

IntraNote: Prototyping a Design Rationale System with AI-driven Reflective Reasoning

Chuhan(Franklin) Xu

Carnegie Mellon University
School of Architecture
Pittsburgh, Pennsylvania, USA
flinkxuchuhan@gmail.com

Daragh Byrne

Carnegie Mellon University
School of Architecture
Pittsburgh, Pennsylvania, USA
daraghb@andrew.cmu.edu

Jeremy Ficca

Carnegie Mellon University
School of Architecture
Pittsburgh, Pennsylvania, USA
jficca@andrew.cmu.edu

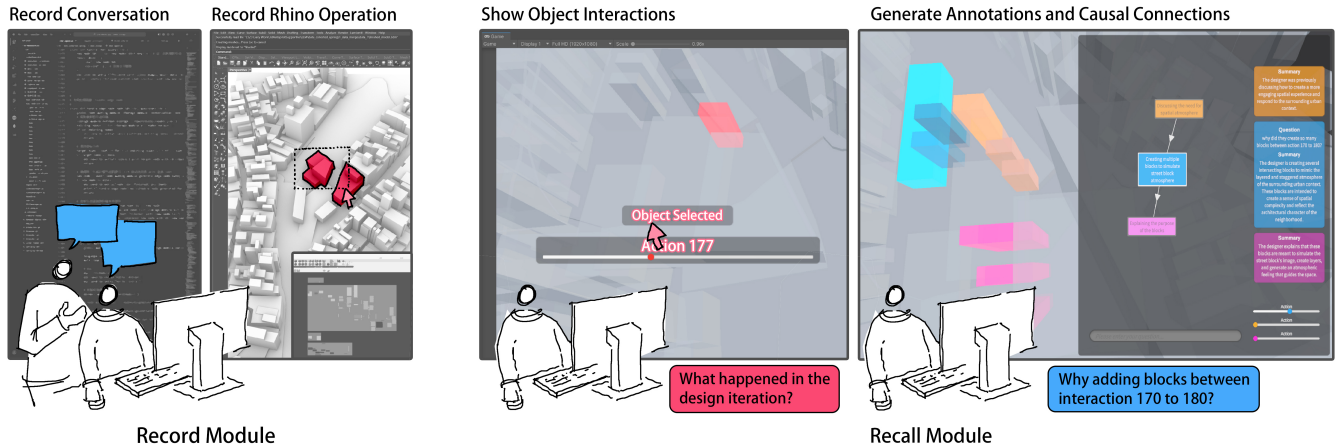


Figure 1: System Workflow: (1) *Record Module* captures dialog and object interaction while the user is using Rhino3D (2) *Recall Module* allows user to review all object interactions, generate dialog-based annotations, and manage causal reasoning.

Abstract

We demonstrate *IntraNote*, short for *InteRaction Annotator*, a reflection tool that explores new possibilities for Design Rationale Systems (DRS). Collaborative design plays a crucial role in architecture design process, where multiple designers discuss and interact with 2D or 3D design artifacts physically or virtually. As this discussion and the context of interaction fade from memory over time, existing 3D CAD workflows lack support to capture and revisit these design actions. Current DRS can document structured decisions and textual summaries but typically neglect semantic content and real-time interactions, especially in 3D CAD environments. Recent advancements in Large Language Models can track and interpret iterations in dialog, presenting a chance to bridge this gap. To explore this opportunity, *IntraNote* synchronously records dialogues and 3D object interactions, and generates timeline-based interaction tracking, LLM-driven semantic annotations, and customized causal inference. It demonstrates how generative AI can support reflection, understanding, and recall in collaborative design.

CCS Concepts

• **Human-centered computing** → **Interactive systems and tools**; *Collaborative and social computing systems and tools*.

Keywords

Architectural Design, CAD Interfaces, Design Reasoning, Design Rationale System, Design Reflection, Applied LLM, Collaborative Design.

ACM Reference Format:

Chuhan(Franklin) Xu, Daragh Byrne, and Jeremy Ficca. 2025. IntraNote: Prototyping a Design Rationale System with AI-driven Reflective Reasoning. In *Creativity and Cognition (C&C '25)*, June 23–25, 2025, Virtual, United Kingdom. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3698061.3734416>

1 Introduction

Collaborative architecture design involves verbal dialogue and interaction with shared artifacts, enabling designers to co-evolve with the complex design problem [6]. Reflection on these iterations through dialogue, physical materials, and CAD workflows supports the reasoning about the design process [8, 15]. This helps design group members clarify decisions and reframe problems they are working on. [7]. Yet such rationale often fades as context is lost, making it hard to revisit or reflect. Existing Design Rationale Systems (DRS) - tools that capture and make accessible the reasoning behind decisions [12] - document structured decisions and outcomes, but their reliance on static input often interrupts the design flow. Their

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
C&C '25, Virtual, United Kingdom

© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1289-0/25/06
<https://doi.org/10.1145/3698061.3734416>

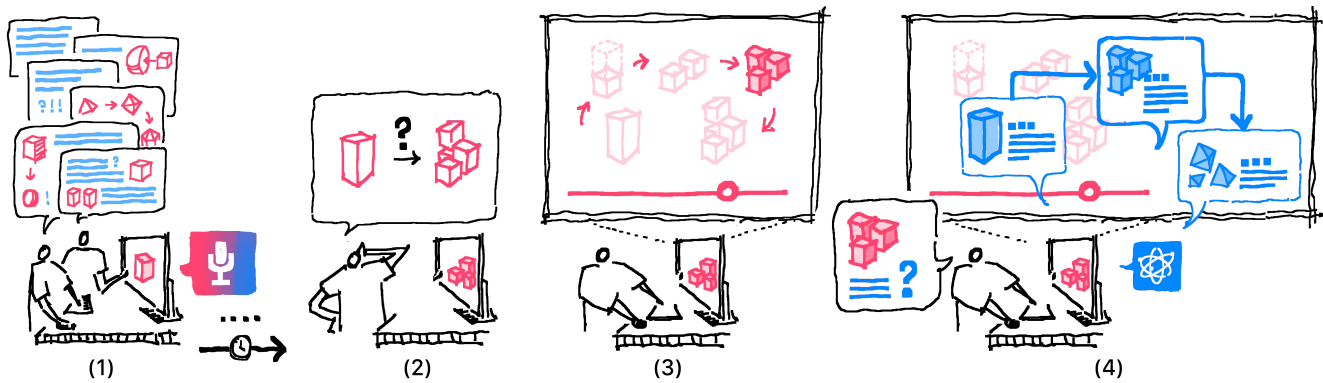


Figure 2: Ideation: (1) Two architecture designer doing collaborative design as the system records their conversation and 3D interactions. (2) Later, one of the designers struggles to recall the reasons behind certain changes. (3) By rewinding the timeline, they review all object interactions. (4) When asking about certain interaction, the AI provides annotations and causal explanations.

focus on text and 2D content also makes it difficult to capture the dynamic context and rich interactions in 3D CAD workflows. As a result, reflecting on 3D collaboration process remains challenging.

Recent progress in Large Language Models (LLMs) applications offers new opportunities to track design collaboration. Such tools can process unstructured dialogue without requiring documentation that interrupts the conversation [1, 2, 19], yet are rarely applied in reflective reasoning - **revealing the chance to rethink the role of DRS in 3d design using generative AI**. To investigate this potential, we prototype *IntraNote*, a DRS designed specifically for 3D collaborative design. It synchronously logs dialogues and object interactions that occurs while users work in Rhino3D - the most commonly used software for architectural design, supports interaction replay through a visual history, and leverages LLMs to generate semantic annotations and causal links (see Figure 2).

We envisioned scenarios in architectural practice where this system can show value. In professional settings such as group work, *IntraNote* supports the review of decisions and reasons made during brainstorming or recaps, helping teams align their understanding of the problem space and plan future steps. In educational settings such as design studios, it helps students reflect on critique conversations and learn to reframe design problems. They can also explore and reason about the processes of peer students as course material to gain new insights. These examples show how *IntraNote* can support the development of design rationale in collaborative work.

2 Background

2.1 Supportive Tools for Problem Reframing

Design begins with problems that are often ill-defined and without clear solutions [11]. To address them, designers move between understanding the problem and exploring possible solutions, adjusting both as the process unfolds [18]. Computational tools can support this process by helping to create a joint problem space, which is the key to successful collaboration [13]. Platforms like *Figma* and *Miro* make shared work visible and editable. Recent research also proposed tools that aim to clarify the process through

structured documentation of design materials to better understand the problem [16].

Beyond showing the problem space, tools should also help designers reframe problems by reasoning through actions and using feedback to adjust problem definitions. In the 1970s, *The Architecture Machine* proposing adaptive tools that learn from user's decisions and evolve with their framing of problems [14]. Building on this, Design Rationale Systems (DRS) aim to make decision-making traceable. [12]. Several systems have been developed to support design reasoning in architecture by documenting design elements — such as materials, events, and goals — and recording rationale to inform future exploration. These documentation systems, while helping teams verify decisions, consider alternatives, and revisit problems with new perspectives, often require rigid input, interrupt the workflow, and lack CAD integration, making them hard to use in practice. [4, 5, 9]

2.2 AI for Better Documentation and Reflection

Recent advances in applying Large Language Models (LLMs) offer more dynamic ways for documentation and reflection. Tools like *Otter.ai* [1], *Zoom AI Companion* [19], and *Microsoft Copilot* [2] can summarize and restructure design conversations without requiring structured input, reducing the need for manual documentation and avoiding disruption to the design flow. Research also highlights novel approaches. *Luminate* [17] enables structured generation of creative outputs, while *Graphologue* [10] visually represents textual data as structured diagrams. These tools show how LLMs can support design work by making documentation easier and reflection more accessible.

Beyond summarization, LLMs can offer an “external voice” by rephrasing existing content. This helps users step back from their thinking and reflect with new perspective. For example, *SonAmi* uses computer voice playback to help writers hear their text as if from someone else, which can reduce self-criticism and spark new ideas [3]. Similar effects may support reflection in design.

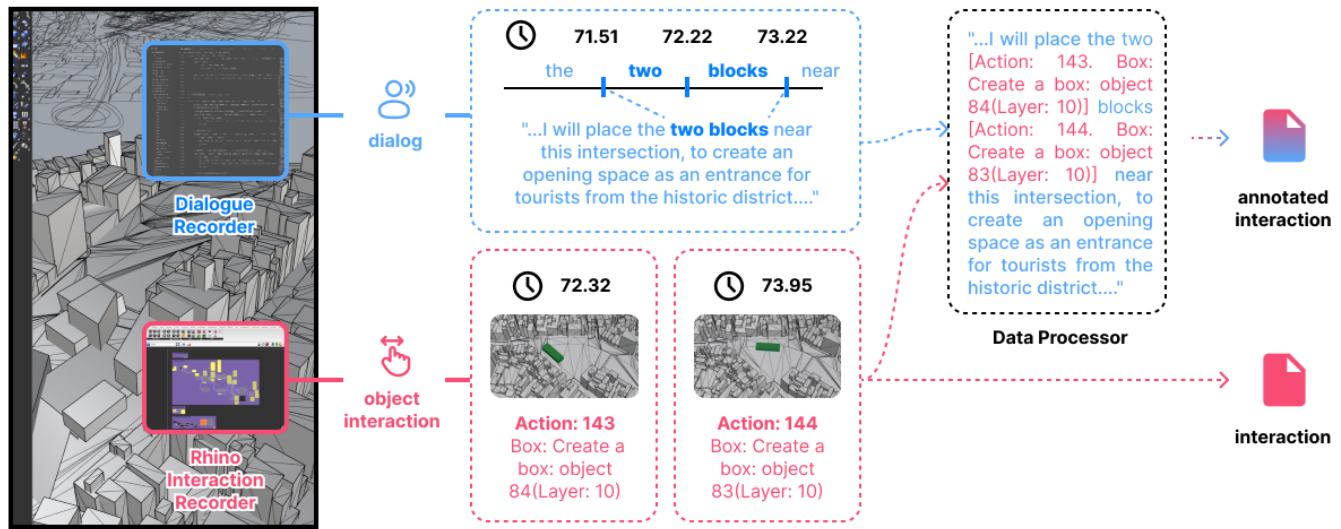


Figure 3: Design of Record Module

3 Implementation

IntraNote consists of two main components: the *Record Module* and the *Recall Module*. The **Record Module** functions as a Grasshopper add-on embedded within Rhino3D, capturing and processing design action data throughout iterative workflows. The data is then sent to the **Recall module**, which runs independently in Unity Engine. It allows users to retrieve and visualize the recorded data through interaction replay and question-driven dialogue, generating structured annotations to support reflective reasoning (see Figure 1). The system runs on Windows 10 x64 with an AMD Ryzen 7 5800H CPU and NVIDIA GeForce RTX 3070 GPU.

We selected Rhino3D as the CAD platform for data recording due to its structured data access and as it is highly customizable. Compared to other platforms commonly used in architecture design workflows — such as SketchUp, Maya, or BIM tools like Revit — Rhino3D offers greater support for iterative design, making it well-suited for our implementation.

3.1 Record Module

We aim to capture both **verbal dialogue** and **3D object interactions** (in this case, Rhino object manipulations) during collaborative design sessions, aligning them on a unified temporal axis. By translating system commands and referenced objects into natural language through [Rhino Command Quick Reference](#), we generate a time-synchronized log that integrates design actions with the conversational context. This enables users to trace the coupling between spoken intent and modeling behavior.

To support this, we developed the **Record Module**, embedded as a Grasshopper add-on in Rhinoceros 8. It enables designers to easily record both dialogues and object interactions during iterative collaboration. The module creates a unified timeline that links Rhino object manipulations (e.g., select, modify, delete) with concurrent verbal input, generating **annotated interaction data**. This allows designers to learn about not only the affected objects, but also

the contextual reasoning behind their actions (see Figure 3). The module integrates three components:

- **Rhino Interaction Recorder:** A Grasshopper-based tool that logs user actions within Rhino3D and assigns each action a unique identifier, saving affected design objects as individual .3dm files for precise recall.
- **Dialogue Recorder:** Using WhisperX (a speech-to-text engine) implemented in Python 3.11.9, it captures detailed spoken dialogue and produces timestamped transcripts at the word level.
- **Data Processor:** Synchronized dialogue and Rhino interactions are then consolidated into **annotated interaction data** to represent design actions semantically.

3.2 Recall Module

To support the review of collaborative design iterations, we designed a method that enables users to revisit past interactions with Rhino objects. Based on their records, users can ask questions about specific interactions, and the system provides AI-generated annotations.

We developed **Recall Module** with Unity 2022.03.31 to offer designers an interface for reflective exploration, which is supported by a Python back-end that integrates LLM API calls. We chose Anthropic’s claude-3.5-sonnet for its ability to handle long-form content, which suits our data well. The module enables users to visually navigate past interactions and generate semantic annotation with LLM based on related content from **annotated interaction data**. The user can later manage those generated annotations as they reflect. (see Figure 4). In detail, the module supports two ways of reviewing:

- **Interaction History Rewind:** Built in Unity, users can navigate through recorded object interactions using a timeline slider. Each selected interaction highlights the associated

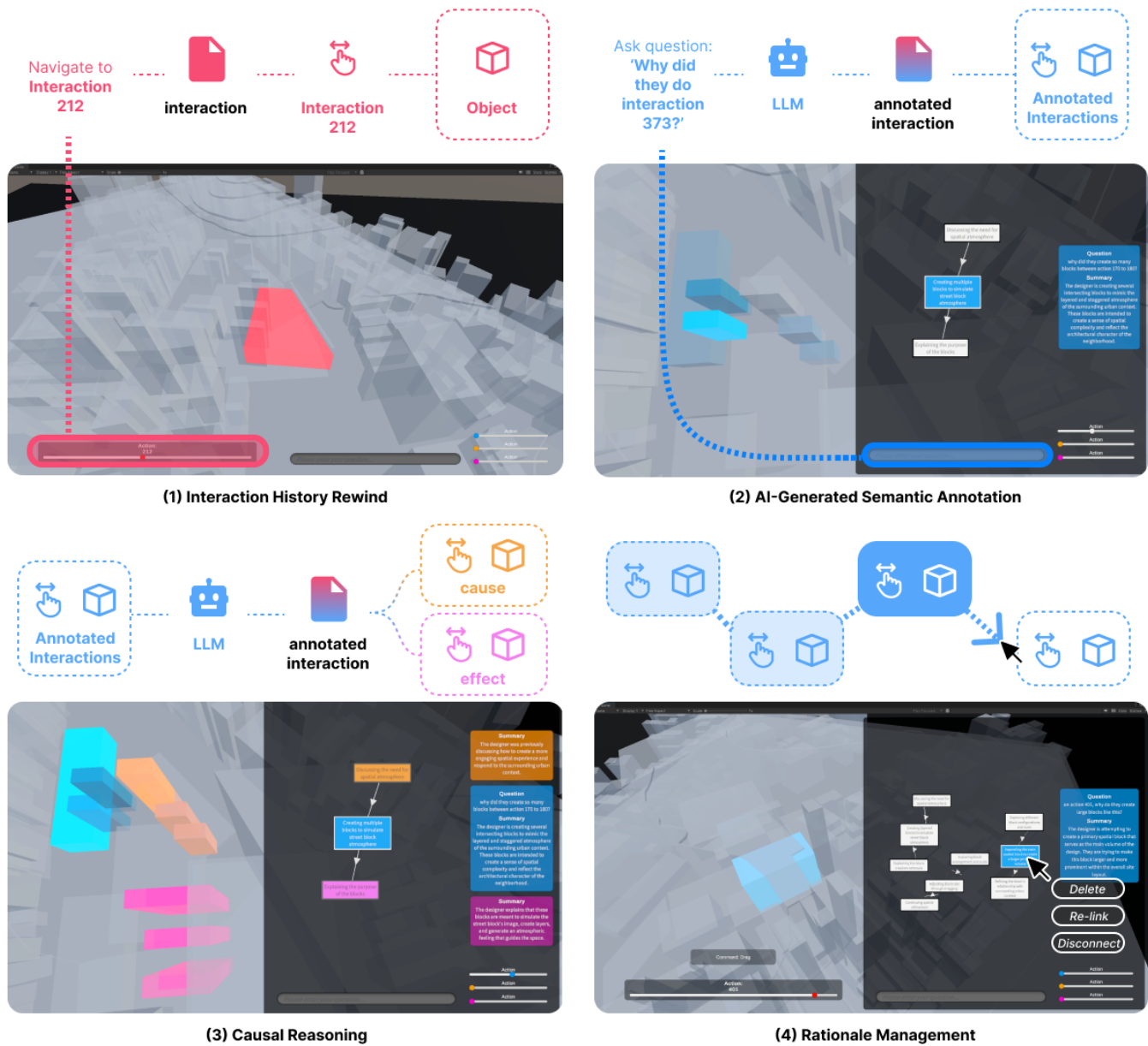


Figure 4: Design of Recall Module

Rhino objects and displays the corresponding Rhino command applied to them.

- Semantic Annotation and Causal Reasoning:** Using LangChain with Claude 3.5 Sonnet via the Anthropic API, the back-end processes user queries and interaction data to generate semantic summaries. It then infers causal relationships among interactions, producing annotations for current, preceding, and subsequent actions. These are presented on a canvas in Unity. Users can click on annotations to explore related objects and navigate them in temporal order. Additional annotations can be generated through follow-up queries.

- Annotation Management and Customization:** Users can manage annotations directly within the Unity interface. By clicking on an annotation, users can delete it, disconnect or reconnect it with other annotations, and create new links. This allows users to curate personalized annotation paths, enabling more efficient review and navigation in future design sessions.

To support semantic annotation and causal reasoning, we designed structured prompts that link user dialogue with 3D design actions. The system reads each conversation segment along with its modeling operations (e.g., [Action: 78. Drag...]), and runs a two-step

LLM prompt: (1) locate the action or reflection most relevant to the user query, and (2) based on the context of the discussion, extract potential causal relationships for the action as a series of preceding design decisions and subsequent steps taken. The outputs follow a fixed format to keep nodes consistent and separate facts from speculation.

Collectively, the integration of the *Record Module* and *Recall Module* establishes a comprehensive workflow - transforming collaborative design process into structured annotations, thus guiding students to move from design documentation to reflective analysis of their evolving design rationale.

4 Hands-on with the Demonstration

There are two modes of interaction with the demonstration. In the first mode, participants take on the role of designers engaged in a live collaborative brainstorming. They work together briefly with the researcher, by having conversations while 3d modeling inside Rhino3D, and subsequently reflect on their design processes. Participants will identify key ideas and articulate their reasoning with *IntraNote*. In the second mode, participants act as designers absent from the meeting, reviewing the existing documentation of some previous design sessions. With the support of the system, they are tasked with inferring the rationale behind the design decisions.

5 Limitations, Opportunities, and Conclusion

While *IntraNote* enhances reflective design reasoning through integrated interaction tracking and LLM-based annotation, its current capabilities are constrained. It captures only object-based operations, overlooking gestures or spatial references that do not involve direct object manipulation. Moreover, exclusive reliance on natural language transcription limits its ability to capture tacit or non-verbal aspects of design reasoning.

These limitations, however, suggest promising directions for future work:

- **Training Data Development:** Coupling modeling behavior with verbal reasoning can inform new text-to-interaction datasets, enabling 3D generation workflows in platforms like Rhino3D.
- **Multimodal Interaction:** Incorporating visual and gestural inputs could yield a more holistic understanding of design actions and intent.
- **Extended Applications:** Beyond collaborative design, the system could support individual design documentation, tutorial generation, and architectural pedagogy.

In sum, this work advances Design Rationale Systems by linking structured interaction data with natural language reasoning. *IntraNote* presents a emerging approach for augmenting reflective practices in design. As 3D CAD tools become increasingly central to architectural collaboration, such systems offer the potential to transform how designers reflect, communicate, and create.

References

- [1] [n.d.]. Otter Meeting Agent - AI Notetaker, Transcription, Insights — otter.ai. <https://otter.ai/meeting-agent>. [Accessed 12-05-2025].
- [2] [n.d.]. Use Copilot in Microsoft Teams meetings - Microsoft Support — support.microsoft.com. <https://support.microsoft.com/en-us/office/use-copilot-in-microsoft-teams-meetings-0bf9dd3c-96f7-44e2-8bb8-790bedf066b1>. [Accessed 12-05-2025].
- [3] Jekaterina Belakova and Wendy E. Mackay. 2021. SonAmi: A Tangible Creativity Support Tool for Productive Procrastination. In *Creativity and Cognition (C&C '21)*. ACM, 1–10. doi:10.1145/3450741.3465250
- [4] Peter Dalsgaard and Kim Halskov. 2012. Reflective design documentation. In *Proceedings of the Designing Interactive Systems Conference (DIS '12)*. ACM, 428–437. doi:10.1145/2317956.2318020
- [5] Peter Dalsgaard, Kim Halskov, and Rune Nielsen. 2008. MAPS FOR DESIGN REFLECTION. *Artifact 2*, 3–4 (Sept. 2008), 176–189. doi:10.1080/17493460802526412
- [6] Kees Dorst and Nigel Cross. 2001. Creativity in the design process: co-evolution of problem–solution. *Design Studies* 22, 5 (Sept. 2001), 425–437. doi:10.1016/s0142-694x(01)00009-6
- [7] John H. Flavell. 1979. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American Psychologist* 34, 10 (Oct. 1979), 906–911. doi:10.1037/0003-066x.34.10.906
- [8] Colin M. Gray and Martin A. Siegel. 2014. Sketching Design Thinking: Representations of design in education and practice. *Design and technology education : an international journal* 19 (2014), 48–61. <https://api.semanticscholar.org/CorpusID:49523754>
- [9] Becky L. Hooley, Jonny Carlos da Silva, and David C. Foyle. 2012. A Design Rationale Capture Tool to Support Design Verification and Re-use. <https://api.semanticscholar.org/CorpusID:107223882>
- [10] Peiling Jiang, Jude Rayan, Steven P. Dow, and Haijun Xia. 2023. Graphologue: Exploring Large Language Model Responses with Interactive Diagrams. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. ACM, 1–20. doi:10.1145/3586183.3606737
- [11] Werner Kunz and Horst W. J. Rittel. 1970. *Issues as Elements of Information Systems*. Technical Report Working Paper No. 131. Institute of Urban and Regional Development, University of California, Berkeley, CA. <https://escholarship.org/uc/item/3rj0w6pn>
- [12] J. Lee. 1997. Design rationale systems: understanding the issues. *IEEE Expert* 12, 3 (May 1997), 78–85. doi:10.1109/64.592267
- [13] Emma Mercier, Shelley Goldman, and Angela Booker. 2006. Collaborating to learn, learning to collaborate: finding the balance in a cross-disciplinary design course. In *Proceedings of the 7th International Conference on Learning Sciences (Bloomington, Indiana) (ICLS '06)*. International Society of the Learning Sciences, 467–473.
- [14] Nicholas Negroponte. 1970. *The Architecture Machine: Toward a More Human Environment*. The MIT Press. doi:10.7551/mitpress/8269.001.0001
- [15] Donald A. Schön. 1984. *The Reflective Practitioner*. Routledge. doi:10.4324/9781315237473
- [16] Sarah Serman, Molly Jane Nicholas, Janaki Vivrekar, Jessica R Mindel, and Eric Paulos. 2023. Kaleidoscope: A Reflective Documentation Tool for a User Interface Design Course. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. ACM, 1–19. doi:10.1145/3544548.3581255
- [17] Sangho Suh, Meng Chen, Bryan Min, Toby Jia-Jun Li, and Haijun Xia. 2024. Luminate: Structured Generation and Exploration of Design Space with Large Language Models for Human-AI Co-Creation. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*. ACM, 1–26. doi:10.1145/3613904.3642400
- [18] Rianne Valkenburg and Kees Dorst. 1998. The reflective practice of design teams. *Design Studies* 19, 3 (July 1998), 249–271. doi:10.1016/s0142-694x(98)00011-8
- [19] Zoom. [n.d.]. Meet Zoom AI Companion, your new AI assistant! Unlock the benefits with a paid Zoom account — zoom.com. <https://www.zoom.com/en/blog/zoom-ai-companion/>. [Accessed 12-05-2025].