Figmant — A Plugin for End-User Development of Wizard of Oz Experiments for Human-AI Interaction Design

Tommaso Turchi[1], Francesco Faenza[1], Alessio Malizia[1,3], Giacomo Bosio[2] and Nicoletta Bruno[2]

[1]Department of Computer Science, University of Pisa, Pisa, Italy
[2]Hedron, Livorno, Italy
[3]Molde University College, Molde, Norway

Abstract

This paper introduces Figmant, a Figma plugin that enables designers to conduct Wizard of Oz (WoZ) experiments for evaluating novel human-AI interactions without requiring actual AI implementation. Figmant extends the familiar Figma environment to allow designers to rapidly prototype, test, and iterate on AI-driven interfaces. The plugin provides a dual-interface system: one for participants interacting with the simulated AI, and another for the "wizard" to control the simulated AI responses. Built using react-figma, Figmant democratizes the prototyping of AI interactions, empowering designers with limited technical expertise to participate fully in the design of AI-driven user experiences. The plugin enables designers to simulate a wide range of AI behaviors through component state manipulation, dynamic content changes, and contextual frame navigation — all without requiring programming knowledge.

Keywords

end-user development, wizard of oz, prototyping, human-AI interaction, figma plugin, design tools

1. Introduction

As Artificial Intelligence (AI) becomes increasingly integrated into everyday digital products, designers face a growing challenge: they must design interactions for AI-driven systems without necessarily having the technical expertise to implement those systems. This creates a gap in the design process where designers rely on developers to implement their designs, often leading to a disconnect between design intent and implementation reality.

The Wizard of Oz (WoZ) methodology [1] offers a potential solution, allowing designers to simulate AI behavior during user testing without requiring actual AI implementation. However, existing WoZ platforms are typically custom-built for specific domains [2], requiring technical expertise that many designers lack.

This paper presents Figmant, a Figma plugin that enables designers to conduct WoZ experiments directly within their familiar design environment. By lowering the technical barriers to conducting WoZ tests, Figmant empowers designers to take a more active role in the development of AI-driven interfaces, aligning with the core principles of End-User Development (EUD) [3]. Unlike general-purpose WoZ platforms that require extensive configuration, Figmant integrates directly with Figma's design components and interactive features, allowing designers to simulate intelligent behaviors by manipulating prototypes in ways that participants perceive as AI-driven interactions.

The significance of this problem is underscored by the increasing prevalence of AI in digital products and the corresponding need for designers to rapidly explore, test, and iterate on AI-driven interactions. Without accessible tools, designers are often unable to validate their ideas or gather user feedback early, leading to missed opportunities for innovation and a disconnect between design intent and implementation. By addressing this gap, Figmant aims to empower a broader range of practitioners

Joint Proceedings of IS-EUD 2025: 10th International Symposium on End-User Development, 16-18 June 2025, Munich, Germany.
tommaso.turchi@unipi.it (T. Turchi); f.faenza1@studenti.unipi.it (F. Faenza); alessio.malizia@unipi.it (A. Malizia); giacomo@hedron.it (G. Bosio); nicoletta@hedron.it (N. Bruno)
0000-0001-6826-9688 (T. Turchi); 0000-0002-2601-7009 (A. Malizia)

to participate in the design of intelligent systems, ultimately improving the quality and inclusivity of AI-driven user experiences.

2. Related Work

Our research builds upon three main areas of prior work: Wizard of Oz methodologies, end-user development for AI, and design tool extensions.

2.1. Wizard of Oz for AI Interface Design

The Wizard of Oz technique has been widely used to prototype and evaluate novel interfaces before their technical implementation. Klemmer et al. [4] developed Suede, a prototyping tool for speech interfaces using the WoZ approach. More recently, Hu et al. [2] introduced Wizundry, a platform supporting multiple wizards collaborating to simulate more sophisticated speech-based interfaces.

These systems demonstrate the value of WoZ methodologies for exploring novel interaction paradigms, but they typically require specialized platforms separate from designers' usual workflows. While Wizundry provides powerful collaboration features for speech-based interfaces with multiple wizards, Figmant takes a different approach by integrating WoZ capabilities directly into Figma, focusing on visual interface manipulation within a tool already familiar to designers.

2.2. End-User Development for AI

End-User Development (EUD) provides a foundation for empowering non-programmers to shape and adapt digital systems to their needs. The meta-design framework, as articulated by Fischer et al. [5], emphasizes the creation of environments where users are not just passive consumers but active co-designers, able to extend and reconfigure systems. This approach is particularly relevant for AI, where the complexity and opacity of underlying algorithms often create barriers for designers and other stakeholders. By applying meta-design principles, tools can be developed that lower these barriers, enabling designers to experiment with and test possible AI behaviors without requiring deep technical expertise. This aligns with the broader EUD vision described by Lieberman et al. [3], which advocates for making systems modifiable by end users. Recent work by Dove et al. [6] further highlights the challenges faced by UX practitioners in prototyping and envisioning AI-driven experiences. Figmant builds on these insights by providing a meta-design environment within Figma, allowing designers to simulate, adapt, and iterate on AI interactions as part of their regular workflow, thus operationalizing EUD principles in the context of human-AI interaction design.

2.3. Design Tool Extensions

The extension of design tools to support new capabilities has been explored in various contexts. Myers and Stylos [7] addressed API usability, highlighting the importance of providing appropriate abstractions for different user groups. In the context of voice interfaces, Cambre and Kulkarni [8] discussed the challenges of designing for voice and the need for better prototyping tools.

Figmant builds on these insights by extending Figma — a widely used design tool — with capabilities specifically tailored to the needs of designers working on AI interfaces. By leveraging Figma's existing component system and adding a layer for WoZ simulation, Figmant provides a seamless extension to designers' existing workflows.

3. Figmant System Design

Figmant is designed as a Figma plugin that enables designers to conduct Wizard of Oz experiments for AI interfaces. The plugin extends Figma's capabilities to support the specific needs of simulating AI

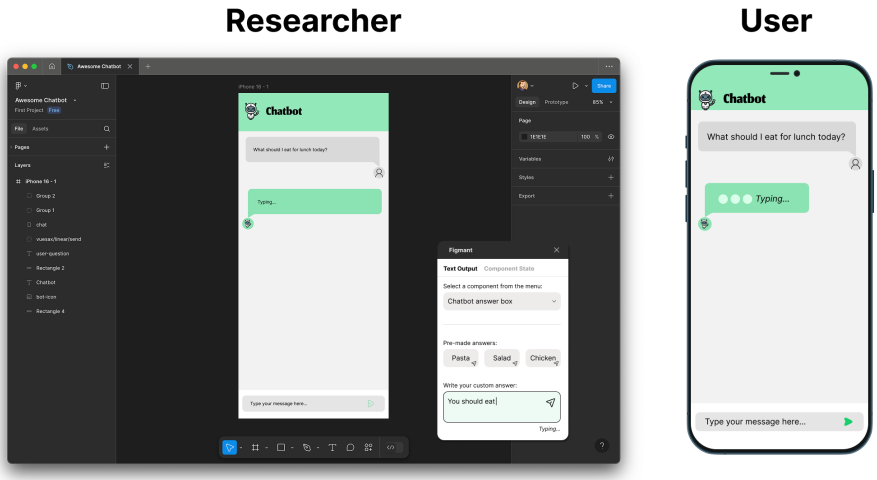

Figure 1: The Figmant architecture showing the Participant and Wizard interfaces

interactions while maintaining the familiar design environment that designers already use. Below we describe the key components and features of our system.

3.1. Plugin Architecture

Figmant is built using react-figma, allowing for seamless integration with the Figma environment. The plugin leverages Figma's API to access and manipulate design elements while maintaining a clear separation between design assets and the WoZ experiment configuration. This separation ensures that designers can continue to iterate on their designs without disrupting experiment setups.

The plugin architecture consists of three main interfaces:

1. **The Designer Interface**: Allows designers to configure the WoZ experiment, designate which elements can be manipulated by the wizard, and define the range of possible responses.
2. **The Participant Interface**: A clean view of the prototype that participants interact with, believing they are interacting with an AI system.
3. **The Wizard Interface**: A control panel that allows the human "wizard" to observe participant actions and trigger appropriate responses in real-time.

The plugin maintains a synchronized state between these interfaces, ensuring that actions in the wizard interface are immediately reflected in the participant interface.

3.2. Experiment Configuration

Setting up a WoZ experiment in Figmant involves defining interaction points within the Figma prototype and associating them with possible AI responses. Designers can specify:

- **Trigger Components**: UI elements that participants can interact with
- **Response Options**: A library of possible responses for each trigger
- **Response Timing**: Simulated processing time to maintain the illusion of AI processing
- **Recording Settings**: Options for capturing interaction data during the experiment

3.3. Wizard Interface Capabilities

To further illustrate the unique capabilities of Figmant, we provide a detailed walkthrough of the Wizard Interface functionalities:

- **Live Component State Control**: The wizard can instantly toggle between predefined states of any interactive component (e.g., switching a chatbot card from "waiting" to "responded"), enabling dynamic, context-sensitive UI updates.
- **Frame Navigation**: The wizard can navigate the participant's view to any frame in the prototype, simulating context-aware or multi-step AI workflows.
- **Dynamic Content Insertion**: The wizard can insert or modify text, images, or other content in real time, simulating AI-generated responses or suggestions.
- **Pattern Recognition Simulation**: The wizard can trigger specific responses based on observed participant actions, mimicking AI recognition of user intent or behavior patterns.
- **Simulated Processing Delays**: The wizard can introduce artificial delays before responses appear, maintaining the illusion of AI computation.
- **Error and Fallback States**: The wizard can trigger error messages or fallback options, allowing designers to test how users respond to system limitations.
- **Interaction Logging**: All participant actions and wizard interventions can be recorded for later analysis, supporting iterative design and evaluation.

This set of features allows designers to simulate a wide range of AI behaviors and user experiences, even in the absence of actual AI backends.

3.4. Design Principles

The development of Figmant was guided by user-centered design and meta-design principles. User-centered design [9] ensured that the needs, workflows, and pain points of designers were central throughout the process, while meta-design [5] informed the creation of an environment that empowers users to adapt and extend the tool to their specific contexts. By embedding these frameworks into the plugin's architecture, we aimed to maximize accessibility, flexibility, and the potential for end-user innovation.

4. Example Use Cases and Anticipated Benefits

To illustrate how Figmant enables designers to prototype and test AI-driven interfaces, we present several example use cases along with the anticipated benefits and challenges of our approach.

4.1. Example Applications

Conversational AI Interface A designer testing a chatbot interface that generates personalized recommendations can use Figmant to create different component states representing various response types, observe user inputs through the Wizard Interface, select appropriate responses (see Figure 1), trigger typing indicators with timing delays to simulate natural conversation flow, and navigate to specific frames representing different conversation stages. Additionally, designers can prepare multiple alternative dialog flows and test them with participants to explore different conversational strategies.

AI-Driven Content Curation For testing an interface that uses AI to curate content based on user interests, the wizard can observe user behavior, dynamically swap content components to show personalized recommendations, simulate learning over time by gradually refining content selection, and trigger different states showing varying confidence levels.

Context-Aware Assistant For an AI assistant that responds to contextual cues, the wizard can change suggested actions based on the user's task state, navigate between contextual frames based on inferred intent, manipulate component states to reflect different assistance levels, and simulate proactive suggestions by triggering notifications at appropriate moments.

4.2. Expected Benefits

Figmant's integration with Figma should significantly reduce technical barriers, allowing designers with no prior AI development experience to create and test interaction prototypes after minimal training. This tight integration should also increase iteration speed, as designers can quickly modify prototypes and immediately test changes with participants. Additionally, the low cost of creating and testing varied AI behaviors should lead to broader exploration of design possibilities.

4.3. Anticipated Challenges

Despite these benefits, several challenges remain. Maintaining perfect synchronization between wizard and participant interfaces in complex scenarios will be technically demanding. The system may face performance limitations with many possible AI responses or complex UI states. Perhaps most significantly, managing multiple interaction points simultaneously could place substantial cognitive demand on wizards, particularly when simulating sophisticated AI systems.

4.4. Workflow and Outcomes

Figmant automatically records all interactions between participants and the wizard during each Wizard of Oz experiment session. This includes user actions, wizard-triggered responses, and any changes made to the prototype in real time. The resulting interaction logs can be exported for further analysis, enabling designers and researchers to review user behavior, identify usability issues, and evaluate the effectiveness of simulated AI interactions. This data-driven approach supports iterative refinement of both the prototype and the experiment setup, helping teams make evidence-based design decisions and improve the overall user experience.

5. User Involvement and Evaluation

While a formal user study has not yet been conducted, we plan to evaluate Figmant through a series of user-centered studies involving professional designers and design students. These studies will focus on usability, effectiveness, and the impact of Figmant on the prototyping workflow compared to traditional methods. We will employ a combination of observation, interviews, and analysis of interaction logs to assess how easily users can configure and run WoZ experiments, as well as the quality of the resulting prototypes. Feedback from these studies will directly inform future development, with the goal of further lowering barriers to entry and enhancing collaborative and semi-automated wizarding features.

6. Conclusion

Figmant represents a step toward democratizing the design of AI interactions by leveraging the principles of End-User Development. By integrating WoZ methodology directly into designers' existing tools, we reduce the technical barriers to prototyping AI interactions while maintaining the flexibility needed to explore innovative design directions.

We believe this approach can significantly improve designers' ability to prototype and test AI-driven interfaces without requiring deep technical expertise. The ability to manipulate component states, navigate between frames, and dynamically update content should prove especially valuable for simulating AI behaviors that would otherwise require complex implementation.

The importance of this work to the EUD community lies in its focus on empowering designers — end-users of design tools but not typically developers — to create functional prototypes of AI systems. By bringing the power of WoZ methodology into familiar design environments, Figmant exemplifies how end-user development principles can be applied to emerging technology domains.

As we continue to develop Figmant, our future work will focus on several key areas: implementing semi-automated response capabilities with conditional logic to reduce wizard cognitive load; extending

the platform to support multiple wizards collaborating to simulate more complex AI systems, inspired by Hu et al.'s [2] work; enhancing manipulation capabilities with animation controls and multimodal input processing; and exploring transitions from wizard-simulated behaviors to actual AI implementations, creating a bridge between early WoZ testing and later-stage AI development.

Acknowledgments

This work was produced with the co-funding of the European Union – Next Generation EU, in the context of The National Recovery and Resilience Plan, Investment 1.5 Ecosystems of Innovation, Project Tuscany Health Ecosystem (THE), ECS00000017. Spoke 3.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT and Grammarly for grammar and spelling checks, paraphrasing, and rewording. After employing these services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] N. Dahlbäck, A. Jönsson, L. Ahrenberg, Wizard of oz studies — why and how, Know.-Based Syst. 6 (1993) 258–266. URL: https://doi.org/10.1016/0950-7051(93)90017-N. doi:10.1016/0950-7051(93)90017-N.

[2] S. Hu, H. C. Yen, Z. Yu, M. Zhao, K. Seaborn, C. Liu, Wizundry: A cooperative wizard of oz platform for simulating future speech-based interfaces with multiple wizards, Proc. ACM Hum.-Comput. Interact. 7 (2023). URL: https://doi.org/10.1145/3579591. doi:10.1145/3579591.

[3] H. Lieberman, F. Paternò, M. Klann, V. Wulf, End-User Development: An Emerging Paradigm, Springer Netherlands, Dordrecht, 2006, pp. 1–8. URL: https://doi.org/10.1007/1-4020-5386-X_1. doi:10.1007/1-4020-5386-X_1.

[4] S. R. Klemmer, A. K. Sinha, J. Chen, J. A. Landay, N. Aboobaker, A. Wang, Suede: a wizard of oz prototyping tool for speech user interfaces, in: Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology, UIST '00, Association for Computing Machinery, New York, NY, USA, 2000, p. 1–10. URL: https://doi.org/10.1145/354401.354406. doi:10.1145/354401.354406.

[5] G. Fischer, E. Giaccardi, Y. Ye, A. G. Sutcliffe, N. Mehandjiev, Meta-design: a manifesto for end-user development, Commun. ACM 47 (2004) 33–37. URL: https://doi.org/10.1145/1015864.1015884. doi:10.1145/1015864.1015884.

[6] G. Dove, K. Halskov, J. Forlizzi, J. Zimmerman, Ux design innovation: Challenges for working with machine learning as a design material, in: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, CHI '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 278–288. URL: https://doi.org/10.1145/3025453.3025739. doi:10.1145/3025453.3025739.

[7] B. A. Myers, J. Stylos, Improving api usability, Commun. ACM 59 (2016) 62–69. URL: https://doi.org/10.1145/2896587. doi:10.1145/2896587.

[8] J. Cambre, C. Kulkarni, One voice fits all? social implications and research challenges of designing voices for smart devices, Proc. ACM Hum.-Comput. Interact. 3 (2019). URL: https://doi.org/10.1145/3359325. doi:10.1145/3359325.

[9] R. D. Pea, User centered system design: new perspectives on human-computer interaction, Journal educational computing research 3 (1987) 129–134.