On the Regulatory Potential of User Interfaces for AI Agent Governance

K. J. Kevin Feng[ω*] **Tae Soo Kim**[κ*] **Rock Yuren Pang**[ω*] **Faria Huq**[Γ]
Tal August[Ψ] **Amy X. Zhang**[ω]
[ω]University of Washington [κ]KAIST [Γ]Carnegie Mellon University [Ψ]UIUC
kjfeng@uw.edu

Abstract

AI agents that take actions in their environment autonomously over extended time horizons require robust governance interventions to curb their potentially consequential risks. Prior proposals for governing AI agents primarily target system-level safeguards (e.g., prompt injection monitors) or agent infrastructure (e.g., agent IDs). In this work, we explore a complementary approach: regulating *user interfaces* of AI agents as a way of enforcing transparency and behavioral requirements that then demand changes at the system and/or infrastructure levels. Specifically, we analyze 22 existing agentic systems to identify UI elements that play key roles in human-agent interaction and communication. We then synthesize those elements into six high-level interaction design patterns that hold regulatory potential (e.g., requiring agent memory to be editable). We conclude with policy recommendations based on our analysis. Our work exposes a new surface for regulatory action that supplements previous proposals for practical AI agent governance.

1 Introduction

AI agents—compound AI systems that take actions in their environment on behalf of a user under limited direct supervision—are increasingly being built and deployed into the real world [4, 7, 19, 20, 29, 31, 34, 45]. Modern agents have demonstrated proficiency in autonomously pursuing complex, multi-step goals in economically-valuable domains including software engineering [28, 51], online shopping [52], and machine learning research [8, 48]. Their proficiency in autonomous task completion is rapidly improving [36], a source of both excitement and concern. Agents may unlock new levels of productivity and economic growth, but their deployment is accompanied by heightened risks [3, 45]. These risks come from the inherent difficulty in anticipating these risks [4], increased attack surfaces for malicious actors [15, 33, 42], accelerating the gradual disempowerment of humans [32], and a general loss of human control over AI systems [2].

Prior work in agent governance proposed ways to mitigate risks, primarily through two avenues: *system-level safeguards* and *agent infrastructure* [5–7, 40, 41, 45, 47]. System-level safeguards fortify the AI model or agent scaffolding; it includes techniques such as training a model to seek user confirmation for sensitive actions or proactively refuse harmful requests, and building monitors for prompt injection attacks [40, 41]. Agent infrastructure is external to the system but still mediates and influences agents' interactions with their environments; these include agent IDs [6, 37], isolated channels for agent activities [7], and frameworks for authenticated delegation of authority [47]. While promising, these approaches operate "behind the scenes"—in model training or system architecture, limiting end users' visibility into and control over agent behavior during actual deployment.

[*]Equal contribution.

Workshop on Regulatable ML at the 39th Conference on Neural Information Processing Systems (NeurIPS 2025).

In this work, we address this challenge by proposing a new target for regulation in user-facing agentic systems: *the user interface* (UI). Regulation of UIs and UI elements is a well-established practice outside of AI for ensuring consumer safety. In the US, UK, EU, and Canada, marketing emails are legally required to contain a button for one-click unsubscribe [16, 22, 23]. The Americans with Disabilities Act (ADA) and the EU Accessibility Act require UIs to be perceivable and operable by people with visual and motor impairments [1, 14]. The General Data Protection Regulation (GDPR) in the EU and the California Consumer Privacy Act (CCPA) both mandate the presentation of UIs for privacy consent management and opt-out controls [21, 49]. Because UI elements map to underlying system functionality and act as a key layer for user-system communication and control, regulating UIs of agents can jumpstart the implementation of certain system- and infrastructure-level interventions. An illustration of the conceptual differences between system-, infrastructure-, and UI-level interventions can be found in Appendix A.1.

We begin our investigation of the regulatory potential by analyzing 22 deployed agentic systems to identify UI elements in human-agent communication. We synthesize our analysis into six high-level design patterns that can serve as targets for regulation, mapping each pattern to previously proposed system- and infrastructure-level governance for agents. We conclude with policy recommendations and fruitful avenues of collaboration for the technical, policy, and design communities.

2 Method

We collected and analyzed 22 agentic systems from academic papers in human-AI interaction, product releases, and open-source projects. To be considered "agentic," the system needed to satisfy the following 4 inclusion criteria at the time of analysis: 1) Is publicly available; 2) Is an interactive software system; 3) Operates using multi-step workflows; and 4) Calls tools and/or executes actions. These criteria were inspired by previous literature (e.g., [3, 4, 19]), and partially for practical reasons (e.g., we cannot analyze a system in a rigorous, reproducible way if its details are not publicly available). Full definitions for our inclusion criteria can be found in Appendix A.3.

Three authors divided up the analysis using visual thematic analysis, a common method in HCI for identifying design patterns in UIs [18, 50]. Further details about our analysis process are in Appendix A.4. Examples from our analysis and the full list of agentic systems are also in the Appendix.

3 Six Design Patterns with Regulatory Potential

Our analysis yielded six design patterns (see visual examples in Appendix A.2) that can serve as levers for regulatory action. These are not exhaustive but a starting point—new patterns may emerge as more agentic systems are deployed and UIs evolve. Fortunately, strong incentives already exist for adopting these patterns due to usability benefits. However, we still consider it important for these patterns to be regulated to prevent developers from removing them out of convenience, market pressures, or A/B testing.

3.1 Visible thoughts, plans, and actions

Description. The agent's reasoning, planning, and actions are represented as a step-by-step sequence that users can trace in real time or retrospectively. This trace may appear inline with the chat, alongside the cursor, or alongside components on which the agent is acting. By surfacing both what the agent is doing and why, these interfaces make the decision-making process explicit.

Regulatory promises and challenges. Revealing the agent's reasoning and actions enhances transparency, accountability, and user oversight. It enables users to identify unsafe behavior, intervene, and calibrate trust in the system. However, these benefits depend on the faithfulness of the expressed reasoning to the agent's actual internal processes [10, 12]. Another challenge is managing granularity: too little detail undermines oversight, while too much risks overwhelming users and reducing usability.

Connections with prior governance proposals. Regulating this design pattern advances prior calls for agent oversight. Visible thought and action traces serve as an *oversight layer* in the agent infrastructure, as proposed by Chan et al. [7], and operationalizes Shavit et al.'s [45] proposed practice of *legibility of agent activity* to ensure the safety of agentic AI systems.

3.2 Mechanisms for control transfer

Description. Explicit transfer of control encompasses two complementary mechanisms: interruption and takeover. Interruption allows the user to pause or stop the agent's ongoing activities at any point. Takeover goes further by allowing the user to directly assume control over the task environment (e.g., OpenAI Operator, Orca [27]) or steps in the agent's workflow (e.g., Cocoa [17]).

Regulatory promises and challenges. Interruption and takeover controls are central to user agency and safety, allowing users to intervene directly in an agent's activity. Their effectiveness depends not only on whether the agent halts cleanly when interrupted, but also on how the system manages actions already in progress or recently completed—for example, whether it should abort immediately, finish the current step, or how to design safe rollback and follow-up procedures after a takeover.

Connections with prior governance proposals. Control transfer implements prior calls for oversight infrastructure [7] and practices for interruptibility [45]. Regulating this pattern can also address the concerns raised by Kolt regarding delegation and loyalty [31], by allowing users to reassert authority on their interest and agency, rather than relying on the agent to decide when to involve the human.

3.3 Watch mode

Description. While operating in environments with sensitive information (e.g., financial websites or email clients), agents requires the user's direct supervision and extracts limited information from the environment. For example, while OpenAI Operator and ChatGPT agent takes regular screenshots during its operational trajectory, it pauses this behavior when it enters watch mode. In Gumbo [43], the system filters out sensitive information from users' screen activities before using them as context.

Regulatory promises and challenges. "Watch mode" addresses privacy and security concerns by preventing agents from inadvertently capturing or handling sensitive information. There remains a tradeoff on the frequency of "watch mode". On one hand, detecting sensitive information must be robust to avoid false negatives. On the other hand, if triggered too often, it interrupts workflows with excessive control transfers. More, requiring user simply "watching" does not guarantee meaningful attention or control — users may become complacent or distracted.

Connection with prior governance proposals. This pattern directly implements authenticated delegation frameworks by ensuring human oversight remains active during sensitive operations [47]. Smiliar to the proposal to govern Tesla's Full Self-Driving (FSD) mode where drivers are required to keep their hands on the wheel but may still become inattentive. This is important, especially for high-stakes tasks such as activities involving sensitive information (e.g., banking or health).

3.4 Customizable rule-based governance

Description. The user can modify the agent's default behavior by specifying custom rules. This can include rules for how to perform consequential actions and conditions under which the agent seeks user approval. User-specified rules override any "self-approval" settings where the agent approves its own actions (e.g., Cursor's auto-run [11]).

Regulatory promises and challenges. Agents must robustly validate and follow user-specified rules, particularly when conditions are underspecified or ambiguous. For approval requests, agents might subtly nudge users toward approval with techniques such as reward hacking [44], undermining the protective intent of these mechanisms. Additionally, malicious rules (e.g., ones that disable the agent's built-in safeguards) will need to be detected and removed.

Connection with prior governance proposals. This pattern can operationalize multiple governance principles including oversight layers [7], task and resource scoping [47], constraining action-space and requiring approval [45], setting default behaviors [45], and frameworks for delegated authority [31]. Generally, by allowing users to specify custom rules through the agent's UI, regulators can improve the flexibility of one-size-fits-all governance approaches imposed by developers.

3.5 Inspectable and editable agent memory

Description. Users can inspect and edit (i.e, modify, delete, add to) the agent's memory, which often includes preferences and user characteristics automatically inferred by the agent throughout the

course of one or more interactions with the user. Further, options for inspecting and editing agent memory should be easily accessible and discoverable by the user. See examples from Cursor and Gumbo [43] in Appendix A.2.5.

Regulatory promises and challenges. Ensuring that users can easily access and edit an agent's memory is important for transparency, privacy, and agency. Memory may contain private information the user would not want the agent to consider in its outputs, or worse, accidentally exposed through adversarial attacks. The memory may also contain inaccurate inferences that users should be able to correct or delete. Regulations should further ensure that developers do not disincentivize users from taking these actions by hiding them deeply in a settings menu, which be done by requiring a minimum number of clicks to access them.

Connections with prior governance proposals. Kolt [31] raised the issue of information asymmetry in agent governance, where agents having access to information that humans do not place us in a vulnerable position. Transparently exposing agents' memories and allowing users to edit them can help alleviate this problem by aligning human and agent information sources.

3.6 Sandboxes for agents with low-level environmental control

Description. In systems that expose low-level control of the operational environment to the agent (e.g., access to the terminal on a computer), the UI clearly shows the sandboxed nature of agent activity, *as well as information about the sandbox health*. Many systems display the former (see Appendix A.2.6 for examples), but not the latter. Sandbox health may include information such as its age and whether any evidence of activity leakage has been detected.

Regulatory promises and challenges. The advanced capabilities of agents may render sandboxes for traditional software ineffective [45]. Requiring sandboxes and their health to be displayed to users will encourage the development of monitoring methods that can detect when sandboxes may be broken by the agent or otherwise ineffective. Open questions include what metrics are best for tracking sandbox health, and how to test the robustness of sandboxes without incurring the risks from breakage. Poor metrics may deceive users into thinking a sandbox is healthy when it is in fact not.

Connections with prior governance proposals. Regulating this design pattern can help determine not only *how* an agent's action space is constrained [45], but also help monitor *the effectiveness* of that method. This is clearly already top-of-mind for developers using terminal-based coding agents, evidenced by the development of sandboxing tools like VibeKit[2].

4 Policy Recommendations

In light of our design patterns, we conclude with the following policy recommendations.

1. **Prioritize regulation based on (lack of) existing implementation incentives.** As mentioned in Section 3, incentives for implementing some of the described design patterns already exist, as they improve system usability. Thus, regulation should first target patterns with the weakest usability incentives. For example, many developers have voluntarily made an agent's thoughts visible, but few currently communicate sandbox health. The latter should then be prioritized as a regulatory target.
2. **Learn from lessons in dark patterns regulation.** The GDPR introduced a new wave of UI regulations to counter "dark patterns"—UI designs that steer users into making unintended, potentially harmful decisions for an online platform's benefit [35]. Agent UI regulation can draw from many lessons from dark patterns regulation. For example, setting up strategic collaborations between technical, policy, and design experts should be a top priority [24], as should empirical validation of effects on users to catch any unintended backfire effects [39].
3. **Prepare evaluators to verify adherence to UI regulation.** National AI safety institutes (e.g., US CAISI, UK AISI) and third party evaluation organizations (e.g., METR, Apollo Research) already partner with developers for pre-deployment safety evaluations of agentic systems [40]. Policymakers should work with evaluators to build new evaluations to verify whether a UI design pattern has been implemented and behaves as expected.

[2] `https://www.vibekit.sh/`

References

[1] ADA.gov. The Americans with Disabilities Act (ADA) protects people with disabilities from discrimination. https://www.ada.gov/.

[2] Yoshua Bengio, Sören Mindermann, Daniel Privitera, Tamay Besiroglu, Rishi Bommasani, Stephen Casper, Yejin Choi, Philip Fox, Ben Garfinkel, Danielle Goldfarb, Hoda Heidari, Anson Ho, Sayash Kapoor, Leila Khalatbari, Shayne Longpre, Sam Manning, Vasilios Mavroudis, Mantas Mazeika, Julian Michael, Jessica Newman, Kwan Yee Ng, Chinasa T. Okolo, Deborah Raji, Girish Sastry, Elizabeth Seger, Theodora Skeadas, Tobin South, Emma Strubell, Florian Tramèr, Lucia Velasco, Nicole Wheeler, Daron Acemoglu, Olubayo Adekanmbi, David Dalrymple, Thomas G. Dietterich, Edward W. Felten, Pascale Fung, Pierre-Olivier Gourinchas, Fredrik Heintz, Geoffrey Hinton, Nick Jennings, Andreas Krause, Susan Leavy, Percy Liang, Teresa Ludermir, Vidushi Marda, Helen Margetts, John McDermid, Jane Munga, Arvind Narayanan, Alondra Nelson, Clara Neppel, Alice Oh, Gopal Ramchurn, Stuart Russell, Marietje Schaake, Bernhard Schölkopf, Dawn Song, Alvaro Soto, Lee Tiedrich, Gaël Varoquaux, Andrew Yao, Ya-Qin Zhang, Olubunmi Ajala, Fahad Albalawi, Marwan Alserkal, Guillaume Avrin, Christian Busch, André Carlos Ponce de Leon Ferreira de Carvalho, Bronwyn Fox, Amandeep Singh Gill, Ahmet Halit Hatip, Juha Heikkilä, Chris Johnson, Gill Jolly, Ziv Katzir, Saif M. Khan, Hiroaki Kitano, Antonio Krüger, Kyoung Mu Lee, Dominic Vincent Ligot, José Ramón López Portillo, Oleksii Molchanovskyi, Andrea Monti, Nusu Mwamanzi, Mona Nemer, Nuria Oliver, Raquel Pezoa Rivera, Balaraman Ravindran, Hammam Riza, Crystal Rugege, Ciarán Seoighe, Jerry Sheehan, Haroon Sheikh, Denise Wong, and Yi Zeng. International ai safety report. Technical Report DSIT 2025/001, 2025. URL https://www.gov.uk/government/publications/international-ai-safety-report-2025.

[3] Stephen Casper, Luke Bailey, Rosco Hunter, Carson Ezell, Emma Cabalé, Michael Gerovitch, Stewart Slocum, Kevin Wei, Nikola Jurkovic, Ariba Khan, et al. The ai agent index. *arXiv preprint arXiv:2502.01635*, 2025.

[4] Alan Chan, Rebecca Salganik, Alva Markelius, Chris Pang, Nitarshan Rajkumar, Dmitrii Krasheninnikov, Lauro Langosco, Zhonghao He, Yawen Duan, Micah Carroll, et al. Harms from increasingly agentic algorithmic systems. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, pages 651–666, 2023.

[5] Alan Chan, Carson Ezell, Max Kaufmann, Kevin Wei, Lewis Hammond, Herbie Bradley, Emma Bluemke, Nitarshan Rajkumar, David Krueger, Noam Kolt, et al. Visibility into ai agents. In *Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency*, pages 958–973, 2024.

[6] Alan Chan, Noam Kolt, Peter Wills, Usman Anwar, Christian Schroeder de Witt, Nitarshan Rajkumar, Lewis Hammond, David Krueger, Lennart Heim, and Markus Anderljung. Ids for ai systems. *arXiv preprint arXiv:2406.12137*, 2024.

[7] Alan Chan, Kevin Wei, Sihao Huang, Nitarshan Rajkumar, Elija Perrier, Seth Lazar, Gillian K Hadfield, and Markus Anderljung. Infrastructure for ai agents. *arXiv preprint arXiv:2501.10114*, 2025.

[8] Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, et al. Mle-bench: Evaluating machine learning agents on machine learning engineering. *arXiv preprint arXiv:2410.07095*, 2024.

[9] Valerie Chen, Alan Zhu, Sebastian Zhao, Hussein Mozannar, David Sontag, and Ameet Talwalkar. Need help? designing proactive ai assistants for programming. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2025.

[10] Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, et al. Reasoning models don't always say what they think. *arXiv preprint arXiv:2505.05410*, 2025.

[11] Cursor. Agents overview. https://docs.cursor.com/en/agent/overview, 2025.

[12] Scott Emmons, Erik Jenner, David K Elson, Rif A Saurous, Senthooran Rajamanoharan, Heng Chen, Irhum Shafkat, and Rohin Shah. When chain of thought is necessary, language models struggle to evade monitors. *arXiv preprint arXiv:2507.05246*, 2025.

[13] Will Epperson, Gagan Bansal, Victor C Dibia, Adam Fourney, Jack Gerrits, Erkang Zhu, and Saleema Amershi. Interactive debugging and steering of multi-agent ai systems. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–15, 2025.

[14] European Commission. European accessibility act. `https://commission.europa.eu/strategy-and-policy/policies/justice-and-fundamental-rights/disability/union-equality-strategy-rights-persons-disabilities-2021-2030/european-accessibility-act_en`.

[15] Ivan Evtimov, Arman Zharmagambetov, Aaron Grattafiori, Chuan Guo, and Kamalika Chaudhuri. Wasp: Benchmarking web agent security against prompt injection attacks. *arXiv preprint arXiv:2504.18575*, 2025.

[16] Federal Trade Commission. CAN-SPAM Act: A Compliance Guide for Business. `https://www.ftc.gov/business-guidance/resources/can-spam-act-compliance-guide-business`.

[17] KJ Feng, Kevin Pu, Matt Latzke, Tal August, Pao Siangliulue, Jonathan Bragg, Daniel S Weld, Amy X Zhang, and Joseph Chee Chang. Cocoa: Co-planning and co-execution with ai agents. *arXiv preprint arXiv:2412.10999*, 2024.

[18] KJ Kevin Feng, Maxwell James Coppock, and David W McDonald. How do ux practitioners communicate ai as a design material? artifacts, conceptions, and propositions. In *Proceedings of the 2023 ACM designing interactive systems conference*, pages 2263–2280, 2023.

[19] KJ Kevin Feng, David W McDonald, and Amy X Zhang. Levels of autonomy for ai agents. *arXiv preprint arXiv:2506.12469*, 2025.

[20] Iason Gabriel, Geoff Keeling, Arianna Manzini, and James Evans. We need a new ethics for a world of ai agents. *Nature*, 644(8075):38–40, August 2025. doi: 10.1038/d41586-025-02454-5. URL `https://ideas.repec.org/a/nat/nature/v644y2025i8075d10.1038_d41586-025-02454-5.html`.

[21] GDPR.EU. Complete guide to GDPR compliance. `https://gdpr.eu/`.

[22] Government of Canada. Canada's anti-spam legislation. `https://ised-isde.canada.ca/site/canada-anti-spam-legislation/en`.

[23] GOV.UK. Marketing and advertising: the law. `https://www.gov.uk/marketing-advertising-law/direct-marketing`.

[24] Colin M Gray, Cristiana Santos, Nataliia Bielova, Michael Toth, and Damian Clifford. Dark patterns and the legal requirements of consent banners: An interaction criticism perspective. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, pages 1–18, 2021.

[25] Faria Huq, Zora Zhiruo Wang, Frank F Xu, Tianyue Ou, Shuyan Zhou, Jeffrey P Bigham, and Graham Neubig. Cowpilot: A framework for autonomous and human-agent collaborative web navigation. *arXiv preprint arXiv:2501.16609*, 2025.

[26] Interaction Design Foundation. Interaction Design Patterns. `https://www.interaction-design.org/literature/book/the-glossary-of-human-computer-interaction/interaction-design-patterns`.

[27] Peiling Jiang and Haijun Xia. Orca: Browsing at scale through user-driven and ai-facilitated orchestration across malleable webpages. *arXiv preprint arXiv:2505.22831*, 2025.

[28] Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? *arXiv preprint arXiv:2310.06770*, 2023.

[29] Sayash Kapoor, Benedikt Stroebl, Zachary S Siegel, Nitya Nadgir, and Arvind Narayanan. Ai agents that matter. *arXiv preprint arXiv:2407.01502*, 2024.

[30] Majeed Kazemitabaar, Jack Williams, Ian Drosos, Tovi Grossman, Austin Zachary Henley, Carina Negreanu, and Advait Sarkar. Improving steering and verification in ai-assisted data analysis with interactive task decomposition. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, pages 1–19, 2024.

[31] Noam Kolt. Governing ai agents. *arXiv preprint arXiv:2501.07913*, 2025.

[32] Jan Kulveit, Raymond Douglas, Nora Ammann, Deger Turan, David Krueger, and David Duvenaud. Position: Humanity faces existential risk from gradual disempowerment. In *Forty-second International Conference on Machine Learning Position Paper Track*, 2025.

[33] Priyanshu Kumar, Elaine Lau, Saranya Vijayakumar, Tu Trinh, Scale Red Team, Elaine Chang, Vaughn Robinson, Sean Hendryx, Shuyan Zhou, Matt Fredrikson, et al. Refusal-trained llms are easily jailbroken as browser agents. *arXiv preprint arXiv:2410.13886*, 2024.

[34] Seth Lazar. Governing the algorithmic city. *Philosophy & Public Affairs*, 53(2):102–168, 2025.

[35] Arunesh Mathur, Gunes Acar, Michael J Friedman, Eli Lucherini, Jonathan Mayer, Marshini Chetty, and Arvind Narayanan. Dark patterns at scale: Findings from a crawl of 11k shopping websites. *Proceedings of the ACM on human-computer interaction*, 3(CSCW):1–32, 2019.

[36] METR. Measuring AI Ability to Complete Long Tasks. `https://metr.org/blog/2025-03-19-measuring-ai-ability-to-complete-long-tasks/`, 03 2025.

[37] Microsoft. Agent Identity and Lifecycle. `https://microsoft.github.io/autogen/stable/user-guide/core-user-guide/core-concepts/agent-identity-and-lifecycle.html`, 2024.

[38] Hussein Mozannar, Gagan Bansal, Cheng Tan, Adam Fourney, Victor Dibia, Jingya Chen, Jack Gerrits, Tyler Payne, Matheus Kunzler Maldaner, Madeleine Grunde-McLaughlin, et al. Magentic-ui: Towards human-in-the-loop agentic systems. *arXiv preprint arXiv:2507.22358*, 2025.

[39] Midas Nouwens, Ilaria Liccardi, Michael Veale, David Karger, and Lalana Kagal. Dark patterns after the gdpr: Scraping consent pop-ups and demonstrating their influence. In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pages 1–13, 2020.

[40] OpenAI. ChatGPT Agent System Card. `https://cdn.openai.com/pdf/839e66fc-602c-48bf-81d3-b21eacc3459d/chatgpt_agent_system_card.pdf`, 2025.

[41] OpenAI. Operator System Card. `https://cdn.openai.com/operator_system_card.pdf`, 2025.

[42] Atharv Singh Patlan, Ashwin Hebbar, Pramod Viswanath, and Prateek Mittal. Context manipulation attacks: Web agents are susceptible to corrupted memory. *arXiv preprint arXiv:2506.17318*, 2025.

[43] Omar Shaikh, Shardul Sapkota, Shan Rizvi, Eric Horvitz, Joon Sung Park, Diyi Yang, and Michael S Bernstein. Creating general user models from computer use. *arXiv preprint arXiv:2505.10831*, 2025.

[44] Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R Johnston, et al. Towards understanding sycophancy in language models. *arXiv preprint arXiv:2310.13548*, 2023.

[45] Yonadav Shavit, Sandhini Agarwal, Miles Brundage, Steven Adler, Cullen O'Keefe, Rosie Campbell, Teddy Lee, Pamela Mishkin, Tyna Eloundou, Alan Hickey, et al. Practices for governing agentic ai systems. *Research Paper, OpenAI, December*, 2023.

[46] Amanpreet Singh, Joseph Chee Chang, Chloe Anastasiades, Dany Haddad, Aakanksha Naik, Amber Tanaka, Angele Zamarron, Cecile Nguyen, Jena D Hwang, Jason Dunkleberger, et al. Ai2 scholar qa: Organized literature synthesis with attribution. *arXiv preprint arXiv:2504.10861*, 2025.

[47] Tobin South, Samuele Marro, Thomas Hardjono, Robert Mahari, Cedric Deslandes Whitney, Dazza Greenwood, Alan Chan, and Alex Pentland. Authenticated delegation and authorized ai agents. *arXiv preprint arXiv:2501.09674*, 2025.

[48] Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, et al. Paperbench: Evaluating ai's ability to replicate ai research. *arXiv preprint arXiv:2504.01848*, 2025.

[49] State of California Department of Justice. California Consumer Privacy Act (CCPA). `https://oag.ca.gov/privacy/ccpa`.

[50] Martijn Van Welie and Hallvard Trætteberg. Interaction patterns in user interfaces. In *7th. Pattern Languages of Programs Conference*, pages 13–16, 2000.

[51] John Yang, Carlos E Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems*, 37:50528–50652, 2024.

[52] Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents. *Advances in Neural Information Processing Systems*, 35:20744–20757, 2022.

A Appendix

A.1 Illustration of different types of agent governance interventions

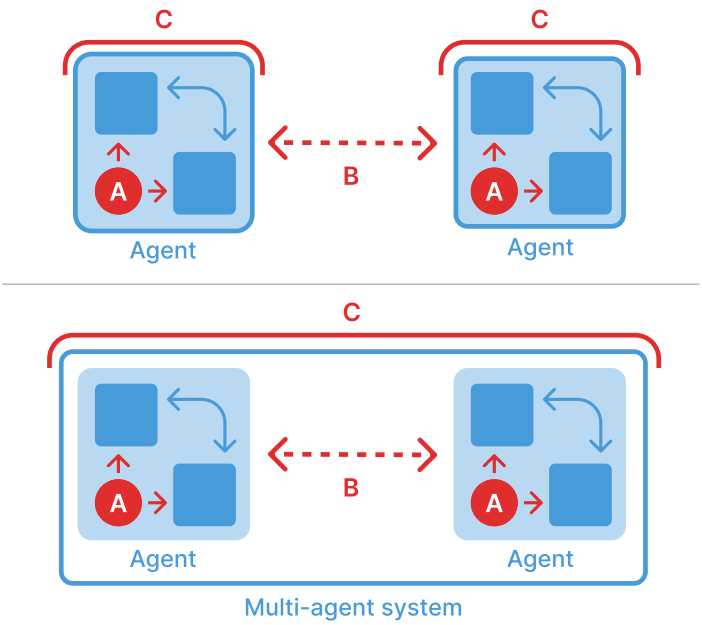
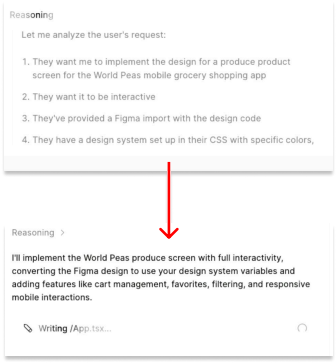


Figure 1: Governance interventions are shown in red for single-agent (top) and multi-agent (bottom) systems. **(A)** depicts a system-level intervention: a component (e.g., prompt injection monitor) that communicates with other components within the agent's architecture. **(B)** depicts an infrastructure-level intervention: a protocol through which two agents communicate. **(C)** depicts a UI-based intervention, such as controls for interrupting the agent mid-operation.

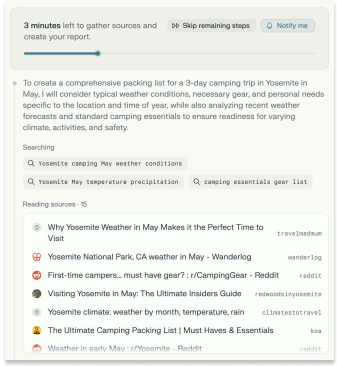
A.2 Examples of Design Patterns from Analysis

A.2.1 Visible thoughts, plans, and actions

Figma Make
Reasoning followed by actions



Perplexity Deep Research
Interleaved thoughts and actions



Ai2 Scholar QA
Plan with detailed steps & actions

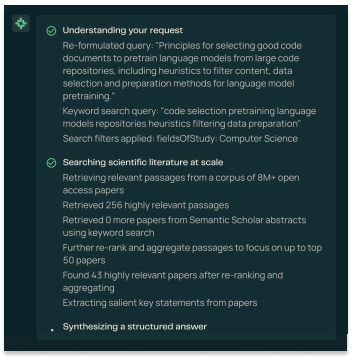
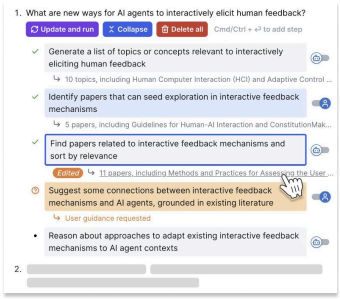


Figure 2: Examples of systems that make agentic reasoning and actions visible to the user, as part of the *visible thoughts, plans, and actions* design pattern.

Cocoa
Interactive and editable plan



Gemini Deep Research
User can request plan changes

Magentic-UI
Plan editable directly or via chat

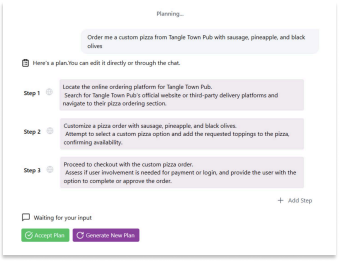
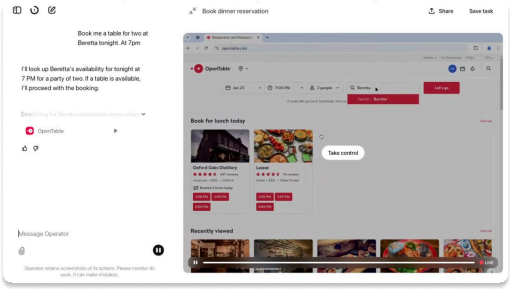


Figure 3: Examples of systems that provide a plan of action and allow the user to edit the plan before execution, as part of the *visible thoughts, plans, and actions* design pattern.

A.2.2 Mechanisms for control transfer

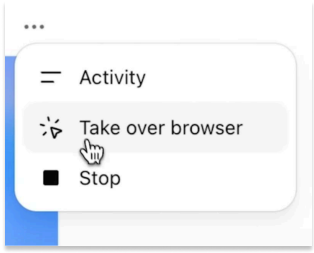
OpenAI Operator

Hover over virtual browser to take control



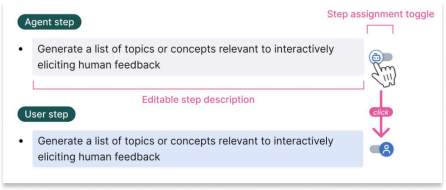
ChatGPT Agent

Take over option in menu



Cocoa

Toggle to assign step to agent or user



Orca

Hover over agent's cursor to take control

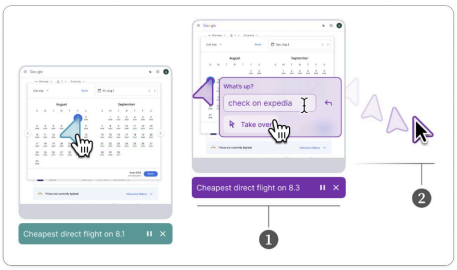


Figure 4: Examples of the *mechanisms for control transfer* design pattern.

A.2.3 Watch mode

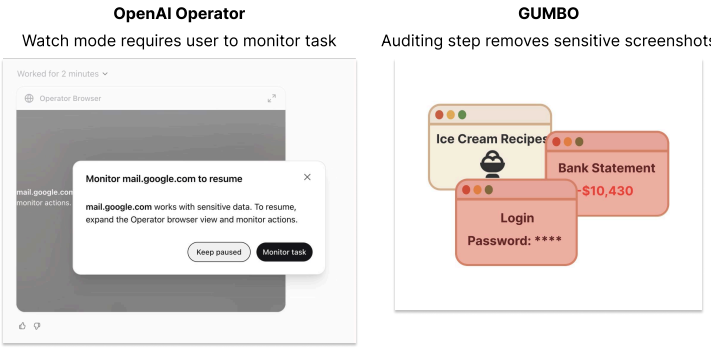


Figure 5: Examples of the *watch mode* design pattern.

A.2.4 Customizable approval-seeking conditions

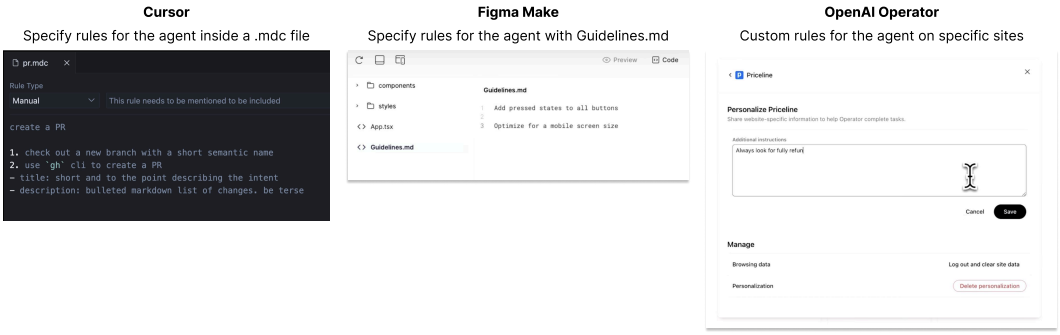


Figure 6: Examples of the *customizable approval-seeking conditions* design pattern.

A.2.5 Browsable and editable agent memory

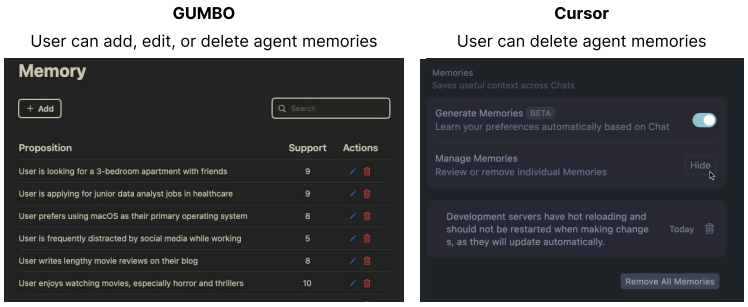


Figure 7: Examples of the *browsable and editable agent memory* design pattern.

A.2.6 Sandboxes for agents with low-level environmental control

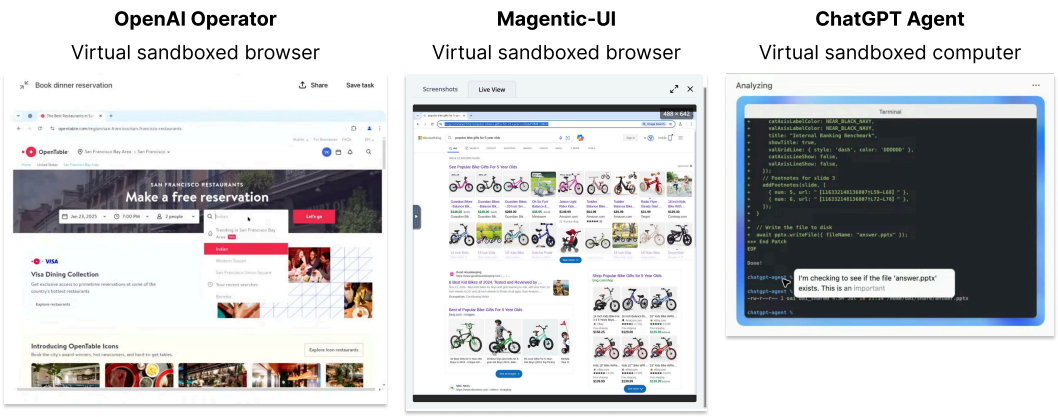


Figure 8: Examples of the *sandboxes* design pattern. Note that one of the systems analyzed have built-in indicators of sandbox status and health.

A.3 Inclusion Criterial Details

1. **Is publicly available.** The system is available to use for the public (e.g., not in private beta). Alternatively, the system has detailed documentation of its functionality and interactive components via a paper, blog post, and/or video demo available to the public.
2. **Is an interactive software system.** The system affords continuous user interaction through a graphical user interface (GUI) and/or a command line interface (CLI).
3. **Operates using multi-step workflows.** The system plans, reasons, and acts over two or more action-taking steps.
4. **Calls tools and/or executes actions.** The system uses external software tools (e.g., APIs) to perform actions that an unscaffolded LLM cannot perform alone.

A.4 Analysis Method Details

The authors read the available papers and watched the available demos, while also testing the system directly where possible. Screenshots were taken throughout the process to document individual elements within each agent UI. The UI element screenshots were collected in a shared FigJam[3] board, labeled with a short description of its functionality, and clustered based on common functionalites. The UI elements and labels were discussed at weekly meetings. The authors then synthesized the UI elements into six higher-level *interaction design patterns*—general design solutions to issues arising in UIs and UX [26]—based on functional similarity and repeated use across different systems.

[3]FigJam is a collaborative whiteboarding tool: `https://www.figma.com/figjam/`.

A.5 Agentic Systems Analyzed

Name	Domain	Environment	URL (system or paper)
AGDebugger [13]	Multi-agent systems	Specialized application	https://github.com/microsoft/agdebugger
Ai2 ScholarQA [46]	Scientific research	Web	https://scholarqa.allen.ai/chat
ChatGPT Agent	Computer use	Computer	https://openai.com/index/introducing-chatgpt-agent/
Claude Code	Coding	Terminal	https://www.anthropic.com/claude-code
Cocoa [17]	Scientific research	Specialized application	https://arxiv.org/abs/2412.10999
CowPilot [25]	Browser use	Web	https://arxiv.org/abs/2501.16609
Cove	General productivity	Specialized application	https://cove.ai
Cursor Agent Mode	Coding	IDE	https://docs.cursor.com/en/agent/modes#agent
Figma Make	Design	Specialized application	https://www.figma.com/make/
Gemini Deep Research	General productivity	Web	https://gemini.google/overview/deep-research/?hl=en
GitHub Copilot Agent Mode	Coding	IDE	https://github.com/features/copilot
Gumbo [43]	Computer use	Computer	https://arxiv.org/abs/2505.10831
Interactive task decomposition [30]	Data Analysis	Specialized application	https://arxiv.org/abs/2407.02651
Jules	Coding	Specialized application	https://blog.google/technology/google-labs/jules/
Lovable	Coding	Specialized application	https://lovable.dev/
Magentic-UI [38]	Browser use	Web	https://microsoft.github.io/magentic-ui/
Manus	General productivity	Web	https://manus.im/
OpenAI Deep Research	General productivity	Web	https://openai.com/index/introducing-deep-research/
OpenAI Operator	Browser use	Web	https://operator.chatgpt.com/
Orca [27]	Browser use	Web	https://orca.jiang.pl/
Perplexity Deep Research	General productivity	Web	https://www.perplexity.ai/hub/blog/introducing-perplexity-deep-research
Proactive programming assistant [9]	Coding	IDE	https://arxiv.org/abs/2410.04596

Table 1: List of agentic systems analyzed. We describe our inclusion criteria in Section 2. Citations are included for systems with academic papers.