

Frontend Diffusion: Exploring Intent-Based User Interfaces through Abstract-to-Detailed Task Transitions

Qinshi Zhang

qiz065@ucsd.edu

University of California, San Diego
San Diego, California, USA

Mohan Chi

chi70@purdue.edu

Purdue University

West Lafayette, Indiana, USA

Latisha Besariani Hendra

lbesarian2-c@my.cityu.edu.hk

City University of Hong Kong
Hong Kong, China

Zijian Ding*

ding@umd.edu

University of Maryland, College Park
College Park, Maryland, USA

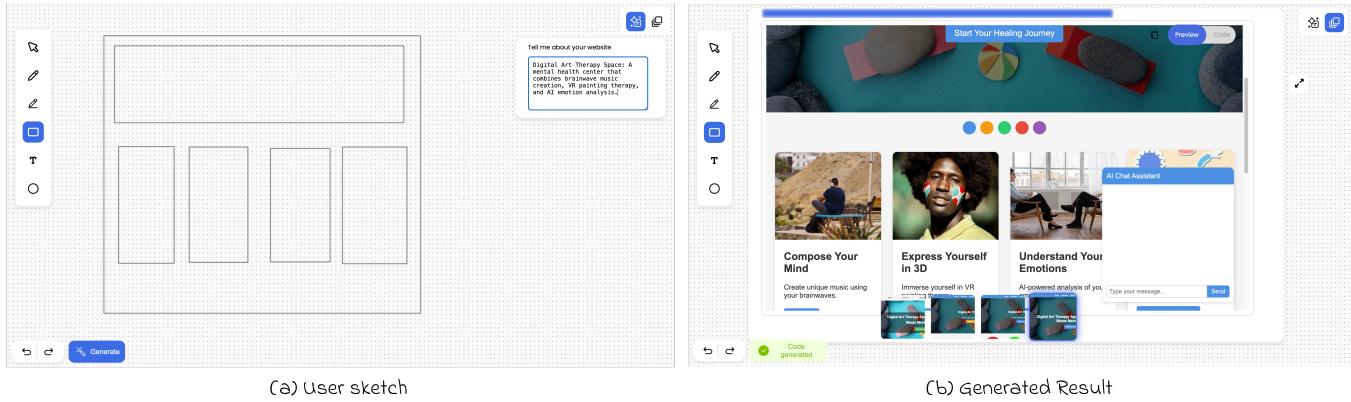


Figure 1: Frontend Diffusion is an end-to-end, LLM-powered tool for high-quality frontend code generation. Users can input a rough sketch on the canvas (left), which the system then transforms into a high-quality website page (right).

ABSTRACT

The emergence of Generative AI is catalyzing a paradigm shift in user interfaces from command-based to intent-based outcome specification. In this paper, we explore abstract-to-detailed task transitions in the context of frontend code generation as a step towards intent-based user interfaces, aiming to bridge the gap between abstract user intentions and concrete implementations. We introduce Frontend Diffusion, an end-to-end LLM-powered tool that generates high-quality websites from user sketches. The system employs a three-stage task transition process: sketching, writing, and coding. We demonstrate the potential of task transitions to reduce human intervention and communication costs in complex tasks. Our work also opens avenues for exploring similar approaches in other domains, potentially extending to more complex, interdependent tasks such as video production.

*Corresponding author

Permission to make digital or hard copies of part or all 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).

UIST '24 Adjunct, October 13–16, 2024, Pittsburgh, PA, USA

© 2024 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8655-5/21/10.

<https://doi.org/10.1145/3474349.3480203>

CCS CONCEPTS

- Human-centered computing → Interaction design theory, concepts and paradigms.

KEYWORDS

intent-based user interface, task transition, code generation

ACM Reference Format:

Qinshi Zhang, Latisha Besariani Hendra, Mohan Chi, and Zijian Ding. 2024. Frontend Diffusion: Exploring Intent-Based User Interfaces through Abstract-to-Detailed Task Transitions. In *The Adjunct Publication of the 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24 Adjunct)*, October 13–16, 2024, Pittsburgh, PA, USA. ACM, New York, NY, USA, 3 pages. <https://doi.org/10.1145/3474349.3480203>

1 INTRODUCTION

The development of Generative AI, particularly the capabilities of Large Language Models (LLMs) in interpreting and executing natural language, may be viewed as heralding the first new user interface paradigm shift in 60 years [8]. This shift moves from command-based interactions, typified by command line interfaces and graphical user interfaces, to intent-based outcome specification [8]. This emerging intent-based paradigm potentially enables users to communicate their intentions to machines without necessarily translating them into machine-comprehensible commands, whether

through programming languages or graphical buttons. This shift may foster interfaces that support more abstract human expressions, especially for command-intensive tasks such as coding [2, 3]. Currently, the interfaces for those tasks continue to necessitate substantial human intervention, where individuals typically specify incremental steps while LLMs generates corresponding code, akin to agile programming [11].

However, ongoing advancements in LLMs suggest the potential for developing a framework that may bridge the gap between intent-level expression and command-level implementation, enhancing output quality while reducing the need for extensive human intervention. For example, previous research has demonstrated that LLMs can complete fixed-scope content curation tasks based on human intent without further intervention or intent iteration [6]. However, LLMs require greater human intervention for tasks involving increasing amounts of information [4, 7]. It motivates us to develop more effective scaffolding paradigm for LLMs to respond to human intent and complete tasks in an agent-like manner.

Recent research has indicated the feasibility of bridging intent expression in abstract tasks to concrete implementation at a more granular level. Examples include the transition from sketching to writing [1] and from design to data analysis [5]. Building upon these findings, we propose exploring more extensive intent-to-command transitions, such as progressing from sketching to writing (planning) and ultimately to coding (see Figure 2). Our choice of website frontend generation as a user interface coding task [10] is motivated by its similarity to sketching. In both cases, the code or sketch serves as a representation of visual elements [9].



Figure 2: Task transition paradigm to bridge more abstract, intent-based task expression to more concrete, command-based implementations. The task transition process flows from sketching (sketch and theme inputs) to writing (product requirement document generation) and coding (frontend code generation and iteration).

2 SYSTEM DESIGN

We introduce Frontend Diffusion¹, an end-to-end LLM-powered high-quality frontend code generation tool, spanning from sketching canvas to website previews. As outlined in the introduction, the frontend generation task progresses through three stages: sketching, writing, and coding. Our system currently utilizes the Claude 3.5 Sonnet language model (Sonnet)² for all text and code generation. While Claude represents one of the most advanced language models as of July 2024, we anticipate rapid developments in LLMs.

¹<https://github.com/Carolzhangzz/frontendiffusion>

²<https://www.anthropic.com/news/clause-3-5-sonnet>

Therefore, the task transition techniques described herein are designed to be model-agnostic, ensuring their applicability to future, more advanced LLMs.

2.1 Sketching: Visual Layout Design and Theme Input

The system's initial phase comprises a graphical user interface with two key components: a canvas panel for visual representation of the envisioned website layout, and a prompt panel for textual descriptions of the website theme. Upon completion of the user's sketch and theme input, the user can activate the code generation process via "Generate" button. The system then converts the sketch into SVG format, followed by a subsequent transformation into JPG format. This two-step conversion process was implemented based on empirical evidence from our tests, showing that language models exhibit better performance when processing images in JPG format compared to images in SVG format.

2.2 Writing: Product Requirements Document Generation

This phase transforms the user's visual and textual inputs into a structured document, referred to as the Product Requirements Document (PRD), which serves as a blueprint for the website's development process. The PRD generation process leverages Sonnet. To enhance the visual appearance of the generated websites, the system integrates the Pexels API³ for image retrieval. The language model is specifically prompted to include image terms and size descriptions (e.g., [school(large)]). These descriptors are subsequently utilized to query the Pexels API, which returns relevant image URLs for incorporation into the PRD.

2.3 Coding: Website Generation and Iterative Refinement

The coding phase of the system consists of two primary components: (1) Initial code generation: the system utilizes the generated PRD and the original user prompt as inputs for code generation, employing Sonnet to produce the initial website code; (2) Iterative refinement: the system implements an iterative refinement process to automatically enhance the generated website with richer functionality and reduced flaws. This process involves analyzing the initial code to generate optimization suggestions, merging these suggestions with the original theme, and utilizing the enhanced theme along with the previously generated PRD to regenerate the code. The system executes this iterative refinement process multiple times (by default, n=4). Users can navigate between iterations by selecting preview thumbnails displayed at the interface's bottom, and can access or copy the generated code for each version.

3 RESULTS AND DISCUSSION

The generated websites, as illustrated in Figure 3, exhibit generally satisfactory visual appearances. These include contextually appropriate textual content, imagery, color schemes, layouts, and functionalities. Those results align with our "intent-based" objective of only requiring users to express their intent and scaffolding

³<https://www.pexels.com/api>

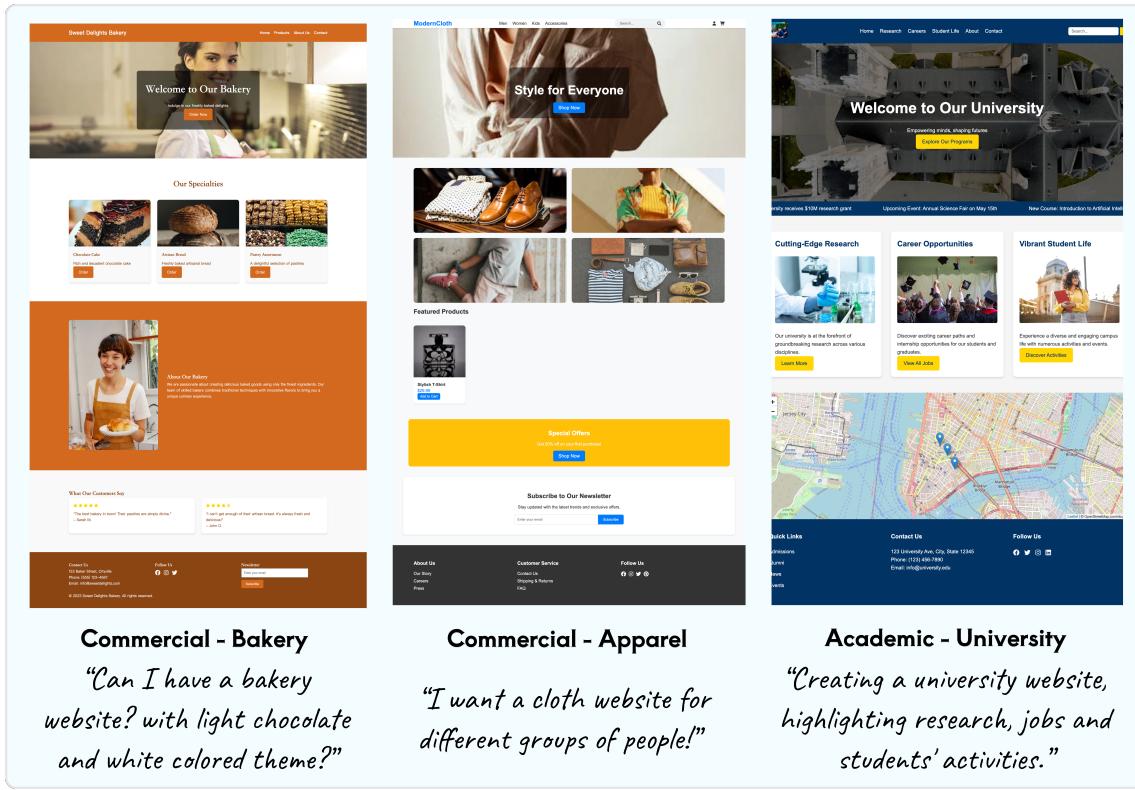


Figure 3: Screenshots of generated websites spanning commercial and academic domains. The theme prompt as user intent to generate each website is shown beneath the corresponding website.

Generative AI to deliver the final output, potentially reducing the communication costs between users and Generative AI systems.

This task transition paradigms may motivate further exploration of intent-based interfaces, potentially extending to more complex tasks with interdependent components such as video generation. For example, we might envision an abstract-to-detailed task transition process for generating video advertisements that begins with sketches and thematic inputs, transitions to script writing, then proceeds to generate textual and visual descriptions of storyboards, followed by video generating and editing, and culminating in iterative video refinement. We aim to further investigate the potential of intent-based user interfaces in streamlining complex, interdependent workflows across various domains.

Future work could focus on studies empirically validating the effectiveness of this task transition approach in more diverse and complex task environments. Additionally, research into optimizing the task transition process and enhancing the quality of inter-task communication may yield improvements in the overall performances.

REFERENCES

- [1] John Joon Young Chung, Wooseok Kim, Kang Min Yoo, Hwaran Lee, Eytan Adar, and Minsuk Chang. 2022. TaleBrush: Visual Sketching of Story Generation with Pretrained Language Models. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*. ACM, New Orleans LA USA, 1–4. <https://doi.org/10.1145/3491101.3519873>
- [2] Zijian Ding. 2024. Advancing GUI for Generative AI: Charting the Design Space of Human-AI Interactions through Task Creativity and Complexity. In *Companion*
- [3] Zijian Ding. 2024. Towards Intent-based User Interfaces: Charting the Design Space of Intent-AI Interactions Across Task Types. *arXiv preprint arXiv:2404.18196* (2024).
- [4] Zijian Ding and Joel Chan. 2023. Mapping the Design Space of Interactions in Human-AI Text Co-creation Tasks. <http://arxiv.org/abs/2303.06430> [cs].
- [5] Zijian Ding and Joel Chan. 2024. Intelligent Canvas: Enabling Design-Like Exploratory Visual Data Analysis through Rapid Prototyping, Iteration and Curation. *arXiv preprint arXiv:2402.08812* (2024).
- [6] Zijian Ding, Alison Smith-Renner, Wenjuan Zhang, Joel R. Tetreault, and Alejandro Jaimes. 2023. Harnessing the Power of LLMs: Evaluating Human-AI Text Co-Creation through the Lens of News Headline Generation. <http://arxiv.org/abs/2310.10706> arXiv:2310.10706 [cs].
- [7] Zijian Ding, Arvind Srinivasan, Stephen Macneil, and Joel Chan. 2023. Fluid Transformers and Creative Analogies: Exploring Large Language Models' Capacity for Augmenting Cross-Domain Analogical Creativity. In *Proceedings of the 15th Conference on Creativity and Cognition (C&C '23)*. Association for Computing Machinery, New York, NY, USA, 489–505. <https://doi.org/10.1145/3591196.3593516>
- [8] Jakob Nielsen. 2023. AI: First New UI Paradigm in 60 Years. *Nielsen Norman Group* 18, 06 (2023), 2023.
- [9] Chenglei Si, Yanzhe Zhang, Zhengyuan Yang, Ruibo Liu, and Diyi Yang. 2024. Design2Code: How Far Are We From Automating Front-End Engineering? <http://arxiv.org/abs/2403.03163> arXiv:2403.03163 [cs].
- [10] Jason Wu, Eldon Schoop, Alan Leung, Titus Barik, Jeffrey P. Bigham, and Jeffrey Nichols. 2024. UICoder: Finetuning Large Language Models to Generate User Interface Code through Automated Feedback. <http://arxiv.org/abs/2406.07739> arXiv:2406.07739 [cs].
- [11] Chen Zhu-Tian, Zeyu Xiong, Xiaoshuo Yao, and Elena Glassman. 2024. Sketch Then Generate: Providing Incremental User Feedback and Guiding LLM Code Generation through Language-Oriented Code Sketches. <http://arxiv.org/abs/2405.03998> arXiv:2405.03998 [cs].