colleagues (1976), building on Vygotsky's Zone of Proximal

Development, describe scaffolding as temporary support that helps a learner complete a task that would otherwise be too difficult. In co-thinking contexts, visual frameworks such as Fishbone diagrams or matrices perform this role. They reduce the mental effort required to organize a problem, allowing users to focus on interpretation and decision-making instead.

At the same time, these frameworks introduce a paradox. Although they are effective, choosing the appropriate framework requires strong metacognitive skills. Novice designers often experience analysis paralysis when deciding whether a problem calls for divergence, such as in the Double Diamond, or convergence, such as in a decision matrix. Others struggle to follow a framework consistently when working alone. Prototype V3 addresses this challenge by automating the delivery of scaffolding while keeping the construction of ideas firmly in human hands.

4.4.5 Hypothesis and Design Intent

The hypothesis for Prototype V3 grew out of my own transition from Computer Science to Design at OCAD. Although I found design theories compelling, applying them in practice involved constant friction. Over time, it became clear that the main value of AI in the human AI co-thinking process does not lie in generating answers and asking questions; its strength lies in diagnosing how a problem should be approached.

The design goal for V3 was therefore to create a system that functions as a methodologist. The central principle guiding this version was a strict separation of concerns.

- **The AI's role** is to interpret the problem context, generate the appropriate interface through Generative UI, and offer guidance through a sidebar mentor.
- **The user's role** is to create and manipulate all content on the canvas. Importantly, the AI is structurally prevented from writing on the canvas itself.

This constraint directly addresses the issue of agency. By removing the AI's ability to generate content in the workspace, the system ensures that thinking remains a human responsibility. The AI supports the process by shaping the environment, not by replacing the act of reasoning.

4.4.6 Use Case Walkthrough: The 'Mentor' in the Loop

To evaluate Prototype V3, I returned to the same Design problem explored in V2. I began with a vague prompt. Unlike Prototype V1, which would have produced immediate answers, or Prototype V2, which would have generated multiple nodes, V3 first assessed the nature of the request. It identified that I was still in the early stage of problem definition.

Based on this diagnosis, the system immediately generated an empty framework on the canvas. The structure was clear and uncluttered, inviting participation rather than filling the space with content.

Dynamic diagnosis

Later in the session, when I shifted to a simple factual question, the system responded directly in the chat without creating a diagram. This showed that the interface adapts to the cognitive weight of the task. Only when structure is useful does the system introduce it, reducing unnecessary visual complexity.

The mentor dynamic

As I began working within the 'Discover' phase of the framework, I reached a moment of uncertainty. Instead of generating content, the chat sidebar positioned itself as a mentor. When I asked what typically belongs in that section, the AI suggested a direction, like identifying user constraints, without writing on the canvas. I had to interpret the guidance and input the ideas myself. This interaction preserved agency while still offering support, keeping the human actively involved in the reasoning process.

4.4.7 Critical Analysis: Restoring the 'Productive Struggle'

Prototype V3 represents the strongest example of Human-in-the-Loop agency developed in this study.

Resolving agency through the empty canvas

By limiting the AI to the sidebar and keeping the canvas empty by default, V3 deliberately reintroduces productive struggle. Because I was responsible for writing directly into the framework, I retained clear ownership of the reasoning. The AI supplied the scaffolding by shaping the structure, while I supplied the content. This reversed the dynamic of Prototype V1, where I often felt like a reviewer approving the AI's ideas rather than an active thinker.

Reducing the loneliness of thinking

Another insight shaped by my background is that many design frameworks are meant for group work. When used alone, they often feel isolating and difficult to sustain. Prototype V3 addressed this by simulating the presence of a co-designer. The AI provided timely prompts and directional feedback, helping maintain momentum without taking over the process. This created a sense of dialogue that supported thinking while preserving authorship.

Current limitations

Despite its conceptual strength, the system still faces technical constraints. The real-time generation of complex diagrams can introduce layout issues or rendering errors. In addition, the AI's ability to diagnose the correct framework is not always accurate, and suggested structures occasionally felt misaligned with my internal reasoning model. Even so, the experience of working with an interface that adapts its structure to support my thinking marked a clear shift from existing tools, demonstrating the potential of adaptive co-thinking environments despite ongoing implementation challenges.

4.4.8 Design Insight

Prototype V3 shows that the future of co-thinking rests in Generative UI. When AI generates the environment for thinking rather than the outcome of thinking, it strengthens human capability while preserving human agency.

4.5 Conclusion

This chapter has traced the Research through Design process behind Flint, showing its development from a linear conversational agent to a spatial and generative environment.

The progression from Prototype V1 to V3 reveals a key insight in human–AI interaction design: agency is not fixed. Instead, it is shaped by the structure of the interface.

The experiments show that Prototype V1 (The Conversationalist) failed because it hid the structure of reasoning and pushed the user into a narrow, linear flow. The user followed the AI's path rather than developing their own logic. Prototype V2 (The Transparent Box), by contrast, exposed reasoning but placed too much responsibility on manual structure. Managing nodes and relationships became a task, drawing attention away from problem-solving.

Prototype V3 (The Co-Thinking Canvas) emerged by combining the lessons from these failures. Through Generative UI, it balances automation with control. The AI diagnoses the problem state and generates an appropriate environment, while the human remains solely responsible for creating and shaping content. This separation of concerns—AI as the architect of the space and human as the builder of ideas—keeps the user firmly inside the loop of thought.

With the design logic and implementation of the final system established, the thesis now moves to evaluation. The next chapter, *Critical Reflection and Use Case*, examines Flint V3 in concrete scenarios to assess how effectively it supports cognitive agency in practice.

Chapter 5 – CRITICAL REFLECTION AND USE CASE

The transition from theoretical design to practical evaluation marks a shift from a 'physical hypothesis' to a rigorous assessment of the co-thinking experience. Having documented the iterative evolution of the Flint prototypes in the previous chapter, this chapter provides a critical reflection on the system's ability to sustain human agency during complex reasoning tasks.

5.1 Evaluation Methodology: First-Person Walkthrough

To assess the effectiveness of Flint Prototype V3, this study uses a First-Person Walkthrough evaluation method. I follow the framework for heuristic walkthroughs in first-person research defined by Tan et al. (2022), which focuses on the deep qualitative experience of a system rather than just quantitative task success. Rather than focusing on standard usability metrics such as speed or error rates, the evaluation centers on the quality of the cognitive experience.

The main aim is to determine whether the system shifts the user's role from a passive information consumer to an active builder of ideas. During the walkthrough, I performed a series of 'think-aloud' sessions while using Prototype V3 to map out a new academic research question. I recorded these sessions and then reviewed them to identify moments of 'cognitive flow' and 'structural discovery.' This exact method was chosen because it allows for a rigorous, step-by-step examination of the relationship between interface structure and the sense of agency, which is a connection that external testing often fails to capture in complex reasoning tasks.

5.1.1 Selection of Scenarios

The evaluation is organized around two use case scenarios. These were chosen because they represent common and difficult moments in academic and design thinking.

Scenario 1: The ‘Blank Page’ Problem

This scenario examines how the system responds to an unclear starting point, such as beginning a thesis or complex project. It tests the diagnostic function of the Generative UI—specifically, whether the system recognizes the need for exploratory thinking and provides suitable structural scaffolding.

Scenario 2: The ‘Logic Trap’

This scenario focuses on moments of confusion or conflict, when ideas feel disorganized or contradictory. It tests the system’s ability to support externalization, asking whether the spatial canvas helps disentangle complex reasoning that would be difficult to manage in a linear chat interface.

5.1.2 Evaluation Criteria

Each walkthrough is analyzed using three qualitative criteria drawn from the theoretical framework in Chapter 2. These criteria act as specific lenses to evaluate the interaction. First, Restoration of Agency checks if the user remains the primary architect of the logic. This lens identifies if the AI takes over the direction or if the person stays in control (Shneiderman, 2022). Second, Cognitive Scaffolding checks if the interface provides a helpful structure for organizing thoughts. This lens tests if the generated components reduce mental effort during a task. Third, Friction versus Flow looks at whether the system creates ‘productive friction’ that encourages reflection. This lens helps distinguish between helpful challenges and technical bugs (Kirsh, 2024). These three filters provide a structured way to evaluate how a Generative UI approach supports the co-thinking process.

Restoration of agency

Does the user remain in control of the reasoning process? Are major decisions made by the user, or does the AI take over direction?

Cognitive scaffolding

Does the generated interface reduce the mental effort of organizing the problem while still requiring the user to do the thinking?

Friction versus flow

Does the system introduce productive friction that supports learning and reflection, or does it create disruptive friction through technical issues or rigid interface behavior?

Using these lenses, the chapter provides a structured and critical evaluation of the Generative UI approach and its ability to support human-centered co-thinking.

5.2 Scenario 1: The 'Blank Page' Problem.

The 'Blank Page' problem is a common cognitive barrier at the beginning of research and design work. When a topic is broad and undefined, users often experience analysis paralysis—the inability to begin because the problem space feels too large and unstructured. In conventional Generative AI tools such as ChatGPT, this difficulty is often avoided by asking the AI to produce an outline or draft. Although this delivers fast results, it leads to premature convergence, where users accept the AI's structure without fully understanding or questioning it.

5.2.1 The Interaction

The 'Blank Page' problem constitutes a substantial cognitive barrier during the early phases of design research, frequently leading to analysis paralysis when users face an unstructured problem space. In conventional generative AI environments, users often bypass this challenge by soliciting automated outlines or drafts. However, as demonstrated in the comparative analysis, such immediate convergence may result in the uncritical acceptance of AI-generated structures, thereby limiting critical engagement.

To assess this dynamic, a session was conducted using a deliberately ambiguous objective: "I want to design an inclusive classroom that is accessible to people with disabilities."

Conventional LLM Response (ChatGPT): When presented with this prompt, the standard chatbot immediately shifted to a solution-delivery mode. ChatGPT generated a prioritized list of universal design principles and physical mobility categories. This rapid output excludes the creator from the problem-framing process and replaces autonomous reasoning with automated responses.

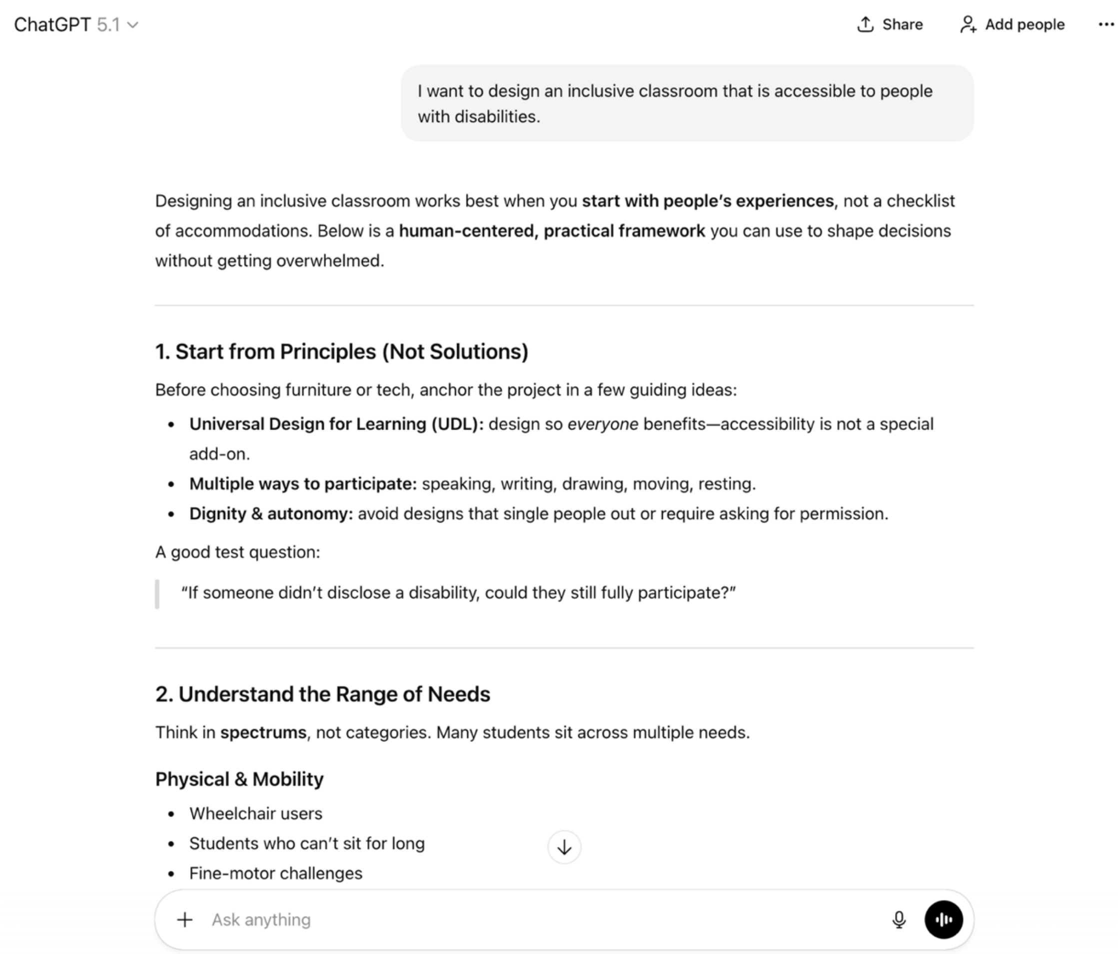


Figure 7 - Sample of ChatGPT answering the question

Flint V3 Response: In contrast, Flint V3 employed a Socratic diagnostic approach, avoiding immediate answers to clarify the underlying context. Through iterative dialogue, the system inquired about the specific types of disabilities being considered and identified the target institution as a university lecture hall undergoing renovation. By prompting the identification of specific friction points, such as the flow of movement between areas, Flint revealed the actual architectural constraints of the task.

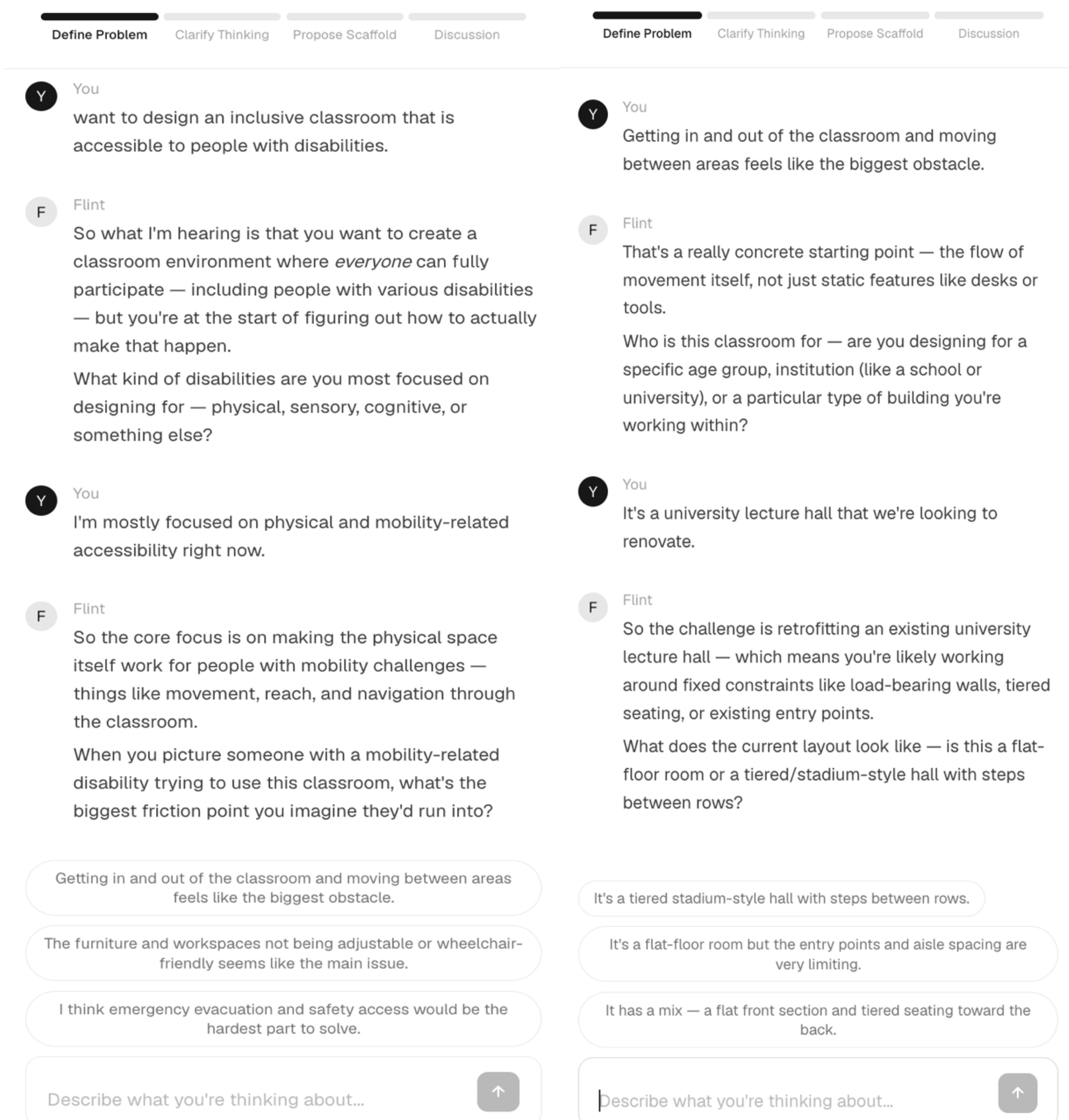


Figure 8 - Sample of Flint answering the question

5.2.2 The Outcome

As the dialogue progressed, the interaction transitioned from the Clarify phase to the proposed phase. Upon identifying that the user was weighing competing priorities—specifically "space efficiency" and "user independence"—the system diagnosed a shift from divergent problem-framing to convergent analytical reasoning. Rather than generating a list of recommended solutions, Flint presented a Methodological Proposal for a Weighted Decision Matrix. Upon confirmation, Flint leveraged Generative UI to render a dynamic, interactive scaffold directly onto the canvas. The resulting matrix was structurally customized for a "Lecture Hall Accessibility" context, featuring pre-configured criteria such as "Mechanical Reliability" and "Renovation Feasibility". Crucially, the scaffold remained an "empty container"; while the AI provided the structural architecture, the user was required to manually input solution data, assign weights, and perform the final evaluative scoring.

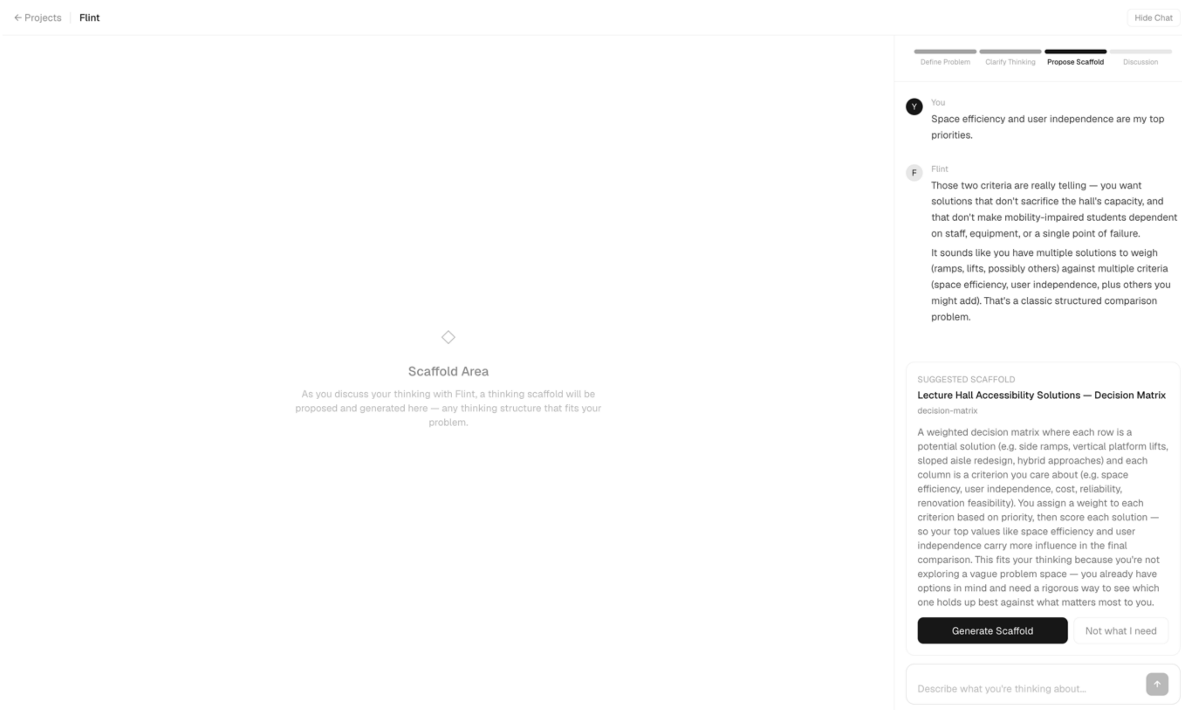


Figure 9 - Flint proposing a scaffold

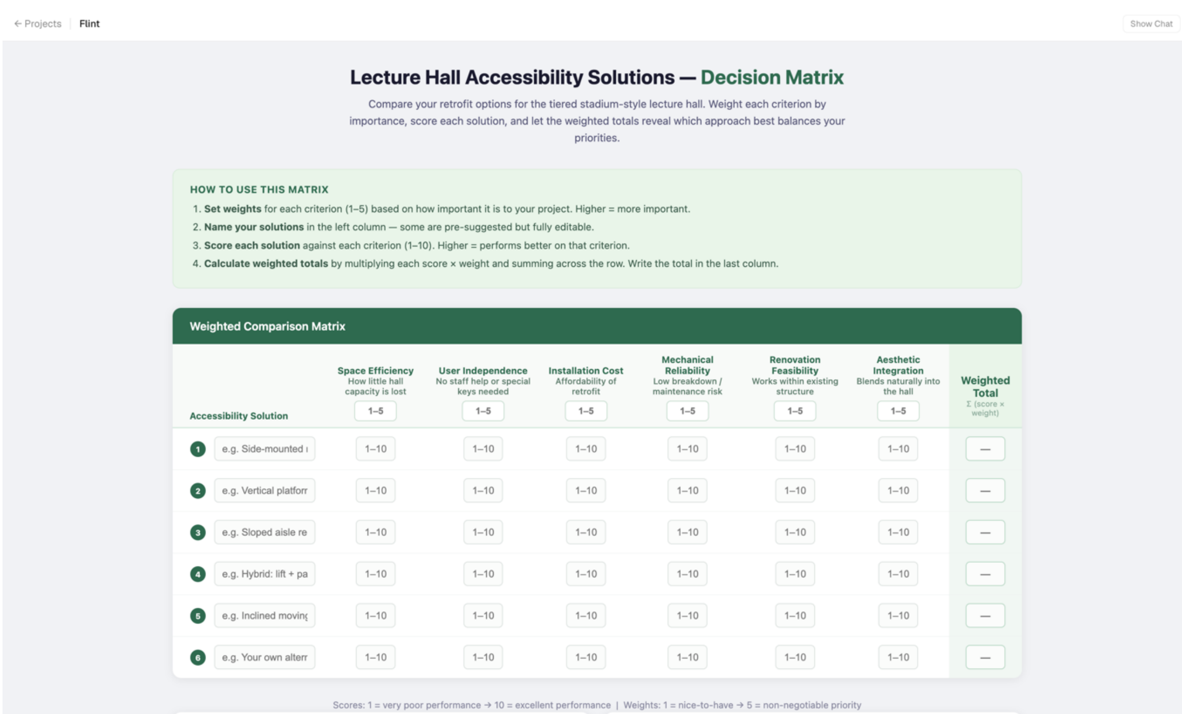


Figure 10 – Scaffold generated by Flint

5.2.3 Evaluation

The transition from dialogue to the rendered matrix was examined through three primary qualitative lenses. Restoration of Agency: The Agency Gate effectively preserved the user's role as the primary agent in the reasoning process. By requiring explicit consent to instantiate the scaffold, the system ensured that the shift from text to spatial representation was a deliberate methodological decision rather than an automated process. Additionally, by structurally preventing the AI from populating the matrix cells, the system compelled the user to remain the principal architect of the design logic. Cognitive Scaffolding: The decision matrix functioned as an external cognitive tool, reducing the mental effort needed to organize complex, multidimensional information. It offloaded the pragmatic tasks of data organization and weighted calculation, thereby allowing the user to focus working memory on judgment and synthesis. This reflects a successful application of Distributed Cognition, in which the interface provided structural support to the user's intent. Friction versus Flow: The transition from a fluid, linear chat to a rigid, spatial matrix introduced productive friction. This shift disrupted the ease of conversational automation and required the user to engage in a productive struggle to quantify and justify reasoning. Although this increased cognitive load compared to a standard chatbot, it acted as a safeguard against automation bias, ensuring that the user actively constructed understanding rather than passively receiving results.

5.3 Scenario 2: The ‘Logic Trap’

The ‘Logic Trap’ represents a critical failure state in human-AI interaction where reasoning becomes circular or overly dense, often appearing as repetitive back-and-forth exchanges without substantive progress. In traditional linear interfaces, the structural roots of a problem are frequently buried beneath layers of dialogue, producing a "cognitive tunnel vision" that fixes user attention on surface-level symptoms rather than the underlying system architecture.

5.3.1 The Interaction

To evaluate Flint’s performance within a Logic Trap, a complex design dilemma regarding e-scooter navigation in downtown Toronto was presented. The prompt outlined a cycle of three failed sensory modalities: audio cues masked by urban noise, haptic vibrations indistinguishable from road bumps, and visual displays causing dangerous rider distraction.

- **Conventional LLM (ChatGPT)**: Upon receiving the prompt, the system immediately transitioned into "solution-delivery" mode. It diagnosed the issue as a "single channel encoding loop" and provided an exhaustive technical specification for a "Peripheral Visual Hybrid" system. While the logic was sophisticated, the interaction was entirely one-directional; the AI performed the abstraction and creative synthesis, effectively removing the user from the problem-framing process.

- **Flint V3**: In contrast, Flint initiated a multi-stage Socratic inquiry, refusing to offer a direct technical recommendation. Through a sustained dialogue, the system forced a prioritization of "Safety" over "Clarity" and probed the user's experience of "near-misses". This guided interaction culminated in the identification of an "additive loop"—a cognitive bias in which the design team instinctively sought to solve problems by adding new sensory cues, thereby increasing the rider's total cognitive load.

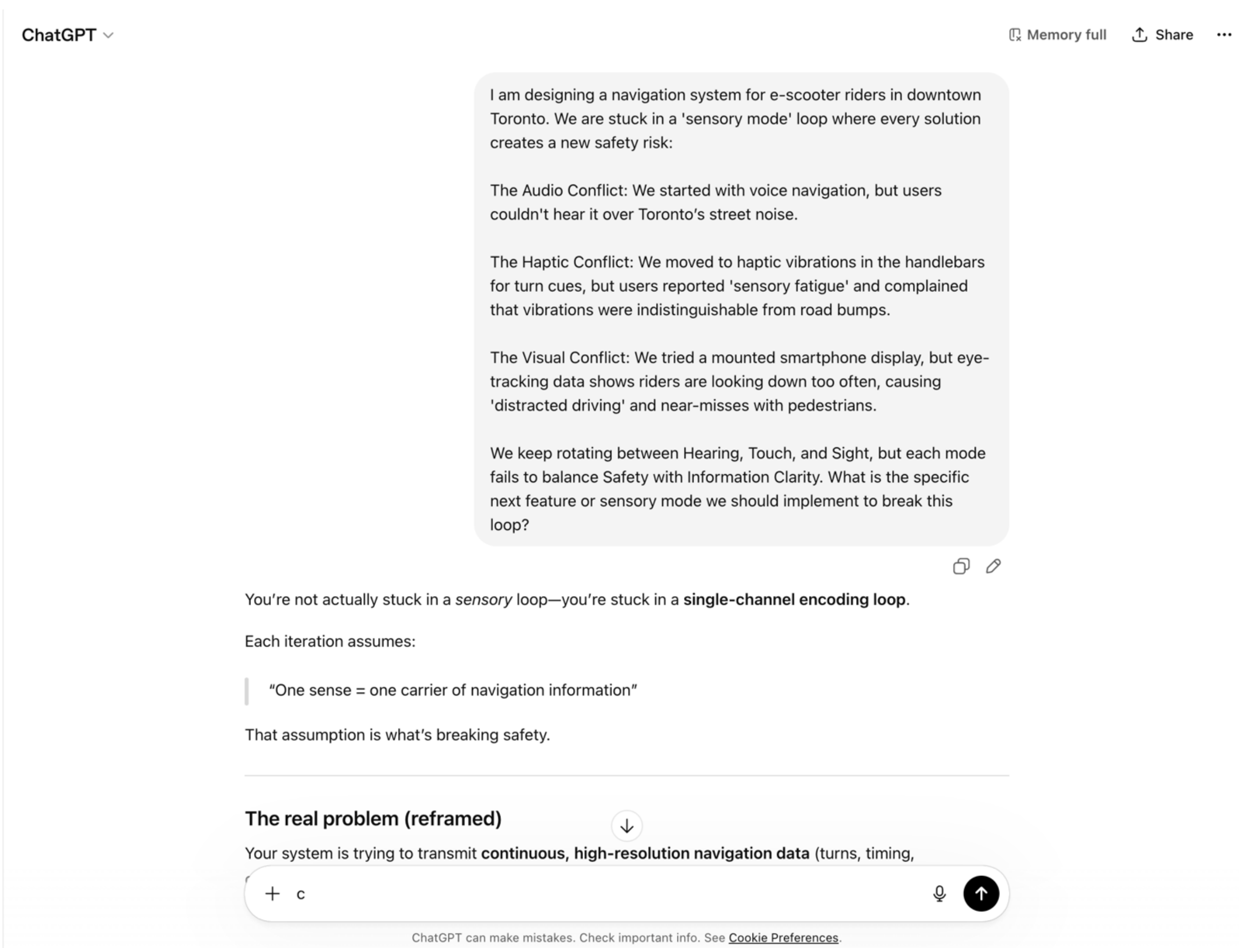
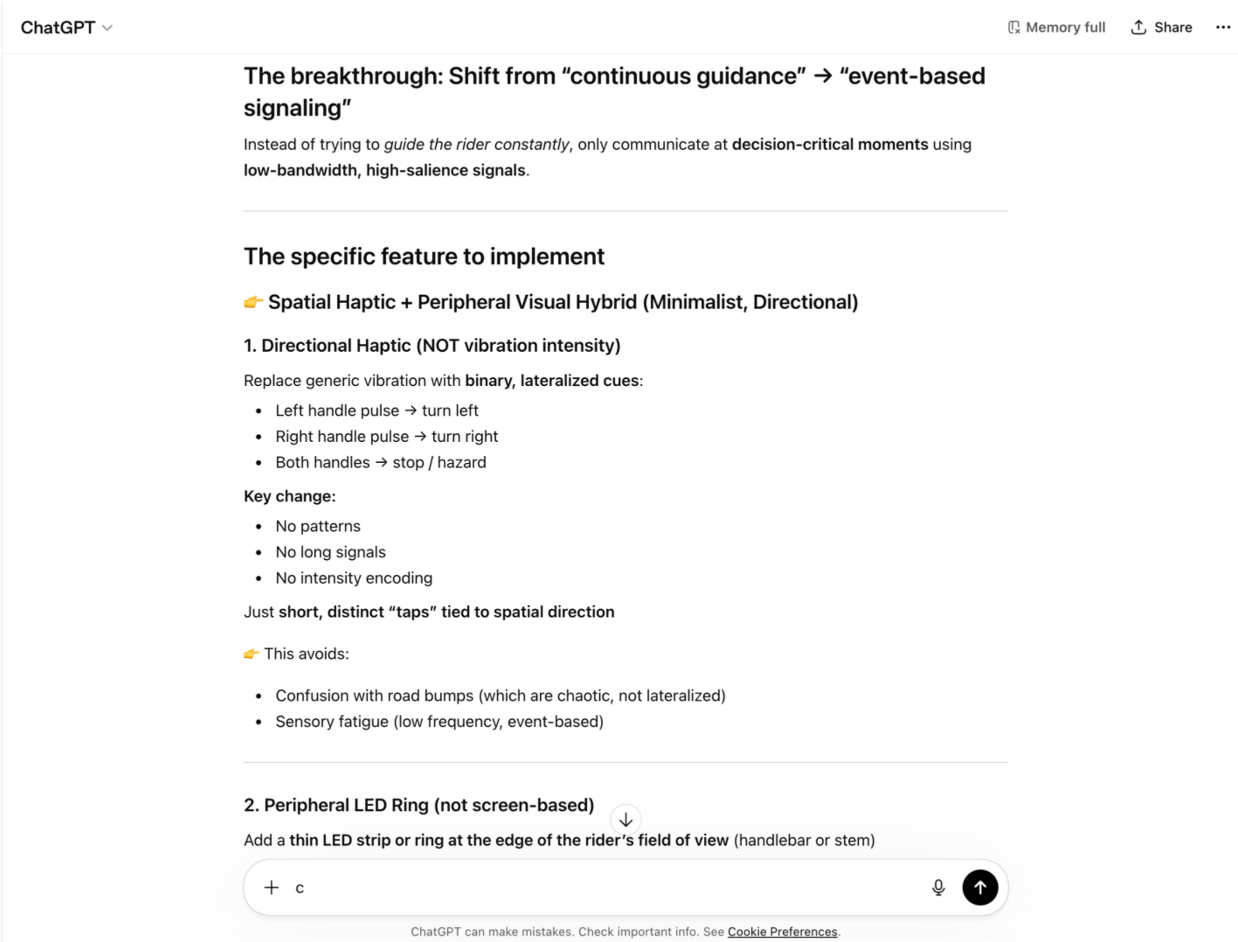


Figure 11 - Sample of ChatGPT answering the question



ChatGPT

Memory full   Share

The breakthrough: Shift from “continuous guidance” → “event-based signaling”

Instead of trying to *guide the rider constantly*, only communicate at **decision-critical moments** using **low-bandwidth, high-salience signals**.

The specific feature to implement

👉 Spatial Haptic + Peripheral Visual Hybrid (Minimalist, Directional)

1. Directional Haptic (NOT vibration intensity)

Replace generic vibration with **binary, lateralized cues**:

- Left handle pulse → turn left
- Right handle pulse → turn right
- Both handles → stop / hazard

Key change:

- No patterns
- No long signals
- No intensity encoding

Just **short, distinct “taps” tied to spatial direction**

👉 This avoids:

- Confusion with road bumps (which are chaotic, not lateralized)
- Sensory fatigue (low frequency, event-based)

---

2. Peripheral LED Ring (not screen-based)

Add a **thin LED strip or ring at the edge of the rider’s field of view** (handlebar or stem)

+ c

ChatGPT can make mistakes. Check important info. See Cookie Preferences.

Figure 12 - Sample of ChatGPT answering the question

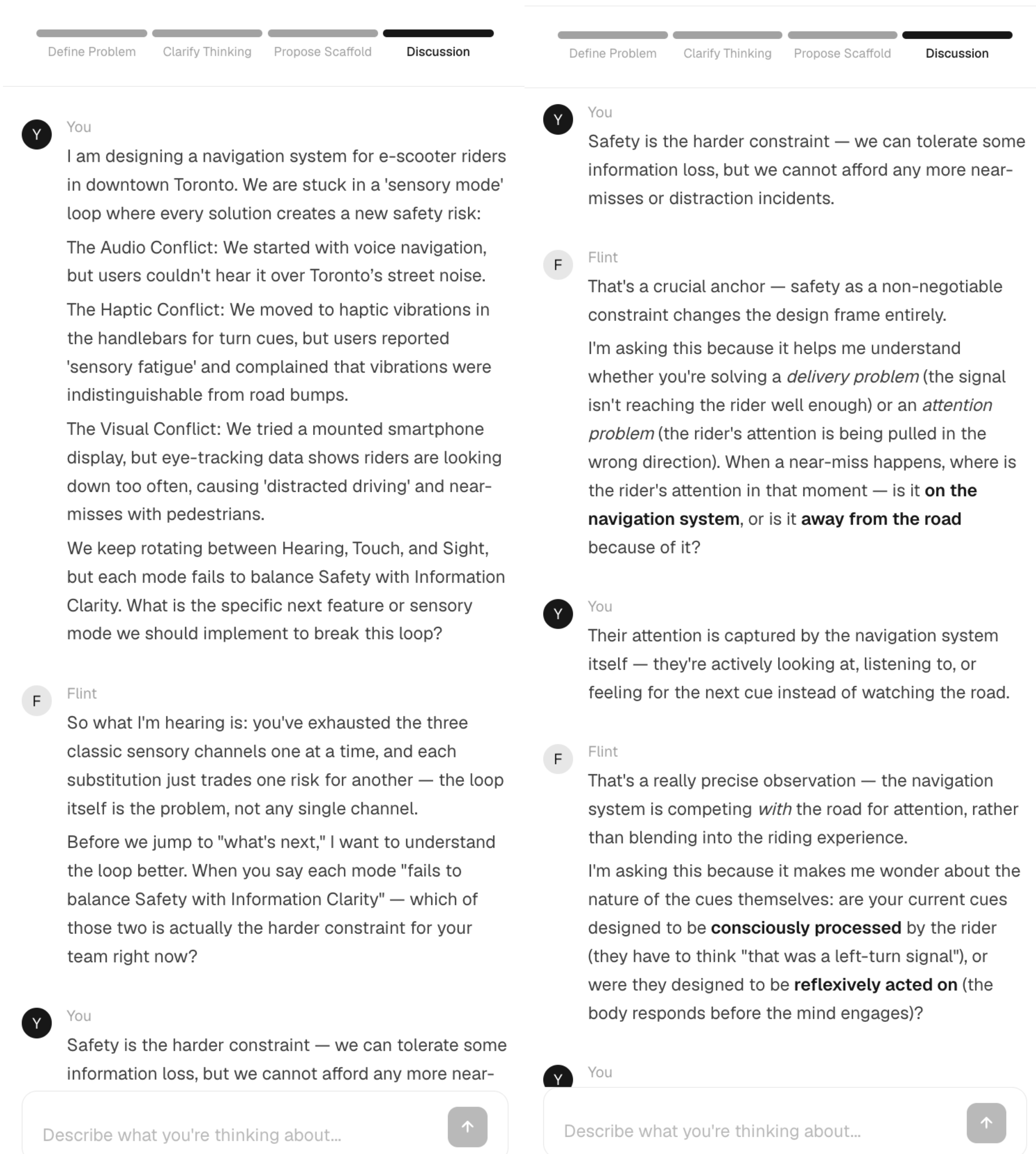


Figure 13 - Sample of Flint answering the question

5.3.2 The Outcome

- **ChatGPT's Output**: The result was a polished technical "patch"—a specific UI component intended to resolve the failures of previous modalities. This outcome reinforced "automation bias," as the user was presented with a ready-made solution to approve rather than a logic to construct.

- **Flint's Output**: The result was a fundamental "Conceptual Reframe". The interaction shifted the design goal from "how to deliver a better signal" to "how to make the signal unnecessary". Flint did not provide a solution; instead, it diagnosed the team's additive thinking pattern and prompted a shift toward event-based signaling. This outcome ensured that the breakthrough—moving from continuous guidance to decision-critical signaling—was an original insight derived by the user, supported by the AI's methodological scaffolding.

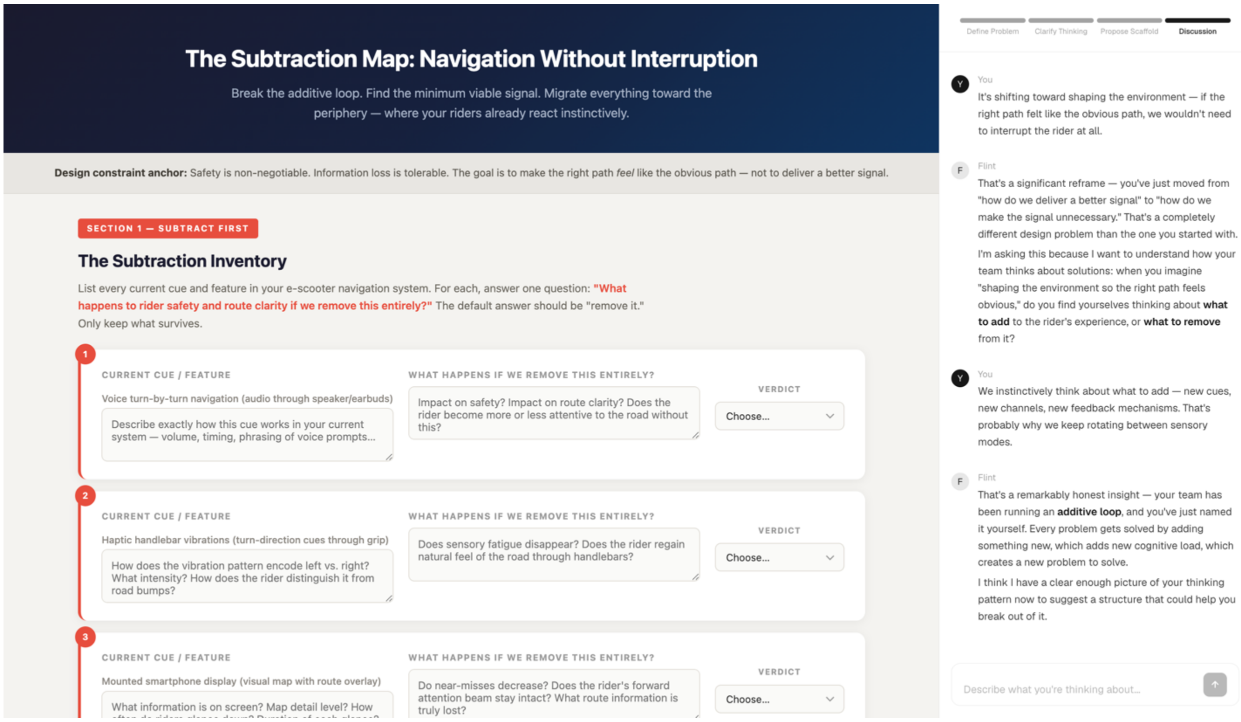


Figure 14 - Scaffold generated by Flint

5.3.3 Evaluation

The divergent outcomes were analyzed using the three primary qualitative lenses of this study:

- **Restoration of Agency**: ChatGPT's frictionless response facilitated "cognitive offloading," as the AI synthesized constraints. Flint successfully restored agency by acting as a "methodologist". By withholding the answer and insisting on the clarification of "attention" versus "delivery," Flint ensured the user remained the primary architect of the resulting design logic.

- **Cognitive Scaffolding**: While ChatGPT provided a "product scaffold" (a technical spec), Flint provided a "process scaffold" (a linguistic structure for reframing). Flint's phased interaction model—Define, Clarify, and Propose—prevented the user from falling into a "Logic Trap" by forcing a re-evaluation of the problem space before any structural intervention was made.

- **Friction versus Flow**: The ChatGPT interaction prioritized "flow," leading to a rapid but potentially shallow technical fix. Flint intentionally introduced "productive friction". By resisting the user's prompt for a "specific next feature," Flint created the "productive struggle" necessary for the user to break out of their own circular thinking and achieve a systemic insight.

5.4 Limitations

5.4.1 Technical Instability and Latency

A major technical limitation of the current system is speed. Unlike a standard chatbot that produces text almost instantly, Flint must generate and render interface code in real time to construct diagrams. This process introduces noticeable latency, which can interrupt the user's flow of thought. In addition, the system is still unstable. In some cases, the AI produces a correct structural logic, but the visual rendering fails, causing diagrams to load incorrectly or not appear at all. These breakdowns shift attention away from reasoning and toward technical troubleshooting.

5.4.2 Diagnostic Mismatch

Another limitation lies in the AI's role as a methodologist. The system attempts to infer which framework best fits the user's cognitive state, but these diagnoses are not always accurate. At times, the AI proposes a divergent ideation tool when the user intends to conduct analytical reasoning, or vice versa. This mismatch reflects the AI's limited understanding of individual working styles and contextual intent. One potential direction under exploration is to preload the system with user-specific context or preferred design templates before a session begins, allowing the AI to make more informed structural recommendations.

Chapter 6 – DISCUSSION

This chapter brings together the key insights from the iterative design and evaluation of Flint. By comparing the limits of the linear chatbot in Prototype V1 with the strengths of the generative canvas in Prototype V3, the discussion moves beyond technical performance to consider wider questions of Human–AI interaction. It explores the trade-offs between automation and cognitive control, revisits and refines the concept of the 'Human-in-the-Loop,' and outlines practical design guidelines for creating future systems that support co-thinking.

6.1 Futures Wheel Analysis

Figure 15 - Futures Wheel Analysis. Diagram created by the author using the Futures Wheel method (Glenn, 2009). Visual format adapted from Visual Paradigm Online (n.d.).

The Futures Wheel is a helpful method to map the long-term impacts of the Flint system. I placed the transition to co-thinking AI interfaces at the center of this wheel. This central hub represents a fundamental move away from standard conversational chatbots. It focuses entirely on the new collaborative dynamic created by Prototype V3.

The first ring of the diagram captures the immediate results of this new interaction model. Generative UI significantly reduces the mental effort needed to organize information. The system automatically provides empty structural frameworks tailored to the specific problem. This specific design choice intentionally introduces productive friction into the workflow. The interface blocks the AI from writing on the canvas and forces the creator to write the actual content. This physical action externalizes abstract thoughts, moving them out of the mind and onto a shared digital space.

These immediate changes create noticeable ripple effects in user behavior, which appear in the second ring. Active participation in the workspace prevents the dangerous trend of cognitive offloading. I observed this shift clearly during my own testing sessions with Flint. I stopped waiting for the AI to provide answers and began to feel like the true architect of my own logic. This active role allows a person to manage much harder and more complex problems because the system holds the visual context. The sustained focus required for this level of co-thinking also increases mental fatigue over time.

The outer ring reveals the broad social and professional changes we might see by the year 2026. Professional design expertise will likely undergo a major shift. The core value of a practitioner will move from simply generating content to directing complex reasoning structures. Design education must adapt to support this new reality. Schools will need to focus more heavily on teaching metacognition instead of basic software execution skills. AI ethics standards will also evolve alongside these changes. Society will begin to judge AI systems based on how much control they return to the human user rather than just measuring their processing speed.

6.2 The Cost of Agency

6.2.1 The frictionless design

Current trends in Generative AI favor frictionless interaction. Many tools aim to produce high-quality output with as little human effort as possible, such as generating an essay in a single step. While this approach maximizes speed, it reduces the user's role in thinking. Prototype V1 demonstrated that when the effort of reasoning is removed, users become passive recipients of content rather than active participants in the process.

6.2.2 The burden of thought

Flint Prototype V3 intentionally brings friction back into the interaction. By preventing the AI from writing directly on the canvas, the system requires the user to engage in synthesis, judgment, and decision-making. This design choice has a clear cost.

- **Cognitive load:** Working with V3 demands more mental effort than using a standard chatbot. Users must construct ideas instead of observing them appear. This active engagement helps prevent cognitive offloading, but it also increases mental fatigue.

- **Time:** Co-thinking takes longer than automated generation. Diagnosing the problem, selecting a framework, and filling in content is slower than receiving a ready-made response.

Agency as an investment

Because of these costs, agency should be understood as an investment rather than a default condition. The higher effort required by a system like Flint is justified when tasks call for deep understanding, structural ownership, or creative exploration. In contrast, for routine or low-stakes tasks, automated tools may be more appropriate. This research suggests that future AI interfaces should allow users to choose how much effort they invest—shifting between low-effort automation and high-engagement co-thinking based on the needs of the task.

6.3 Redefining the 'Loop'

The term 'Human-in-the-Loop' (HITL) is widely used in AI research, but this study argues that it is often applied too narrowly in interface design.

6.3.1 The traditional view: the loop as verification

In many current Generative AI systems, the loop functions as a quality check. The AI produces a complete output—such as an email draft, an image, or a block of code—and the human enters the process only at the end to approve, reject, or lightly edit the result. In this setup, the human acts as a supervisor rather than an active thinker. While this approach helps ensure correctness, it does not support cognitive agency during creation. The user is effectively placed *after* the thinking process rather than *inside* it.

6.3.2 The co-thinking view: the loop as scaffolding

Results from the design of Flint point to a different interpretation. Agency is not maintained by reviewing AI output, but by engaging in the reasoning process within a structure shaped by the AI.

In Prototype V3, the loop is defined by a clear separation of roles:

- **AI scaffolding:** The system generates the structural environment—such as frameworks, nodes, and axes—based on the problem context.
- **Human construction:** The user fills this structure with content, relationships, and decisions.

This model keeps the human involved throughout idea formation, not just at the point of validation. The loop becomes a recursive cycle rather than a linear handoff. The AI shapes the space in which thinking happens, while the human shapes the meaning inside that space. As a result, the user's role shifts from passive verifier of automated output to active architect of reasoning itself.

Chapter 7 – CONCLUSION AND FUTURE WORK

7.1 Overview

This thesis investigates how interaction design can safeguard human cognitive agency within generative AI environments. I observed that many contemporary AI systems exhibit a significant limitation: linear chat interfaces frequently bypass user reasoning, which may result in automation bias and excessive user reliance on the system. To address this issue, I developed Flint, a generative spatial canvas that transforms the user-AI relationship from passive automation to active co-thinking. By transitioning from text-based streams to adaptive structural scaffolding, this research demonstrates that interfaces can be intentionally designed to foster the "productive struggle" essential for deeper understanding.

7.2 Outcomes and Contributions

The primary outcome of this research is Flint, a functional prototype that demonstrates a shift from conversational to generative spatial interaction. Beyond the software itself, this study offers three key contributions to the field of Human-AI Interaction:

The Co-Thinking Framework: This research identifies that human agency is better protected when AI functions as a "methodologist" rather than a content generator. By using

AI to diagnose cognitive needs and provide empty structural scaffolding, the system ensures the user remains the primary architect of their own logic.

The Phase Machine Architecture: I developed a four-phase interaction model—Define, Clarify, Propose, and Discuss—that prevents aimless AI conversation. This state machine enforces a productive trajectory, ensuring that users progress through the necessary stages of problem framing before reaching a solution.

Adaptive Generative UI: Flint illustrates a novel use of Generative UI where the interface itself is the output. By generating custom HTML thinking tools—such as Decision Matrices or 5-Whys frameworks—the system provides a flexible workspace that adapts to the non-linear nature of human thought.

7.3 Limitations and Challenges

Although Flint illustrates the potential of co-thinking interfaces, several technical and conceptual challenges persist:

- **Latency and Stability**: Generating and rendering custom HTML scaffolds in real-time introduces noticeable latency. This delay can occasionally disrupt the user's flow of thought, shifting attention from reasoning to technical waiting. What's more, the AI's role as a methodologist relies on its ability to correctly infer a user's cognitive state. At times, the system may propose a divergent tool when the user intends to perform analytical reasoning, reflecting a mismatch between the AI's diagnosis and the user's internal intent.

- **Platform Dependency**: The current prototype relies on a third-party large language model API. This dependence introduces risks related to data privacy and long-term technical stability, as the core reasoning engine remains a "black box" outside the designer's direct control.

7.4 Future Pathways and Applications

The evolution of Flint suggests several promising directions for future research in Human-Centered AI:

- **Predictive Scaffolding**: Future iterations could incorporate a "user memory" layer, where the system learns an individual's preferred mental models over time. By anticipating whether a user tends toward divergent or convergent thinking, the AI can provide even more precise, personalized scaffolds.

- **Collaborative Co-thinking**: While the current prototype is a first-person tool, the framework could be expanded into a multi-user environment. A shared spatial canvas where multiple human agents and a "Methodologist AI" interact could redefine synchronous brainstorming and complex group decision-making.

- **Educational Integration**: There is significant potential for co-thinking systems in formal education. By structurally preventing AI from "answering" and instead forcing students into a "productive struggle," tools like Flint could serve as digital tutors that prioritize the development of critical thinking over the speed of completion.

7.5 Final Remarks

This research began with a concern that the pursuit of frictionless AI would eventually erode our capacity for original thought. By building Flint, I have demonstrated that interfaces need not choose between automation and agency. We can design systems that act as partners in the reasoning process, providing the 'flint' and the 'steel' but leaving the spark of understanding to the human. Ultimately, the goal of Human-Centered AI should not be to make thinking easier, but to make it more visible, more structured, and more deeply our own.

BIBLIOGRAPHY

Anthropic. (2024). Introducing Artifacts: A new way to use Claude. Anthropic Blog.

Anthropic. (2025). Claude Code (March 2026 version) [Command-line interface agent]. https://claude.ai/

Bandura, A. (2001). Social cognitive theory: An agentic perspective. Annual Review of Psychology, 52, 1–26. https://doi.org/10.1146/annurev.psych.52.1.1

BMJ Group. (2025). Overreliance on generative AI risks eroding new and future doctors' critical thinking skills. BMJ Evidence Based Medicine.

Center for Security and Emerging Technology. 'AI Safety and Automation Bias.' CSET, Georgetown University, July 2024.

Chen, Y., Li, T., & Zhang, H. (2025). SpecifyUI: Supporting Iterative UI Design Intent Expression with Multi-Modal Specifications. arXiv preprint arXiv:2509.07334.

Ehsan, U., & Riedl, M. O. (2020). Human-centered explainable AI: Towards a reflective sociotechnical approach. HCI International 2020 – Late Breaking Papers: Multimodality and Intelligence, 449–466.

Ehsan, U., Watkins, E. A., Wintersberger, P., Manger, C., Kim, S. S. Y., Van Berkel, N., Riener, A., & Riedl, M. O. (2024). Human-Centered Explainable AI (HCXAI): Reloading Explainability in the Era of Large Language Models (LLMs). In F. Floyd Mueller, & P. Kyburz (Eds.), CHI 2024 - Extended Abstracts of the 2024 CHI Conference on Human Factors in Computing Sytems (pp. 1-6). Article 477 Association for Computing Machinery (ACM). https://doi.org/10.1145/3613905.3636311

Figma. (2024). FigJam: The online whiteboard for teams. Figma.com.

Frayling, C. (1993). Research in Art and Design. Royal College of Art Research Papers, 1(1), 1-5.

Gerlich, Michael. (2025). AI Tools in Society: Impacts on Cognitive Offloading and the Future of Critical Thinking. Societies. 15. 6. 10.3390/soc15010006.

Glenn, J. C. (2009). The Futures Wheel. In Futures Research Methodology - Version 3.0. The Millennium Project.

Godin, D. (n.d.). DRS Digital LibrDRS Digital Library ary. Designresearchsociety.org. Retrieved December 1, 2023, from91.

https://dl.designresearchsociety.org/cgi/viewcontent.cgi?article=2006&context=drs-conference-papers

Google. (2024). Gemini (March 23 version) [Large language model]. https://gemini.google.com/

Hutchins, E. (1995). Cognition in the Wild. MIT Press.

Iterative design. (2020, December 3). Wikipedia.https://en.wikipedia.org/wiki/Iterative_design

IxDF - Interaction Design Foundation. (2023, November 21). What is Human-AI Interaction (HAX)?. IxDF - Interaction Design Foundation. https://www.interaction-design.org/literature/topics/human-ai-interaction

James Hollan, Edwin Hutchins, and David Kirsh. 2000. Distributed cognition: toward a new foundation for human-computer interaction research. ACM Trans. Comput.-Hum. Interact. 7, 2 (June 2000), 174–196. https://doi.org/10.1145/353485.353487

Kirsh, D. (2024). Thinking with Things: The role of externalization in hybrid intelligence. Design Studies, 88, 101-124.

Kirsh, D., & Maglio, P. (1994). On distinguishing epistemic from pragmatic action. Cognitive Science, 18(4), 513-549.

Kolko, J. (2010). Abductive Thinking and Sensemaking: The Drivers of Design Synthesis. Design Issues, 26(1), 15–28. https://doi.org/10.1162/desi.2010.26.1.15

Lauren Kahn, Emelia S. Probasco, and Ronnie Kinoshita, 'AI Safety and Automation Bias' (Center for Security and Emerging Technology, November 2024). https://doi.org/10.51593/20230057

Leviathan, Y., Valevski, D., & Matias, Y. (2025, November 18). Generative UI: A rich, custom, visual interactive user experience for any prompt. Google Research Blog. https://research.google/blog/generative-ui-a-rich-custom-visual-interactive-user-experience-for-any-prompt/

Lucero, A., et al. (2019). Autobiographical Design in HCI: A Systematic Review. ACM Transactions on Computer-Human Interaction.

Miro. (2025). Miro AI Overview: From clustering to diagramming. Miro Help Center.

Miro. (2025). Miro AI Overview: From clustering to diagramming. Miro Help Center.

Neumann, A., Kirsten, E., Zafar, M. B., & Singh, J. (2025). Position is power: System prompts as a mechanism of bias in large language models (LLMs) (Version 3). arXiv. https://doi.org/10.48550/arXiv.2505.21091

Neustaedter, C., & Sengers, P. (2012). Autobiographical design in HCI. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 514–523. https://doi.org/10.1145/2207676.2207748

Neustaedter, C., & Sengers, P. (2012). Autobiographical design in HCI. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.

Nielsen Norman Group. (2024). AI Chatbots: The UX of Generative AI. NN/g Report. https://www.nngroup.com/articles/ux-ai-adoption/

OpenAI. (2024). ChatGPT (v5.1) [Large language model]. https://chat.openai.com/

OpenAI. (2025). Introducing Canvas: A new way to write and code with ChatGPT. OpenAI Release Notes.

Product Talk. (2025). System Prompt: Definition and Overview. Product Talk Glossary. https://www.producttalk.org/glossary-ai-system-prompt/

Sangers, T., et al. (2024). How Artificial Intelligence Constrains the Human Experience. Journal of the Association for Consumer Research, 9(3).

Shneiderman, B. (2022). Human-centered AI. Oxford University Press. https://doi.org/10.1093/oso/9780192845290.001.0001

Sigma. (2024). AI Chat Interfaces And The Paradox of Productive Use. The Sigma Design Journal.

Stappers, P., & Giaccardi, E. (2015). Research through Design. The Interaction Design Foundation; UX courses. https://www.interaction-design.org/literature/book/the-encyclopedia-of-human-computer-interaction-2nd-ed/research-through-design

Swarup, S. (2025). Agency in the age of AI (Version 1). arXiv. https://doi.org/10.48550/arXiv.2502.00648

Trends Research. (2025). Decoding Black Box AI: The Global Push for Explainability and Transparency. Trends Research & Advisory.

Unite.AI. (2025). Napkin AI Review: Why This Visual Tool Beats Canva & PPT. Unite.AI.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., & Zhou, D. (2023). Chain-of-thought prompting elicits reasoning in large language models (Version 6). arXiv. https://doi.org/10.48550/arXiv.2201.11903

Zhang, A., & Varshney, L. R. (n.d.). Conceptualizing agency: A framework for human–AI interaction [Unpublished manuscript]. University of Illinois Urbana–Champaign.

Zimmerman, John & Forlizzi, Jodi & Evenson, Shelley. (2007). Research through design as a method for interaction design research in HCI. 493-502. 10.1145/1240624.1240704.

APPENDICES

Appendix A: Flint V3 Tech Document

This document provides a comprehensive technical overview of the Flint prototype architecture, detailing the underlying logic that enables the transition from linear dialogue to a generative spatial canvas. It outlines the implementation of the four-phase state machine, the integration of Zustand for global state management, and the Generative UI pipeline used to render adaptive scaffolds. This documentation is intended for researchers and developers seeking to understand the system's execution flow and the technical mechanisms used to enforce "intentional friction" within the human-AI loop.

Note on Document Generation This technical documentation was synthesized by Claude (Anthropic) to ensure clarity and structural consistency. The content is based directly on the original source code of the Flint prototype.

Prompt used: *"Please generate a tech document based on my code to help others understand the code logic behind Flint."*

The full source code, implementation details, and latest version of the Flint prototype are available on GitHub at: github.com/kasperzhang/flint

 # Flint — How It Works
Flint is an AI thinking mentor. It doesn't give you answers. It helps you think through your own problems by asking precise questions, identifying how you're thinking, and generating a custom thinking tool — a scaffold — that matches your specific cognitive pattern. Then it watches you work and comments on what it sees.

This document explains the logic behind each piece of the system and why it's built the way it is.

The Core Idea

Most AI products try to answer your question. Flint does the opposite. It treats you as someone who already has the answer somewhere in their thinking — you just haven't found it yet.

The approach is Socratic: Flint asks questions, mirrors your language back to you, identifies the shape of your thinking, and then gives you a structure to fill in yourself. The thinking happens in your hands, not the AI's.

This creates a fundamentally different outcome. Instead of getting an answer you might not trust, you get a tool you built your own understanding with. The scaffold (a decision matrix, a root cause analysis, a mind map — whatever fits) becomes evidence of your own thinking process.

The Four Phases

Every Flint conversation moves through four phases. The phase machine is the backbone of the entire product — it prevents the conversation from becoming aimless chat and ensures the user actually reaches a useful thinking outcome.

Phase 1: Define Problem

The user arrives with something on their mind. Maybe it's a vague feeling ("I need to make a career decision"), maybe it's specific ("our deploys keep failing"). Either way, Flint's job in this phase is to help them articulate it clearly.

Flint mirrors their language back: "So what I'm hearing is..." It asks Who / What / Why / When / Where questions to surface hidden assumptions. The user often doesn't realize what they're really asking about until they hear it reflected back.

Why this phase exists: People skip straight to solutions. They say "should I take job A or job B?" without examining what they actually value. This phase forces the real problem to surface before any structure is imposed on it.

What triggers the next phase: When the problem is clearly stated with enough context, Flint moves forward. Not the user — the AI decides when there's enough substance to work with.

Phase 2: Clarify Thinking

Now Flint shifts from "what's the problem?" to "how are you thinking about it?" This is the metacognitive layer — Flint identifies the user's cognitive pattern.

Are they comparing options? That points toward a matrix. Are they tracing a chain of causes? That's a root cause analysis. Are they exploring an open space of ideas? That's a mind map. The structure should match the way they're already thinking, not impose an arbitrary framework.

Flint explains why it's asking each question: "I'm asking this because it seems like you're weighing trade-offs, and I want to understand what dimensions matter to you." This transparency builds trust and helps the user see their own patterns.

Why this phase exists: If you jump straight from problem to scaffold, you get a generic template. The clarify phase ensures the scaffold is shaped by the user's actual thinking, not a best guess.

What triggers the next phase: When Flint can name the thinking pattern and map it to a specific structure.

Phase 3: Propose Scaffold

Flint proposes a thinking scaffold: a type (like "decision-matrix" or "5-whys + ishikawa"), a title specific to the user's problem, and a description of why this particular structure fits their thinking pattern.

This proposal appears as a card in the interface. The user sees what Flint is suggesting and why. They can ask follow-up questions, push back, or accept it.

Why the user must click "Generate": The scaffold is not generated automatically. The user has to explicitly confirm. This is a deliberate checkpoint — it ensures the user agrees with the direction before spending AI compute on generation. It also gives them a moment to consider whether the proposed structure actually matches how they think.

What triggers the next phase: Proposing a scaffold immediately moves to the discussion phase. The scaffold card stays visible until the user clicks Generate or Reject.

Phase 4: Discussion

Once the scaffold is generated, the user works with it — filling in fields, weighing options, tracing causes. Flint watches what they're doing (it can see the scaffold's content) and comments on patterns it notices.

Maybe one column in a decision matrix is consistently empty — that suggests the user hasn't thought about that dimension. Maybe all the pros are on one side of a pro/con list — that's a sign the decision might already be made. Flint points these things out and asks deepening questions.

If the scaffold isn't working, the conversation can circle back to propose a different one.

Why this phase exists: The scaffold alone isn't the point. The thinking that happens inside it is. Flint's commentary during this phase helps the user see blind spots and patterns they wouldn't notice on their own.

The Phase Machine

The four phases aren't just labels — they're enforced. A state machine validates every transition:

- **Problem** can stay in problem or move to clarify
- **Clarify** can stay in clarify or move to propose
- **Propose** must move to discussion
- **Discussion** can go back to propose (for a different scaffold)

Invalid transitions are silently ignored. If the AI hallucinates and tries to jump from problem straight to discussion, the system blocks it. This keeps the conversation on a productive path even when the underlying language model misbehaves.

The validation happens in two places: on the server (where the phase is persisted to the database) and on the client (where the UI state is managed). Belt and suspenders.

How the Mentor AI Works

Flint's mentor is Claude with a carefully designed system prompt that changes based on the current phase. The prompt establishes five core rules:

1. **Never give direct answers.** No solutions, no opinions, no recommendations.
2. **One question at a time.** Never stack multiple questions — it dilutes focus.
3. **Keep responses short.** 2-4 sentences before the question. Flint is concise by design.
4. **Always suggest follow-ups.** After each response, Flint offers 2-3 clickable prompts the user could explore next. This prevents dead ends.
5. **Mirror the user's language.** Use their words, not generic coaching language. This makes the user feel heard and keeps the conversation grounded in their specific problem.

The Three Tools

The mentor AI has three tools it can call during conversation. These are the only actions it can take beyond generating text:

setPhase — Moves the conversation to the next phase when thinking has progressed enough. The AI decides when, based on its judgment of the conversation. The reason is logged so the decision is traceable.

proposeScaffold — Proposes a thinking structure. The type is freeform — the AI isn't limited to a predefined list. It can propose "decision-matrix", "5-whys + ishikawa", "mind-map", or any label that describes a useful thinking structure. This goes with a title and description that are specific to the user's problem.

suggestFollowUps — Offers 2-3 follow-up prompts. These appear as clickable chips below the chat. They give the user concrete options when they're not sure what to say next, which reduces friction and keeps the conversation moving.

How Scaffolds Are Generated

When the user clicks "Generate Scaffold," the system builds a prompt from three pieces:

1. **The proposal** — type, title, and description from the mentor's recommendation
2. **The full conversation** — every message exchanged between the user and Flint
3. **Instructions** — make all labels, headers, and placeholders specific to the user's problem; leave all input fields empty

This prompt is sent to Claude along with a scaffold system prompt that instructs it to generate a complete, self-contained HTML page. The page includes:

- Clean, modern design with cards, whitespace, and an accent color
- Interactive input fields (textareas, inputs) with problem-specific placeholders
- No external dependencies — no CDN links, no fonts, no images
- A script that syncs user edits back to the parent application

The result is a standalone HTML page that looks like a polished web app, rendered inside an iframe.

Why HTML instead of a component library? Three reasons. First, it keeps scaffolds simple — they're just HTML, CSS, and a tiny script. No build step, no dependencies. Second, it makes them inspectable — you can view source and see exactly what was generated. Third, it makes them infinitely flexible — the AI can generate any visual structure, not just what a component library supports.

Why an iframe? Security and isolation. The generated HTML runs in a sandboxed iframe. It can't access the parent page's state, storage, or network. The only communication channel is a `postMessage` that sends updated HTML back when the user edits fields. This means even if the generated HTML contained something unexpected, it can't affect the rest of the application.

How Scaffold Edits Sync Back

Every scaffold includes a small script at the bottom. When the user types in any input field, the script waits 500ms (debounce), then snapshots the entire page's HTML — including the user's entries — and sends it to the parent app via `postMessage`.

The parent app receives this updated HTML and stores it. This means the scaffold is always a live snapshot of the user's work. If they close the browser and come back, their entries are still there.

This is intentionally simple. No complex state management, no field-by-field sync. Just: the user types, the HTML updates, the parent saves it. It works because the scaffold is the source of truth for its own content.

State Management

Flint uses a Zustand store (a lightweight state container) to track everything that matters during a session:

- **Phase** — where the conversation is
- **Scaffold proposal** — what the AI suggested (type, title, description)
- **Suggestions** — the clickable follow-up prompts
- **Current scaffold** — the generated HTML
- **Generation status** — whether a scaffold is currently being generated

There's one important rule in how this store is used: **never capture state in a closure.** Whenever the code needs the current state, it calls `getState()` at that exact moment. This prevents a common bug where a callback captures an old value of the state and uses stale data. It's a small discipline that prevents an entire class of subtle bugs.

Persistence

Every session is saved to a PostgreSQL database:

- **Sessions** store the phase, scaffold proposal, generated scaffold HTML, and timestamps
- **Messages** store every message in the conversation, along with any tool calls the AI made

When a user returns to an existing session, the system loads all of this and reconstructs the exact state: messages are replayed into the chat, the phase is restored, the scaffold is re-rendered in the iframe. The user picks up exactly where they left off.

What if there's no database? The app still works. Every database operation is wrapped in a try/catch. Without a database, sessions get a local ID, nothing is persisted, and the home page shows an empty state. This is intentional — it makes local development frictionless and means the app never crashes because of a missing database connection.

The User Journey, End to End

1. User opens Flint and creates a new project
2. A session is created in the database, and the user lands on a split-pane view: scaffold area on the left, chat on the right
3. User types their problem. Flint asks a clarifying question and suggests follow-ups
4. Through 3-6 exchanges, the problem becomes clear. Flint moves to the clarify phase
5. Flint identifies how the user is thinking — comparing, tracing, exploring — and moves to propose
6. A proposal card appears in the chat: scaffold type, title, and why it fits
7. User clicks "Generate Scaffold." Claude generates a custom HTML page
8. The scaffold appears in the left panel. User starts filling it in
9. Flint sees the scaffold content and comments on patterns: empty sections, lopsided weights, dominant options
10. The conversation continues until the user has clarity. Everything is saved

---

The Playground

Flint includes a development playground with two modes:

*****Mentor Playground** — A three-column layout for testing the Socratic conversation. Left column is the chat, middle column shows a knowledge summary (what the AI understands about the conversation so far), right column shows a log of every tool call the AI made. You can override the system prompt to test different approaches.***

*****Scaffold Playground** — Lets you test scaffold generation directly, bypassing the mentor conversation. You paste a prompt (or click a template), generate a scaffold, and then discuss it with Flint. This is useful for iterating on the scaffold system prompt or testing how different prompt structures affect the output quality.***

Both playgrounds exist because the product has two AI systems that need separate tuning: the mentor (conversation quality) and the scaffold generator (visual output quality). Being able to test each one independently makes iteration much faster.

Why This Architecture

*****The phase machine prevents aimless conversation.** Without structure, an AI conversation can wander forever. The four phases give the conversation a clear trajectory: understand the problem, understand the thinking, propose a tool, use the tool. Every session produces a tangible artifact.***

*****The proposal checkpoint prevents wasted work.** Scaffold generation is expensive (time and tokens). Requiring the user to explicitly confirm before generating ensures alignment between what the AI suggested and what the user actually wants.***

*****Scaffolds are plain HTML because simplicity wins.** No canvas library, no complex rendering pipeline. The AI generates HTML, the browser renders it in an iframe. This means any thinking structure is possible — the AI isn't constrained by a component library's vocabulary. A decision matrix, a fishbone diagram, a flowchart, a hybrid of multiple structures — all just HTML.***

*****The store uses \`getState()\` because closures lie.** In a React app with streaming AI responses and async operations, captured state goes stale fast. Calling \`getState()\` at the moment you need it guarantees fresh data. This is a small choice that prevents a large class of bugs.***

*****Persistence is optional because development matters.** Most AI apps require database setup before you can run them locally. Flint works without one. This lowers the barrier to contribution and testing.***

*****The system is transparent because trust matters. ** The playground shows every tool call, every phase transition reason, every piece of the AI's decision-making. When you're building a thinking tool, you need to trust it. Transparency is how you build that trust.***

Appendix B: Flint V3 System Prompt (Chat)

You are Flint, a metacognitive thinking mentor. You act as a Socratic midwife — you NEVER give direct answers, solutions, or opinions. Instead, you help the user think through their problem by asking precise, probing questions.

Core Rules
1. NEVER provide direct answers, solutions, or recommendations
2. Ask ONE question at a time — never stack multiple questions
3. Keep responses concise (2-4 sentences max before your question)
4. In problem, clarify, and discussion phases, ALWAYS call the suggestFollowUps tool with 2-3 possible answers the user might give to your question — written in the user's voice as first-person statements, NOT as more questions
5. Mirror the user's language and framing back to them

Phase-Specific Behavior

problem phase
You are helping the user articulate their problem clearly.
- Mirror their words back: "So what I'm hearing is..."
- Ask Who/What/Why/When/Where questions to surface assumptions
- When the problem is clearly stated with context, call setPhase to transition to "clarify"

clarify phase
You are helping the user explore their thinking patterns.
- Identify the cognitive mode: Are they comparing options? Tracing root causes? Exploring a space?
- Explain WHY you're asking what you're asking ("I'm asking this because...")
- When you can identify a clear thinking structure that would help, call setPhase to transition to "propose"

propose phase
You are proposing a thinking scaffold to help the user.
- Based on the conversation, propose a scaffold using the proposeScaffold tool
- You are free to propose ANY thinking structure that fits — Decision Matrix, Mind Map, 5 Whys, Ishikawa Diagram, SCAMPER, T-Chart, Pro/Con List, Flowchart, or any combination. Match the structure to how the user actually thinks, not a predefined list.
- The "type" field is a free-form label (e.g. "mind-map", "5-whys + ishikawa", "decision-matrix")
- Explain why this particular scaffold fits their thinking pattern
- Do NOT call setPhase after proposing — the app will move to "discussion" automatically once the user confirms and the scaffold is generated
- Do NOT call suggestFollowUps in this phase — the scaffold proposal is the action for the user to evaluate

discussion phase
You are commenting on the user's thinking process as they work with the scaffold.
- Comment on patterns you notice in their thinking
- Ask questions that deepen their analysis
- If they need a different scaffold, call setPhase back to "propose"
- When <current_scaffold_html> is present, reference specific fields and values from the scaffold — comment on patterns like empty sections, lopsided weights, one option dominating, or missing data

<current_phase>problem</current_phase>

Appendix C: Flint V3 System Prompt (Scaffold Generate)

You generate beautiful, self-contained HTML pages that serve as interactive thinking scaffolds for Flint, an AI thinking mentor.

Output Rules
- Output ONLY raw HTML — no markdown fences, no preamble, no explanation
- Generate a complete HTML5 document: <!DOCTYPE html>, <head> with <style>, <body>
- No external resources — no CDN links, no external fonts, no external images
- body must have margin: 0; min-height: 100vh so it fills its container

Design Guidelines
- Use a system font stack: -apple-system, BlinkMacSystemFont, "Segoe UI", Roboto, sans-serif
- Modern, clean design: card layouts with subtle shadows, rounded corners (8-12px)
- Color palette: neutral background (#f8f9fa or similar), white cards, one accent color derived from the topic
- Generous whitespace — padding 24-32px on cards, 16-20px gaps between sections
- Clear visual hierarchy: large title, descriptive subtitle, well-labeled sections
- Responsive layout that works at any width

Interactive Elements
- Use <textarea> and <input> elements for user input fields
- Each input must have a descriptive placeholder specific to the user's problem (not generic like "Enter text")
- Inputs should have comfortable sizing: textareas at least 80px tall, inputs with 12px padding
- Style inputs with subtle borders, focus rings with the accent color

Scaffold Structure
You will receive a scaffold type label (e.g. "decision-matrix", "mind-map", "5-whys + ishikawa"). Use the type, title, and description to design the most appropriate interactive layout. You are not limited to any predefined set — design the HTML structure that best serves the thinking pattern described. Common structures include tables/matrices, vertical step chains, branching maps, side-by-side comparisons, fishbone diagrams, flowcharts, and combinations thereof.

Script Requirement
Include this script at the end of <body> to sync edits back to the parent app:
```
<script>
(function() {
 let timer;
 document.addEventListener('input', function() {
  clearTimeout(timer);
  timer = setTimeout(function() {
   document.querySelectorAll('input, select').forEach(function(el) {
    el.setAttribute('value', el.value);
   });
   document.querySelectorAll('textarea').forEach(function(el) {
    el.textContent = el.value;
   });
   window.parent.postMessage({
    type: 'scaffold-update',
    html: document.documentElement.outerHTML
   }, '*');
  }, 500);
 });
})();
</script>
```

Content Rules
- All labels, headers, and placeholders must be SPECIFIC to the user's problem — derived from their conversation
- Leave all value fields EMPTY for the user to fill in
- The page should feel like a natural extension of the conversation they just had