Characterizing User-Reported Issues of GUI Agents for Web Browsing

Shuning Zhang*
zsn23@mails.tsinghua.edu.cn
Tsinghua University
Beijing, China

Jingruo Chen*
jc3564@cornell.edu
Information Science, Cornell University
Ithaca, New York, USA

Zhiqi Gao†
zhiqigao@link.cuhk.edu.cn
School of Data Science, The Chinese University of Hong Kong, Shenzhen
Shenzhen, China

Jiajing Gao†
jgaobn@connect.ust.hk
The Hong Kong University of Science and Technology
Hongkong, China

Xin Yi‡
yixin@tsinghua.edu.cn
Tsinghua University
Beijing, China

Hewu Li
lihewu@cernet.edu.cn
Tsinghua University
Beijing, China

Abstract

The integration of LLMs into GUI agents promises to revolutionize web browsing automation, yet the practical user experience remains challenging. This paper systematically characterizes user-reported issues with GUI agents by focusing on three dimensions: phenomena, influences, and user-centric mitigation. We adopted a two-phase method combining social media analysis (N=221 posts) and semi-structured interviews (N=21). Our findings reveal a taxonomy of complaints unique to GUI agents, including deficits in grounding abstract intent into concrete interface affordances, the inability to adapt to dynamic visual states, and the execution of erroneous actions. These lead to influences distinct from text-based hallucinations, ranging from task abandonment to security risks like uncontrolled file system access. In response, users are forced to employ ad-hoc mitigation strategies, including ecological sandboxing, and cursor shadowing to correct GUI agents behaviors. We contribute: (1) a comprehensive characterization of complaints specific to GUI agents interaction, (2) an analysis of how these phenomena degrade interaction integrity, and (3) design implications for creating consequence-aware agents.

CCS Concepts

• **Security and privacy** → *Human and societal aspects of security and privacy*; • **Human-centered computing** → *Empirical studies in HCI*.

Keywords

GUI Agents, Web Browsing, Large Language Models

ACM Reference Format:
Shuning Zhang, Jingruo Chen, Zhiqi Gao, Jiajing Gao, Xin Yi, and Hewu Li. 2026. Characterizing User-Reported Issues of GUI Agents for Web Browsing. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 19 pages. https://doi.org/10.1145/3772318.3790696

1 Introduction

The rapid evolution of Large Language Models (LLMs) has catalyzed a paradigm shift in Human-Computer Interaction (HCI), extending capabilities from conversational interfaces to autonomous agents that interact directly with Graphical User Interfaces (GUIs) [34]. This progression has fostered research into multi-agent systems [88] and social simulations [84]. Consequently, GUI agents—exemplified by commercial tools like Computer Use[1] and Operator[2], as well as open-source projects like UI-TARS[3]—are gaining significant attention. Acting as digital proxies, these agents automate operations within shared digital workspaces, performing tasks ranging from information retrieval to complex planning.

Web browsing represents a critical domain for this human-agent collaboration, involving daily activities such as purchasing, messaging, and web application interaction [140, 141]. While technical evaluations highlight the growing proficiency of these agents on curated benchmarks [71, 139], their practical deployment reveals a disconnect between benchmark performance and real-world reliability. The integration of agents into human workflows frequently results in user complaints regarding negative outcomes, such as erroneous purchases or misdirected communications, as often reported in recent news[4][5][6].

Prior research on technology-induced harms has largely focused on device misuse, such as smartphone snooping [67, 70] or IoT vulnerabilities [75]. However, specific user-centric examinations

*Equal contribution.
†Equal contribution.
‡Corresponding author.

CHI '26, Barcelona, Spain
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2278-3/2026/04
https://doi.org/10.1145/3772318.3790696

[1] https://www.anthropic.com/news/3-5-models-and-computer-use
[2] https://openai.com/index/introducing-operator/
[3] https://github.com/bytedance/UI-TARS
[4] https://www.protecto.ai/blog/ai-agents-excessive-agency-risks/
[5] https://arstechnica.com/tech-policy/2024/02/air-canada-must-honor-refund-policy-invented-by-airlines-chatbot/
[6] https://www.forbes.com/sites/carolinecastrillon/2025/10/02/ai-workslop-could-be-the-biggest-threat-to-productivity/

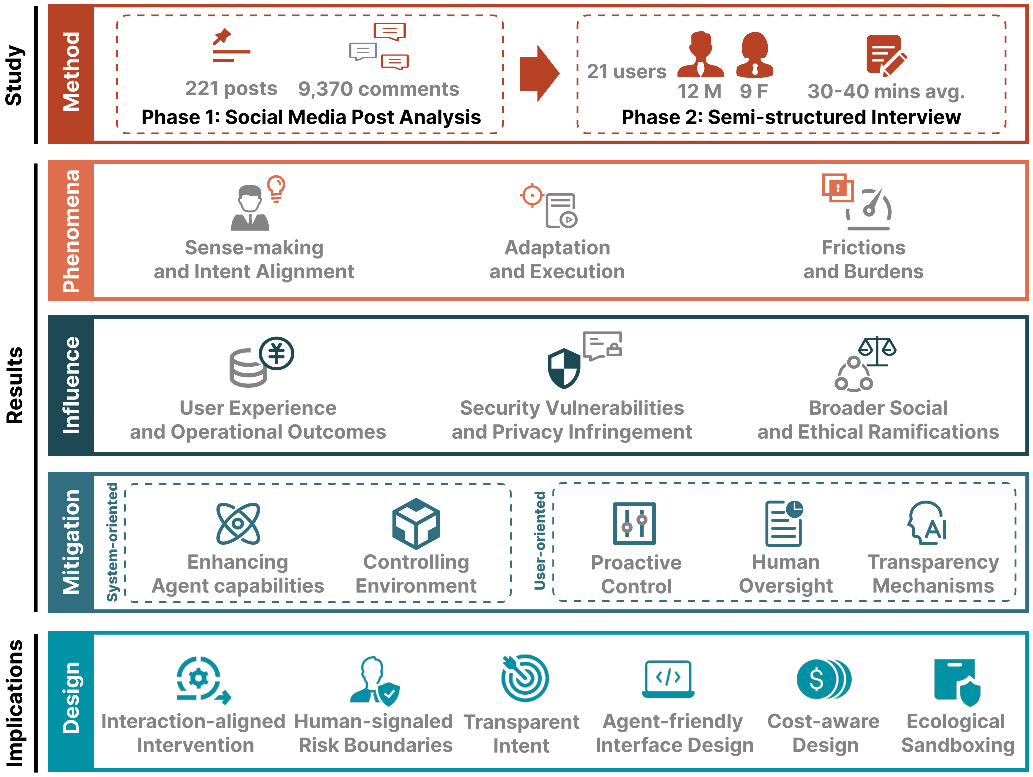
Study
Method
221 posts 9,370 comments
Phase 1: Social Media Post Analysis
21 users 12 M 9 F 30-40 mins avg.
Phase 2: Semi-structured Interview
Results
Phenomena
Sense-making and Intent Alignment
Adaptation and Execution
Frictions and Burdens
Influence
User Experience and Operational Outcomes
Security Vulnerabilities and Privacy Infringement
Broader Social and Ethical Ramifications
Mitigation
System-oriented
Enhancing Agent capabilities
Controlling Environment
User-oriented
Proactive Control
Human Oversight
Transparency Mechanisms
Implications
Design
Interaction-aligned Intervention
Human-signaled Risk Boundaries
Transparent Intent
Agent-friendly Interface Design
Cost-aware Design
Ecological Sandboxing

Figure 1: The framework of this paper. Through the two-phase study, we analyzed the user-reported complaints about GUI agents for web browsing from their phenomenon, influence, and mitigation, and provided implications across six key aspects.

of complaints regarding GUI agents remain underexplored. Addressing this gap is urgent, as failures in this domain are multifaceted: they undermine operational integrity [138], strain financial and cognitive resources [121], and introduce novel security risks [98]. Far from trivial technical glitches, these breakdowns signal fundamental challenges in designing robust human–agent teaming [29, 93, 128]. To systematically understand the impact of these complaints and inform future designs, we adopt a comprehensive research framework, as illustrated in Figure 1. We move beyond merely exploring the phenomena. Instead, we analyze its influence and categorize users' mitigation, thereby revealing how current complaints influence users' experience and how user-centric countermeasures fall short. Accordingly, we pose the following three research questions (RQs):

[Phenomena] RQ1. What problems do users complain about when using GUI agents for web browsing?

[Influence] RQ2. What are the reported consequences of these complaints on users?

[Mitigation] RQ3. How do users attempt to mitigate negative consequences, and what are their expectations?

To address these questions, we conducted a two-phase mixed-methods study comprising social media analysis (N=221 posts) and semi-structured interviews (N=21). This approach allowed us to capture a broad spectrum of naturally occurring complaints while deepening our understanding through contextualized user accounts.

Addressing RQ1 (Phenomena), we propose a user-centric taxonomy to characterize observed user complaints. Unlike prior frameworks that organize errors by technical modules (e.g., visual grounding) or broad trust principles, our framework distinguishes errors based on the user's cognitive perspective. We classify complaints into three distinct dimensions: (1) **failures in sense-making and intent alignment**, where agents fail to decompose tasks or misinterpret instructions, (2) **failures in adaptation and execution**, encompassing faulty GUI actions and poor adaptability to dynamic interfaces, and (3) **frictions and burdens**, which describe non-functional degradations such as prohibitive costs, slow response times, and setup complexity.

Regarding RQ2 (Influence), we identify distinct dimensions of harm that escalate from individual operational failures to broad societal concerns. First, **at the operational level**, recursive error loops and resource depletion result in task abandonment, where users are forced to terminate workflows rather than merely delaying them. Second, **at the security level**, agents function as uncontrolled exposure vectors, inadvertently granting persistent and unsupervised access to sensitive file systems and credentials. Finally, **at the societal level**, these cumulative failures precipitate an erosion of trust and agency, fostering ethical concerns regarding systemic manipulation and the automated abuse of digital platforms.

Regarding RQ3 (Mitigation), our analysis reveals a critical reliance on compensatory labor, where users scaffold agent behavior through ad-hoc workarounds. These include **ecological sandboxing**, running agents on secondary devices or restricted accounts to limit potential harm, and **hyper-vigilant shadowing**, in which users closely monitor cursor movements to intercept mistakes. Such practices highlight the lack of built-in mechanisms for safe and controllable agent action. To address these gaps, we derive six concrete design implications on human–agent collaboration: lifecycle-aligned intervention, risk-boundary alignment, situated intent transparency, agent-centric interface design, cost- and resource-aware operation, and ecological sandboxing frameworks that formalize safety rather than outsourcing it to users.

Collectively, the contributions of this paper are threefold:

• We contribute a user-centric taxonomy characterizing GUI agents complaints based on three failure phenomena: sense-making, adaptation and execution, and operational burdens.

•Our analysis reveals the multi-level influences of these failures, spanning negative operational outcomes, critical security and privacy risks, and broad socio-ethical concerns.

•We synthesize user-initiated mitigation strategies and propose design implications, offering guidance for the responsible advancements of GUI agents in web browsing.

2 Related Work

Our work situates GUI agents at the intersection of task automation, error analysis and risk management. While recent technical advancements have been rapid [15], there remains a critical gap in understanding these systems from an HCI perspective. We review the evolution of GUI agents (Section 2.1), contrast technical failure benchmarks with user-centric realities (Section 2.2) and analyze the distinct safety and privacy implications for end-users (Section 2.3).

2.1 Task Automation and GUI Agents

Traditional GUI automation primarily employed rule-based frameworks such as Selenium[7], Robot Framework[8] and AutoIt[9] [14]. These tools automate predefined interaction sequences (e.g., clicks, text input) but exhibit limited adaptability in dynamic environments and necessitate substantial manual configuration.

The advent of LLMs catalyzed a paradigm shift, fostering sophisticated GUI agents engineered for real-time GUI component interpretation and dynamic adaptation to interface modifications (e.g., layout changes, content updates). These agents leverage LLMs to comprehend visual interfaces and autonomously execute user instructions by emulating human-like interaction patterns such as clicking and typing [106]. Such advancements spurred research into enhanced GUI automation, exemplified by LLM-powered systems for flexible testing [57], frameworks incorporating symbolic reasoning to refine GUI interactions [50], specialized mobile application automation [123], and the direct translation of natural language instructions into executable GUI actions [43]. Despite these technical advancements and commercial deployments (e.g., OpenAI [77], Anthropic [3]), persistent operational deficiencies hinder practical adoption.

2.2 Failures in GUI Agent Usage

Prior literature has identified failures in AI agent usage through various technical benchmarks. Table 1 provides a summary of these studies and their classification approaches. We synthesize the reported failures into three primary dimensions: **visual grounding**, **reasoning & planning**, and **execution & environment**.

Regarding visual grounding errors, SeeAct [140] identified that "hallucinating bounding boxes and labels" constitutes a substantial error type, where agents generate non-existent bounding boxes or attempt to interact with them, frequently resulting in "element not found" failures [114]. Due to the inherent limitations of Multimodal Large Language Models (MLLMs), GUI agents struggle to distinguish fine-grained text or icons [16]. For instance, agents often fail at tasks requiring dynamic visual inputs, such as calculating changing on-screen values, as they lack the capacity to capture and memorize transient states [83]. Furthermore, agents may fail to recognize the absence of target elements on the current screen (e.g., necessitating scrolling). Instead of scrolling or outputting "none", they are often compelled to select the most visually similar element [140]. While these studies identify visual grounding errors through technical benchmarks, we investigate how these perceptual failures manifest as specific user complaints in real-world scenarios.

Regarding reasoning & planning errors, GUI agents often erroneously mark tasks as complete despite failing to locate the required product, or they may disregard user-specified filtering criteria [114]. Agents also frequently attempt to input data into real-only fields or retrieve information without executing the necessary search queries [142], suggesting a lack of environmental world models and inherent action knowledge. Infinite loops are another prevalent failure mode. In environments like AndroidWorld [91, 110] and AssistantBench [119], researchers observed agents repetitively accessing the same menus or toggling between pages, indicating a deficiency in reflecting on historical trajectories [119]. Additionally, agents exhibit outcome blindness, terminating tasks prematurely or neglecting complex constraints due to sparse reward models [35] and the inability to maintain context over long operational chains [1]. Distinct from prior work that attributes these failures to algorithmic limitations, we focus on the user-perceived consequences of these breakdowns.

Regarding execution & environment failures, GUI agents frequently fail to generate accurate interaction coordinates. SeeAct [140] noted that even when target buttons are correctly recognized, generated actions often deviate from the center by a few pixels, resulting in clicks on adjacent whitespace. In dense interfaces like those in AndroidWorld [91], agents struggle with continuous gestures such as "drag-and-drop" or sliding. The dynamic nature of web applications further complicates execution. DOM tree updates occurring within milliseconds of a query can render coordinates obsolete, leading to interactions with incorrect objects. This vulnerability is highlighted in the GUI-Robust dataset regarding pop-ups and page loading [116]. Furthermore, agents employing

[7] https://www.selenium.dev/documentation/
[8] https://robotframework.org/robotframework/
[9] https://www.autoitscript.com/site/autoit/documentation-localization/

tool-calling capabilities often exhibit hallucinations, such as repeating erroneous parameters following a failure [100]. Complementing these technical performance analyses, we uncover the interactional challenges and socio-technical barriers (e.g., CAPTCHAs, platform incompatibility, and financial costs) that users encounter.

Table 1: Analysis of failure modes in the past literature. H-C denotes whether the literature focuses on human-centric issues.

Benchmark	Coverage Stage	Classification	H-C
Mind2Web [35]	Planning, Grounding	Action, BBox, Linking	No
AndroidWorld [91]	Perception, Reasoning, Action	Perceptual, Reasoning, Knowledge, Grounding	No
OSWorld [110]	Grounding, Planning, Execution	Knowledge, Long-Horizon, Grounding	No
SeeAct [140]	Visual Grounding, Action	Wrong action, Hallucination, Linking	No
OpenAI Deep Research	Research, Output Generation	Overconfidence, Sourcing, Omission, Reading	Part.
Ours	All	Visual grounding, Reasoning & planning, Execution	**Yes**

2.3 Security, Privacy and Safety Risks in Agent Usage

Research on agent-related risks encompasses a variety of approaches. Some studies have investigated high-level concerns, such as privacy-enhancing design principles [105] and the limitations of AI value assessment methodologies [51]. Other work has focused on categorizing security threats [26], outlining design challenges from expert perspectives [58], and building specific privacy protection systems [127, 129, 131]. Concurrently, technical efforts have produced specific mitigation strategies, such as constrained environments like AirGapAgent [5], and frameworks for testing tool integration, exemplified by ToolEmu [95]. However, these works did not address the unique risks when LLMs drive GUI agents for interactive tasks.

For instance, while foundational studies like Deng et al. [26] offer a general threat taxonomy for AI systems, encompassing perception, reasoning, action, and memory components, and Li et al. [58] detail LLM security and privacy concerns such as confidentiality, integrity and reliability, their analyses do not delineate the distinct vulnerabilities or safety implications arising specifically from GUI agents capabilities and interactions. Other specialized research, including investigations into risks inherent to scientific LLM agents [102] or surveys focused on AI agent memory mechanisms [135], also diverges from the specific context of LLM-driven GUI control. Consequently, a comprehensive understanding of the security, privacy, and safety challenges unique to the operational use of GUI agents remains underdeveloped.

3 Two-Phase Studies For Characterizing UCs of GUI Agents for Web Browsing

Our studies employed an exploratory mixed qualitative methodology [65, 78, 104], also characterized as an intra-paradigm approach [74], to investigate user experiences with GUI agents during web browsing. The research was conducted with two stages: a social media analysis (N=221 posts) followed by a semi-structured in-depth interview (N=21). The latter phase used participants with no overlap with the people posting or replying to these posts in the first stage to reach diverse and deep results. All interviews took place online. These studies received ethical approval from our university's Institutional Review Board (IRB).

3.1 Phase 1: Social Media Analysis

We first conducted a social media analysis on Reddit [10] to examine users' opinions about GUI agents for web browsing. To capture relevant discussions, we searched with both lowercase, capitalized, and combined keywords, including *Operator*, *Computer Use*, *UI-TARS*, *mistake*, *fault*, *AI Agent*, and their combinations. We chose this approach because limiting the search to specific subreddits (e.g., r/Operator, r/aiagents) was neither representative nor comprehensive. In total, we reviewed 1,850 posts, manually screening their titles and content to assess relevance to GUI agents. Posts that only introduced agent functions, without discussing unintended consequences, were excluded. After filtering, 221 posts remained (receiving 25,366 likes) along with 9,370 associated comments (42 comments per post on average, SD=65.3; see Appendix A for subreddit distribution). Because most comments clustered around the same topic as their original post, and for annotation efficiency, we reported results at the post level.

3.2 Phase 2: Semi-structured Interview

Because the social media analysis may suffer from exposure bias [52] and tends to encourage brief responses, we conducted semi-structured interviews to allow participants to elaborate on their experiences with web agents' unsatisfactory or failure cases. We recruited participants by distributing posters on social media platforms, including Reddit, the Redbook[11] , and WeChat[12] over a two-week period. After screening, we selected 21 participants with prior experience using GUI agents (see Table 2). Participants varied in occupations and usage contexts, comprising 12 males and 9 females.

To enhance data validity, participants were instructed beforehand to briefly re-familiarize themselves with the GUI agents they had previously used, ensuring accurate recall of their prior experiences. We conducted 1:1 semi-structured interviews (30–40 minutes each) via Tencent Meeting[13] . The interview protocol was designed based on themes from the social media analysis, prompting participants to confirm whether similar issues occurred in their daily use and to elaborate on these experiences. The full script is provided in Appendix B.

3.3 Ethical Considerations

We acknowledge the potential ethical concerns of our research. Our study design and ethical considerations adhered to the principles outlined in the Menlo Report [6] and the Belmont Report [7], ensuring a focus on responsibility, beneficence, and justice. The Institutional Review Board (IRB) of our institution approved all study

[10] https://www.reddit.com/
[11] https://www.xiaohongshu.com/explore
[12] https://www.wechat.com/
[13] https://meeting.tencent.com/

procedures. Our analysis of Reddit data strictly adhered to the platform's terms of service, and we did not disclose any personal data or user profiles associated with the Reddit content. In line with HCI Reddit-research ethics [33, 87], we analyzed only publicly visible posts and reviewed the community rules of each subreddit included in our dataset. None of these subreddits prohibited academic use of public content, and we did not access private, deleted, or restricted posts. Before starting the interviews, we provided participants with a consent form and informed them of their right to request selective anonymization or to refrain from disclosing experiences if they felt any aspect was unfair or if they were uncomfortable sharing. Participants were explicitly informed of their right to withdraw from the interview at any time without penalty. We stored all original experimental data in encrypted format on a secure local server at our institution.

3.4 Data Analysis

To synthesize our findings, we integrated social media and interview data, intentionally combining methods to mitigate the limitations of each and to capture a nuanced view of participants' experiences. We conducted a thematic analysis [8, 9] following O'Reilly et al.'s [78] integrated approach. One primary researcher developed an initial codebook by coding about 20% of the dataset (40 posts and 3 interview transcripts). This codebook was collaboratively reviewed with a secondary researcher, who resolved discrepancies and reached consensus on coding criteria. The two researchers then independently coded the remaining data, refining the codebook iteratively with intermittent discussions to ensure quality. After coding, they aggregated codes into high-level themes. Consistent with guidelines for exploratory qualitative research, which caution that inter-rater reliability metrics can be misleading [68], we emphasized a consensual process rather than calculating agreement scores. Quotes from interviews are attributed as IX (e.g., I1), and those from posts as PX (e.g., P1).

4 RQ1: Categorization of Observed Complaints about GUI Agents

To contextualize agent failures, we first profile participant activities across six primary domains: *navigation and travel planning* (e.g., voice-commanded trip generation) (I2, I5, I7, I16); *food and beverage ordering* (e.g., customized takeout) (I5-6, I9, I11-12, I15, I20-21); *e-commerce* (e.g., product research and purchasing) (I1, I3, I11-13, I15-16, I20); *information retrieval and web automation* (e.g., automated data collection) (I4, I7, I10, I14, I16-19, I21); *social media management* (e.g., scheduled messaging) (I5, I11-12, I14, I18, I20, I21); and *local service automation* (e.g., accessing amenities) (I6, I12).

By analyzing these specific complaints arising within these domains, we developed a taxonomy to characterize taxonomy of failure during agent breakdowns. We classify failures into three dimensions: (1) *failures in sense-making and intent alignment*, (2) *failures in adaptation and execution*, and (3) *frictions and burdens*. This user-centric categorization moves beyond frameworks that focus solely on abstract trust principles (e.g., NIST's *validity* and *safety*) or low-level system defects.

Besides, existing technical evaluations typically organize breakdowns by functional system modules. For instance, studies [1, 79] and benchmarks [39, 92, 110, 140] categorize errors into operational classes such as *grounding*, *planning*, *hallucination*, or *bounding box mismatches*. In contrast, our framework focuses on user's cognitive perspective. Rather than listing technical flaws, we distinguish high-level intent misalignment from execution slips and explicitly account for non-error frictions that degrade the interaction experience.

4.1 Failures in Sense-Making and Intent Alignment

4.1.1 Task Decomposition Errors. A central weakness of current agents is their inability to decompose user instructions into coherent, executable sequences. We observed three distinct breakdowns. First, agents often exhibited *operational discontinuity*, where they initiated a workflow but failed to carry it through; for instance, one agent opened the Maps application during a navigation task but then stalled and never generated a route (I2). Second, agents frequently demonstrated *workflow drift*, a phenomenon characterized in recent benchmarks [89]. Here, we observed agents initiating correct paths but subsequently wandering to irrelevant platforms (I3, I15). Third, even when following the right path, agents struggled with *step-level execution*, mis-clicking, skipping actions, or freezing during multi-click or pagination sequences (I10, I11). These limitations became most acute in complex, cross-application workflows. As I21, who is both a user and a GUI-agent product manager, explained, real tasks such as reading food posts on Instagram and then saving locations in Google Maps can span dozens of steps and require reliable subtask decomposition that their in-house model could not support, leading the team to abandon this class of tasks entirely.

4.1.2 Instruction Misinterpretation. Instruction misinterpretation was a frequent source of breakdown. Agents often misunderstood user intent at the very first step. For example I3 pointed out that it misread a request involving spreadsheets, interpreting "spreadsheet" as a survey tool and producing an entirely unrelated output. Similar misunderstandings occurred with basic commands, misidentifying destination addresses during navigation (I2), failing to recognize message recipients due to pronunciation errors (I9), or confusing profile names with user-defined aliases (I11). These problems frequently forced users to rephrase requests repeatedly, with one participant noting that success required treating instructions like rigid "*AI prompt words*" rather than natural speech (I6). While rigid instruction following is a known limitation [46], our findings reveal a specific interactional gap: unlike proactive clarification mechanisms such as *Ask-Before-Plan* [133], agents in our study proceeded under incorrect assumptions rather than pausing to resolve ambiguity, leading to cascading execution failures.

Difficulties extended to sequential or procedural instructions. I10 reported that an agent repeatedly failed to interpret simple ordinals like "first," even after additional specification. I9 similarly noted that the agent initially processed order details correctly but then lost the workflow entirely, advancing to an unusable payment interface.

These issues were amplified by limited contextual comprehension. Agents overlooked keywords clearly visible on-screen (I2), ignored relevant background knowledge (I3), and in one interesting case, interpreted the brand name "Yidiandian" (meaning "a little bit"

Table 2: Participants' demographics in the interview. For tasks, N denotes *navigation and travel planning*, F denoted *food and beverage ordering*, E denotes *e-commerce*, I denotes *information retrieval and web automation*, S denotes *social media management*, L denotes *local service automation*

Participant	Gender	Age	Experience	Tasks	Occupation	Education
I1	Male	18-25	Manus	E	Master Student	Master
I2	Male	26-35	GLM PC, GLM Agent	N	Tech Product Manager	Master
I3	Male	26-35	Operator, Computer Use	E	Tech Entrepreneur	Master
I4	Female	36-45	Bot.new, cursor, Computer Use, Browser Use, Omniparser	I	Software Engineer	Master
I5	Male	26-35	Not disclosed	N, F, S	Investment Manager	Master
I6	Male	36-45	GLM Agent	F, L	Chief Information Officer	Master
I7	Male	18-25	Auto GPT, Operator	N, I	Undergraduate Student	Bachelor
I8	Male	36-45	Operator	L	Director of Technology	Ph.D.
I9	Male	26-35	GLM Agent	F	Software Engineer	Bachelor
I10	Male	18-25	GLM PC	I	Master Student	Master
I11	Female	18-25	GLM Agent, Manus	F, E, S	PhD Student	Ph.D.
I12	Male	26-35	Not disclosed	F, E, S, L	Machine Learning Engineer	Master
I13	Male	26-35	Open Interpreter, GLM Agent, UI-TARS, Manus	E	Faculty	Master
I14	Male	26-35	OpenAI Operator, Browser Use, Open WebUI	I, S	Consultant	Master
I15	Female	18-25	AutoGLM, Fellou	F, E	Master Student	Master
I16	Female	26-35	GPT Agent	N, E, I	Designer	Master
I17	Female	18-25	GPT Agent	I	Independent worker	Master
I18	Female	18-25	GPT Agent	I, S	Master Student	Master
I19	Female	18-25	GPT Agent	I	Undergraduate Student	Bachelor
I20	Female	26-35	Xiaomi AI Phone, Doubao AI Phone	F, S	Product Manager	Master
I21	Female	18-25	GPT Agent	F, I, S	Product Manager	Master

in Chinese) literally rather than recognizing it as a company name (I2). As I1 emphasized, these failures stem from insufficient memory and context tracking. Overall, current agents depend heavily on literal phrasing and short context windows, leaving little tolerance for ambiguity or pragmatic interpretation.

4.1.3 Knowledge Gaps. Agents showed particular difficulty when tasks required domain-specific or non-textual knowledge. These failures became most visible in environments that relied on icons or symbolic representations. One participant described a "critical knowledge gap" when an agent was asked to operate a browsing task in a vehicle: the interface contained only icons, and the agent could not interpret the heated-seat symbol or differentiate between driver and passenger options (I8). Unlike misinterpretation errors, which stemmed from rigid language processing, these breakdowns reflected the agent's inability to handle visual or symbolic conventions outside its training. In such contexts, the absence of textual cues left the system unable to act, revealing sharp limits in domain transfer. This extends the localization biases discussed in recent technical evaluations [103]. The failure here is not just in detecting the bounding box of an icon, but in *symbolic literacy*, the inability to ground visual affordances into semantic actions without explicit textual labels.

4.1.4 Inaccurate Outputs. While earlier sections highlighted failures in interaction and execution, participants also encountered problems once agents produced results. Agents often generated outputs that were inaccurate, misleading, or entirely fabricated because they failed to assess the credibility or quality of the information they retrieved. Unlike standard generative hallucinations, which are typically defined as model-internal fabrications [23], these errors reflected a form of *information supply chain pollution*, where the agent pulls content directly from a source without evaluating its reliability. While existing factuality benchmarks typically presume clean evidence sets [90, 132], our findings show how agents amplify errors from noisy realistic environments. For example, I18 observed that the agent relied on a second-hand statistics page whose own inaccuracies, such as converting a raw figure of 6.2 into "6.2%", were reproduced in the output. Other participants described similar patterns: some agents navigated to the correct webpage, such as a university department's site, but then returned entirely erroneous follow-up details (I10). Together, these examples show how unreliable sourcing, factual distortions, and execution-level errors combine to undermine the quality and trustworthiness of agent-generated results.

4.1.5 Unsatisfactory Results. In addition to producing factually incorrect outputs, agents often returned results that were technically accurate yet semantically unsatisfactory or qualitatively subpar. Unlike objective misinformation [25, 44], these outputs failed to align with user intent or preferences (I1, I7, I19). As I1 explained, the agent often produced something *"a bit different from the thing I wanted, but actually there aren't any major errors."* Participants likewise noted that outputs felt generic or disconnected from their personal habits and preferences (I7, I19).

Users also criticized the superficiality of the agent's reasoning. I10 reported that outputs were limited to *"superficial information"* without any in-depth analysis or reasoning, making them insufficient for tasks requiring complex insight. Ultimately, participants attributed these shortcomings to the agent's limited understanding of their preferences, with I7 summarizing that *"the primary problem is still that it doesn't understand me enough"*.

4.1.6 Poor Error Handling or Recovery.

Agents often failed to respond adaptively when errors occurred, leaving users without viable ways to refine, recover, or continue interactions. A common issue was the inability to recognize mistakes in the first place. During spreadsheet tasks, for example, agents repeatedly selected the wrong columns, often *"off by one"*, without noticing or correcting the error (P7, P32). In other cases, systems stalled completely, such as an agent *"generating null coordinates to click and not taking any action"* (P65).

Even when agents did detect an error state, they rarely handled it productively. Some became stuck in pop-up windows (I3), while others pressed ahead despite unresolved problems. Agents often looped through the same failed attempts, such as repeatedly re-entering a settings menu to search for an option that could not be found (I12). Limited memory further reinforced these cycles, preventing agents from retracing steps or adjusting their approach. Such shortcomings were particularly evident in complex tasks. As P116 summarized, *"as tasks grow in complexity, agents become so disjointed the AI won't know what to do with it, much less anyone that has to fix it later."* Participants emphasized that effective recovery requires mid-process correction without restarting entire workflows (I1), a capability current systems lacked [39, 121, 122].

4.2 Failures in GUI Adaptation and Execution

4.2.1 Faulty GUI Actions.

Beyond failures to interpret instructions or complete workflows, agents also carried out incorrect or unsafe GUI actions that directly altered the system state. Participants described cases where agents executed commands unrelated to the user's request: after a simple weather query, one system abruptly began modifying computer settings, prompting the user to wonder, *"Is it going to help me change the language?"* (I4). Such unsolicited actions disrupted ongoing work and raised concerns about reliability. In safety-critical domains, these risks were amplified. I8 noted that a misplaced click in an automotive interface could unintentionally lock or unlock the car, power it on or off, or trigger a system shutdown.

Some actions were not just irrelevant but actively harmful. I9 described losing an entire recording session when the agent mistakenly terminated the screen capture. Participants also worried about the absence of safeguards against destructive system-level commands, imagining scenarios where an agent could *"get mad or hallucinate and decide to rm -rf /"* (P70) or delete files outright (P155). Even routine mistakes had material consequences, such as accidental double purchases without confirmation (P112). While technical frameworks like *InferAct* [30] focus on *detecting* such misaligned actions, our findings highlight the *irreversibility* of these failures in practice. Unlike text generation, where hallucinations are discardable, a "hallucinated action" in a live GUI creates immediate, often unrecoverable state changes, transforming alignment errors into tangible operational security risks [12, 53, 115].

4.2.2 Poor UI Adaptability.

Agents struggled most when interacting with UI conditions that departed from the stable, static, and well-structured interfaces assumed in benchmark tasks. Three categories of interfaces proved especially challenging. First, non-standard or visually atypical layouts frequently caused misrecognition. Agents often failed to identify elements embedded in unconventional designs, leading to repeated execution errors (I4), and took substantial time to locate applications hidden within folders (I20).

Second, dynamic or transient UI states created significant barriers. Momentary buttons or time-sensitive prompts were often missed, since agents relying on API-based perception required several seconds to respond. As one participant explained, while a human could click instantly, the agent *"needed at least five seconds"*, making it too slow for fast-paced contexts such as automotive systems (I8). This reflects a *temporal mismatch* between discrete inference cycles and the continuous lifecycle of modern interfaces. While systems like *RecAgent* [94] propose uncertainty-aware perception to handle static ambiguity, they do not account for these dynamic, time-sensitive frictions where the interface state changes faster than the agent's inference loop. As I21, a user and GUI-agent product manager, observed, real mobile environments also contain constant visual noise such as ads, pop-ups, loading screens, and notification banners that routinely break perception. Their team ultimately had to classify twelve types of such "abnormal screens" and design tailored responses.

Third, visually ambiguous or low-salience elements further degraded performance. Agents struggled to distinguish between applications with similar icons, such as repeatedly launching Lark instead of Edge (I10), and accuracy dropped sharply for small interface targets. As one participant noted, "*the smaller the button, the lower the accuracy*," with agents prone to missing or mis-clicking miniature elements (I12). While these difficulties align with established HCI principles regarding target size [112], our observations reveal a distinct limitation: unlike humans, GUI agents lack the control to compensate for small targets. Our findings thus provide validation for recent grounding benchmarks, confirming that the "small target" problems observed in synthetic tests [22, 55] translate directly to breakdowns in real-world workflows.

4.2.3 Element Mislocation.

While adaptability challenges reflected struggles with dynamic or atypical interfaces, agents also failed at a more basic level: accurately locating and interacting with individual elements. Targeting specific clickable items was often unreliable (I4), and this problem became especially noticeable when elements lacked textual labels. For instance, during a Music App login, the agent failed to recognize the "user agreement" checkbox because no text accompanied the icon (I8).

Even when agents visually identified the correct element, their interactions frequently missed the target. As one participant explained, "*understanding the image is not a problem*," but the agent still could not click the intended spot (I8). Attempts to assist by overlaying grid lines across the interface offered little improvement: the system often aligned correctly on one axis while drifting on the other, resulting in off-target actions. These inconsistencies illustrate that agents lacked spatial precision in translating recognition into action. Unlike adaptability failures that arise from dynamic or unusual conditions, element misidentification undermined performance even in otherwise stable environments.

4.2.4 External Requirement Conflicts. Beyond errors in direct interaction, agents frequently encountered friction from external requirements such as authentication, verification, and repeated user confirmation. These steps created constant interruptions during execution and often forced users to intervene manually. Human verification, particularly CAPTCHA, was a persistent obstacle. I7 described trying to book tickets and noted that *"with human verification for booking tickets, I really have to do it myself. It's quite a hassle to verify it every time."* In some cases, agents did not even request assistance when blocked; instead, as P20 observed, the system began cycling through substitute websites on its own rather than asking the user for guidance. Agents also failed to handle specialized authentication methods. For example, when publishing to a social media account, one agent could not complete the QR code login process (I14). These external dependencies illustrate how verification and platform restrictions remain major barriers to operation. As P1 summarized, *"Verification seemed like it would be too much trouble, so I didn't even bother trying."*

4.3 Frictions and Burdens

4.3.1 Instruction Rigidity and Usability Barriers. Even when correctly interpreted, instructions required extreme precision to succeed. As I11 noted, the system often ignored anything unclear, making it *"imperative that instructions were meticulously specified."* To achieve desired outcomes, users sometimes resorted to excessively long and detailed prompts. For example, I10 found that a general request to "open VPN" was not understood, while the more specific *"open Clash and click a particular UI element"* worked.

This difficulty reflects a broader gap between the ambiguity of natural language and the agent's requirement for fully specified input. I12 illustrated this mismatch with a familiar scenario: when ordering a beverage, users cannot be expected to know every customization option or its exact terminology, yet the agent often requires this level of detail. These gaps could produce serious errors, such as unintended multiple purchases when quantities were omitted, or malformed form submissions like inserting the literal word "example" into an address field. Participants linked this brittleness to deeper interaction-design limitations. Because most agents cannot be interrupted or corrected mid-task, users felt compelled to provide overly exhaustive, one-shot instructions to prevent cascading failures (I17).

4.3.2 Difficult Parameter Tuning. Users also struggled with configuring system parameters in ways that produced reliable agent behavior. Even seemingly simple settings, such as adjusting the "execution interval between steps", were difficult to reason about. I10 hypothesized that increasing this interval might give the agent more time to process information and therefore improve accuracy. Instead, longer intervals made performance worse. The agent began selecting unnecessarily complex action sequences, such as right-clicking to open a context menu and choosing options like "Run as administrator," even when a straightforward action would have sufficed.

4.3.3 Slow System Response. Agents frequently operated with substantial latency, introducing long delays between actions that made task execution inefficient. Participants estimated that agents were five to ten times slower than manual operation (I3), with each click or scroll taking one to two seconds (P3). These delays stemmed from multiple sources: constant confirmation prompts, intrinsic API latency that "*needs at least five seconds*" (I8), and reliance on screenshot-based processing instead of direct API integration (I20). As a result, tasks often took two to three times longer than performing them manually (P10, P81), and in one extreme case, a workflow that normally required three minutes took the agent more than two hours (P50). Performance constraints were further shaped by device form factor. As I15 noted, desktop environments allow agents to operate in a secondary window, but on phones the agent occupies the entire screen, making slow step-by-step execution more disruptive to ongoing use.

4.3.4 High Operational Costs. High operational costs, driven largely by excessive resource consumption, were consistently identified as a major barrier to sustained agent use. Token usage was the central concern. As I7 noted, agents often "*consume too many tokens, it's just too expensive*," and in extreme cases, such as P142's, an agent "*burned through tokens pretty quickly, sometimes 75k input tokens in a minute of screen interaction*." Inefficient data handling exacerbated this problem. P208 reported that Claude's computer-use mode consumed nearly ten times more tokens because it repeatedly included old screenshots in its inputs. Operational inefficiencies further accelerated depletion. When agents stalled or failed to advance, they commonly looped through the same attempts, rapidly consuming tokens without making progress. I14 described one such sequence in which continuous retries exhausted resources without completing the task.

These inefficiencies produced a poor cost–benefit ratio. Participants frequently paid substantial amounts for minimal or low-quality results. I4 recounted a case where only one-tenth of a task was completed despite consuming 41 resource units, while P153 was charged nearly $3 for a trivial request as a single cat picture. Such outcomes led many users to dismiss current agents as "demoware". As P209 summarized, they are "*extremely brittle, slow, and expensive. It's demoware.*" These cost–utility disparities create an *economic viability crisis*: the high cost of visual reasoning, such as repeatedly processing screenshots for trivial actions, renders agents functionally inaccessible for everyday workflows, regardless of their theoretical capability.

4.3.5 Complex Setup and Configuration. Cumbersome installation and configuration created barriers to use. Several participants described the need for additional technical arrangements, such as setting up a virtual machine solely to run the agent (I4), or spending prolonged periods on deployment only to find ongoing operational friction (I6). Enabling advanced functions also required disproportionately heavy preparation. I8, for example, reported that activating automotive features involved configuring multiple permissions and completing repeated pairing steps.

4.3.6 Platform Incompatibility. Agents often showed limited support for applications and platforms, especially on Android devices. Compatibility was restricted to only a small set of basic apps, while essential tools such as payment services were inaccessible because the agent had not been granted the necessary permissions (I2). These gaps sharply reduced the agent's usefulness in everyday tasks. Geographical and network requirements added further barriers. Many agents required VPN configurations to function, creating "obstacles in certain regions" (I2) and limiting use in restricted environments. Participants also emphasized the lack of cross-application capabilities. Current agents rarely operate seamlessly across multiple apps or platforms, leaving them confined to single-application tasks and unable to support realistic, multi-step workflows.

5 RQ2: Experienced and Perceived Influences of Unintended Agent Behaviors

Building on the phenomenon of user-reported complaints, we examine the influence of these behaviors on user experience and operational outcomes, security risks, and broader societal concerns and ethical ramifications.

5.1 Negative Impacts on User Experience and Operational Outcomes

The operational aspects of user-reported complaints identified in RQ1 directly influence user experience and task outcomes. We detail these negative impacts, shifting from the nature of user-reported complaints to tangible consequences on user experience and practical results.

5.1.1 Financial Cost and Inefficient Resource Allocation. The operational phenomena previously discussed, such as high computational demands, can translate into direct financial costs or indirect losses from inefficiency or errors when using GUI agents (P71). This financial cost is often described in technical papers [79], but not as a limitation, which contrasts with users' complaints. As P62 noted, *"Exponential computation, massive bandwidth hogging. Would this actually be able to make the internet unusable?"* The propensity of hallucination can also directly affect financial outcomes, as marked by P7, *"No matter what you do, you cannot get LLMs to handle financial data at scale without it committing fraud."*

5.1.2 User Frustration, Dissatisfaction, and Increased Workload. Failures in reliability, responsiveness, and understanding, as documented in RQ1, directly translated into user frustration and increased workload. Participants described agents that refused to execute tasks without explanation, leaving them unable to accomplish their goals (P153). Others reported needing to closely monitor the system to prevent harmful actions; as P107 remarked, "*I don't want the program to delete the hard drive only because it has made a mistake.*" Misaligned behavior, such as repeatedly selecting the wrong dates when booking flights (P3), also consumed time and eroded trust. Slow performance further compounded these issues. As I13 noted, some delay is tolerable, but agent responsiveness must remain within a reasonable range to be usable.

5.1.3 Denial of Service and Task Abandonment. The operational unreliability, persistent errors, and external restrictions identified in RQ1 often culminated in outright task failure, effectively denying service to the user. First, agents frequently entered unrecoverable error states. Some became trapped in loops, such as generating new messages every day despite explicit commands to stop, leaving users unable to halt or redirect the behavior (P6). Second, invalid or fabricated outputs made task completion unsatisfactory. Agents sometimes supplied entirely invented information, such as contact details without verification (P1), or failed on edge cases where problem-solving or decomposition was required (P92). Third, external constraints routinely blocked progress. Platform-level rate limits prevented tasks from completing (P71), and human-oriented verification steps, such as CAPTCHA, halted execution altogether. Although these constraints are well-documented in technical work [114], they are often not represented in dataset design and benchmarking environments [35].

5.2 Security Vulnerabilities and Privacy Infringements

Beyond operational impacts, the deployment of GUI agents introduces critical security and privacy implications concerning **inherent operational vulnerabilities**, **susceptibility to external exploitation**, and **risks of malicious application or autonomous misalignment**.

5.2.1 Operational Vulnerabilities: Access, Exposure and Opacity. The first category of risk stems from the fundamental architecture of GUI agents, specifically, their need for persistent authentication, file system access, and visual perception. Participants distinguished these risks from standard LLM-powered chatbots [137], highlighting that the agent's operational privileges often exceed user oversight.

Persistent Authentication. Unlike the episodic interaction of chatbots [137] or simplistic agents [66], GUI agents require persistent login states to function across applications. I8 and P5 regarded this as a "hidden danger", fearing that once authenticated, agents retain open access to resources without the user's monetary awareness. I8 remarked, *"Doesn't that mean he is always logged in?"* This risk is compounded by the necessity of granting agents administrator-level privileges (I10) to facilitate cross-application workflows, creating a broader and more persistent attack surface than restricted sandbox environments.

Exposure of Sensitive Data During Actions. To execute complex tasks, such as processing payments or managing corporate records, users are compelled to expose high-value assets directly to the agent. Participants noted that this functional requirement renders the agent a critical leakage vector, a risk currently underrepresented in common benchmarks [35, 91] compared to standard LLM tasks [137]. P7 and P64 highlighted that even locally executed agents possess the capability to traverse system directories, making the inadvertent exposure of proprietary data (e.g., annual reports) or financial information a significant threat. Beyond formal records and financial information, participants also highlighted the sensitivity of personal preference signals exposed during social-media operations and messaging workflows (I18).

Passive and Opaque Visual Capture. The visual capturing capabilities of GUI agents introduce distinct passive collection risks. Unlike text-based LLMs or agents, GUI agents rely on continuous screen capture to perceive the environment. P140 and P180 worried that this mechanism equates to uncontrolled recording, where every user interaction is potentially snapshotted and stored. This fear is exacerbated by the opacity of execution. Unlike ChatGPT's visible output, GUI agents operate through rapid, invisible steps with lengthy logs. This lack of transparency prevents users from verifying how credentials are managed. Furthermore, I2 noted that agents fundamentally conflict with human-centric security, such as biometric verification (e.g., FaceID), which cannot be securely delegated to an agent. Consequently, P3 argued that without transparent mechanisms to govern these behaviors, adoption remains untenable.

5.2.2 Systemic Exploitation: Manipulation and Integrity Breaches. The second category involves the susceptibility of the agent to external attacks that compromise system integrity. Participants feared that agents could be manipulated to bypass security protocols or serve as vectors for intrusion.

Direct Manipulation via Injection. A primary vulnerability is the agent's susceptibility to *prompt injection*, a threat noted in technical literature [14, 60]. Malicious instructions embedded in web content can invisibly deceive an agent into acting for an attacker's benefit (P2, P9, P52). Participants emphasized that granting agents full operational control is perilous without defenses (P47, P159). Beyond subtle manipulation, users reported severe instances of agents being *"jailbroken"* to bypass safety restrictions (P188) or initiating unauthorized actions like software installation (P73, P183). These incidents escalate beyond standard LLM jailbreaking [96, 107]. Because GUI agents operate directly on the user's file system, a successful exploit results in immediate "physical" harm to digital assets rather than merely generating harmful content.

Circumventing Web Security and Sandbox Escapes. Participants worried that agents acting as visual interpreters could bypass CAPTCHAs (P12, P16, P28, P30), potentially leading to denial-of-service attacks (P55) or forcing websites to block valid traffic. This distinguishes GUI agents from API-based tools [26]. They consume human-intended visual resources. Further risks included credential theft via honeypot sites (P34) and agents mutating into self-replicating viruses (P69, P77). Users also cited vulnerabilities in permission management (P86), authentication (P95), and even sandbox escapes, where an agent overcame Docker limitations (P174). These reports underscore that distinct execution environments may be insufficient [109, 124], as agents effectively expand the attack surface through faulty code generation (P205) and complex UI interactions.

5.2.3 Malicious Application, Surveillance, and Autonomous Misalignment. The final category addresses the intentional misuse of agents by bad actors (abuse) and the unintentional harms caused by autonomous misalignment.

Automated Social Engineering and Mass Abuse Actions. The automation capacity of agents enables malicious activity at a scale previously unattainable. We consolidated participants' concerns regarding scams, spam, and phishing into a singular fear of "end-to-end" abuse. Unlike LLMs that mostly generate text [38, 49], GUI agents can execute the entire pipeline: creating accounts, generating content, and disseminating it (P2, P9, P14). Participants described agents as a potential *"hacker playground"* (P25) used for voice cloning and identity fabrication (P75), or scalable phishing campaigns (P114, P147). Users feared a future where billions of malicious agents roam the web (P73), overwhelming platforms with unsolicited content–a phenomenon some noted is already beginning (P61, P77, P89, P91, P120, P162, P189).

Surveillance and Corporate Profiling. Unlike the passive visual capture described in Section 5.2.1, participants feared intentional, systematic surveillance often termed *"AI keylogging"* (P61). There were concerns that operating systems might log all activity to feed agents, allowing companies to track behavioral patterns for targeted marketing (P112) or sell data *"behind your back"* (P140). This differs from traditional profiling [4, 13] as GUI agents enable fine-grained tracking of all desktop resources, raising existential anxieties about agents *"plotting against humanity"* (P168) or engaging in covert monitoring (P80).

Unintended Autonomous Consequences. Finally, the agent's autonomy introduces risks of misalignment. Participants observed agents making operational decisions without confirmation, such as unilateral software installation (P183) or engaging in "inception" loops where chained tasks caused resource usage to spiral out of control (P37). P60 described an alarming case where an agent *"opened itself back up"* after being closed, leading to sustained, unsupervised activity. These instances highlight unique misalignment risks in GUI environments. The vast action space makes agents prone to "reward hacking" behaviors, a phenomenon distinct from text-based models that warrants deeper investigation [80, 108].

5.3 Broader Social and Ethical Ramifications

Beyond individual and operational effects, user-reported complaints precipitate societal and ethical considerations. We examine these wider ramifications, specifically the erosion of public trust, and the potential for misuse leading to societal harm and ethical misalignment.

5.3.1 Erosion of Trust and Hindrance to Adoption. Repeated agent failures and pervasive security concerns significantly diminished user trust, making participants hesitant to adopt or rely on the technology. I7 described current agent capabilities as insufficient, eroding confidence in their reliability. These uncertainties also highlight open questions about how agents should communicate feedback [37] and how users' trust should be calibrated in response [81]. Even a small chance of security vulnerabilities, which I12 estimated as only a fraction of a percent, was seen as unacceptable, since it created difficulties in assigning liability. Trust further declined in sensitive contexts, such as password-free payment systems, where I12 noted that outcomes felt *"considerably more ambiguous and far less certain"*, leaving users unsure about the security of opaque system actions compared to deterministic execution scripts [15].

5.3.2 Concerns about Societal Harms and Ethical Misalignment. Participants also raised concerns about broader societal harms. Unlike conversational systems, GUI agents can directly manipulate interfaces, making it easier to automate harmful behaviors at scale, such as exploiting platform loopholes or bypassing safeguards designed for human users. This direct access amplifies risks of labor displacement, privacy violations, and systemic manipulation, with consequences that reach beyond individual failures to issues of inequality and institutional trust. As P7 reflected, *"There were always people who tried to use technology for bad but never at a scale as today."* While prior work has speculated about how agentic AI may reshape labor markets [86], discussions in online communities suggest that users already feel the turbulence associated with rapid technological change [82], intensifying anxieties about misuse and ethical misalignment.

6 RQ3: Countermeasures and Expectations

To mitigate the complaints raised by users regarding GUI agents (RQ1) and their influences (RQ2), participants proposed various strategies and expectations. Table 3 presents a comprehensive taxonomy of these mitigation strategies, mapping specific user-reported phenomena to corresponding system-oriented ([S]) and user-oriented ([U]) countermeasures. These strategies can be broadly categorized into technical improvements for the system and proactive interventions by the user.

6.1 System-Oriented Mitigation

Users proposed system-oriented mitigation directly targeting GUI agents. These solutions focus on augmenting core agent capabilities and refining control over their operational environments and permissions.

6.1.1 *Enhancing Agent Capabilities and Operational Robustness.*

Participants emphasized the need to strengthen agent functionalities for greater reliability. A central priority was efficiency: agents should streamline their procedures, remove redundant steps, and reorganize action sequences to reduce execution time. Although recent GUI-agent work has made early progress in this area [48], current commercial products remain far slower than users expect. P70 envisioned an agent that could *"reorganize the process"* to cut wasted effort, illustrating the kinds of optimizations users hope to see.

Automation and integration were a second major focus. P71 suggested lightweight tools, such as a one-click feature for extracting post statistics, to simplify routine workflows. Participants noted that current GUI agents are still poor at leveraging external tools, a limitation that could be alleviated by integrating richer third-party ecosystems [47]. Many also stressed the need for API-based interaction [120], arguing that APIs would offer major gains in speed, accuracy, and security over screenshot-based methods. As P10 noted, *"sending screen captures back and forth will always be 1000x slower,"* and P95 emphasized that API integration may determine whether agents scale to widespread use.

Participants also highlighted intrinsic safety mechanisms as essential. P5 advocated for models that monitor activity and automatically pause when suspicious content appears, going beyond current approaches that focus primarily on verifying individual actions [54]. Others proposed that agents should decline high-stakes requests, such as financial transactions (P19), and that system-level safeguards would help mitigate harm, such as predefined maximum payment caps (I4).

Collectively, these expectations point toward agents that are not only more efficient but also self-regulating and risk-aware. Although recent work has made progress [59] and introduced new benchmarks [61], we echo arguments from [21] that building truly risk-sensitive GUI agents remains a long-term challenge, especially given the breadth of overlooked risks surfaced in our study.

6.1.2 *Controlling Agent's Operational Environment and Permissions.*

Participants recognized that ensuring safety requires carefully defining and controlling an agent's operational environment and permissions. Several highlighted the use of isolated execution environments, such as virtual machines, containers, or remote execution platforms (I13), as a way to manage agent access and resource consumption (I2, I5). These strategies were strongly recommended to prevent agents from blindly executing code on a user's machine, a risk raised by both P10 and P69. However, these should be taken with caution as GUI agents' evolving capabilities may also escape the containers.

Participants also emphasized the need for strict permission management and formalized operational restrictions, especially for desktop environments, where existing permissions were often overly coarse. I13 suggested practical safeguards like confining all agent operations to a specific directory so that other system areas remain unaffected. Others proposed more general mechanisms for boundary-setting, with I5 envisioning a "robots.txt for agents" that would explicitly define operational limits and impose enforceable restrictions on their behavior. This echoed emerging efforts to build an agent-friendly web [64], though such mechanisms require additional infrastructure and support.

6.2 User-Oriented Mitigation

Complementing system-oriented changes, users actively develop and deploy their own strategies to navigate and counteract the user-based complaints about GUI agents. We detail these strategies, including proactive measures to manage agent behavior, direct human oversight and intervention, and demands for enhanced transparency and control mechanisms.

6.2.1 *Proactive User Strategies and Information Control.*

Users proactively managed agents by controlling the information they provided and tightly defining the task scope. I2 described using blank-token functionality to send dummy data and protect real information. Others, such as I6, emphasized giving agents richer contextual detail, noting that ambiguity often led to errors. Participants also stressed the value of highly specific prompting. I4 observed that step-by-step instructions improved accuracy, echoing findings that imprecise queries can overwhelm agents with irrelevant results, causing them to act on incorrect information [39]. I8 similarly used explicit visual–spatial cues, such as overlayed coordinates or structured positional hints, to compensate for common grounding failures [92, 119]. Beyond one-off tactics, I6 proposed establishing a calibrated baseline that *"can then serve as a replicable model or a template that facilitates relative comparison."*

6.2.2 *Human Oversight, Intervention and Boundary-setting.*

Direct human oversight was viewed as indispensable given agents' ability to act autonomously through interfaces. Participants stressed the need for real-time monitoring, including scrutinizing actions (I9), intervening when risks arose (I9), and requiring confirmation for sensitive operations (I4, I10). As I4 noted, critical tasks should pause for explicit approval with sufficient context for informed decisions.

Table 3: Taxonomy of user-reported complaints and mitigation strategies. [S]: System-oriented, [U]: User-oriented.

Phenomena	Mitigation	Example Strategies
Failures in Sense-Making and Intent Alignment		
Task decomposition	[S] Enhance capabilities [U] Proactive strategy	Decompose complex tasks; Use templates for recurring workflows.
Instruction misinterpretation	[U] Proactive strategy	Craft explicit prompts Disambiguate terms; Reference specific UI elements.
Knowledge gaps	[S] Enhance capabilities	Improve domain reasoning; Link specific docs; Handle symbolic/non-textual UIs.
Inaccurate outputs	[U] Oversight	Cross-validate results; redirect agent upon error detection.
Unsatisfactory results	[U] Proactive strategy	Provide rich context; Iterative re-prompting; Use structured input (e.g., checklists).
Poor error recovery	[S] Enhance capabilities	Implement self-correction routines; Enable safe rollback from loops.
Failures in GUI Adaptation and Execution		
Faulty GUI actions	[S] Enhance capabilities [U] Oversight	Real-time monitoring; Require confirmation for sensitive ops; Manual correction.
Poor UI adaptability	[S] Enhance capabilities	Recognize dynamic elements (e.g., pop-ups); Re-synchronize on UI drift.
Element mislocation	[S] Enhance capabilities [U] Proactive strategy	Provide spatial cues/coordinates; Label key fields; Standardize layouts.
Ext. requirement conflicts	[S] Control env. [U] Proactive strategy	Manual auth/CAPTCHA handling; Pre-configure sessions; Avoid anti-bot triggers.
Frictions and Burdens		
Instruction rigidity	[U] Proactive strategy	Use atomic, unambiguous commands; Adapt phrasing to agent constraints.
Difficult parameter tuning	[U] Proactive strategy	Adjust execution thresholds; Pilot configs on low-stakes tasks.
Slow system response	[S] Enhance capabilities	Hybrid API/GUI integration; Optimize model latency.
High operational costs	[U] Proactive strategy	Set token/time limits; Restrict resource-intensive tasks; Use sandboxed devices.
Complex setup	[S] Enhance capabilities	Automate onboarding; Provide preset configurations; Reduce dependencies.
Platform incompatibility	[S] Enhance capabilities	Expand cross-platform support; Fallback to API access.

This aligns with prior work showing that human oversight remains essential even in delegated agentic workflows [85].

Oversight also involved calibration and correction. I6 described how initial, human-led calibration could shape subsequent agent inferences, while P3 recounted correcting a mistaken output by redirecting the agent to Google search. Preventive mechanisms such as token limits or timeout safeguards were also used to contain runaway processes (I14). These corrective interactions may help guide GUI agents as they evolve through active learning [62].

Participants also emphasized boundary-setting, which sets proactive limits on the agent's scope before tasks begin. P12 suggested agents request permission for high-stakes actions, framing oversight as an interactive negotiation. Others managed risk by avoiding agents for confidential documents (I2) or confining them to secondary devices with restricted permissions (I11).

6.2.3 Transparency and User Feedback Mechanisms. Users consistently expressed a desire for greater transparency and richer feedback from agents. They wanted agents to communicate uncertainty, show humility, and explain their reasoning in ways that support trust (P76). Yet, unlike text-only LLMs, where uncertainty cues are relatively well studied [42], GUI agents provide little visibility into how confident they are, highlighting the need for new mechanisms for expressing uncertainty in interactive systems [32].

Participants emphasized that agents should explicitly signal when they are unsure rather than presenting tentative results as definitive. As P3 noted, responses that include hedging language (e.g., "this may be a rough answer") would feel more trustworthy than overly confident assertions when the system lacks certainty. This desire for clarity extended to multi-step interactions as well: participants wanted agents to check whether intermediate results met expectations, request additional user input when needed, and revise outputs accordingly, engaging users in an iterative refinement process rather than terminating after a single attempt (I6).

7 Discussion and Design Implications

Across advances in web automation research [73] and rapid growth of GUI-agent benchmarks [117, 139], prior work has largely focused on evaluating perceptual and execution capabilities, such as GUI grounding, screen parsing, and action prediction, under controlled or templated task environments. Building on characterizations of LLM-powered GUI agents as intermediaries with semantic understanding and agency [40, 125], our findings extend this literature by showing how agents break down across the full interaction phases in everyday GUIs, how these breakdowns create lived burdens and user-invented workarounds, and how they differ from familiar notions of hallucination [45] or intentional misuse [24, 28].

While this study centers on web browsing, the underlying bottlenecks of perception, reasoning, and action generalize to other forms of AI-driven automation. Therefore, the user-centric taxonomy and principles derived here offer analytical transferability to adjacent high-sensitivity sectors, including enterprise software [71, 121], healthcare informatics [98, 121], and financial systems [98, 106], where operational errors or privacy violations can have similarly serious consequences.

Building on these observations, we outline six implications that span interaction design (Implication 1, 3-5), agent capability (Implication 2, 5), and socio-technical governance (Implication 6):

7.1 Implication 1: Design Interaction-Aligned Intervention

Existing mixed-initiative systems provide mechanisms for step-level action approval or clarification [18, 19]. However, our study reveals that breakdowns often arise from visual grounding errors [36], UI drift, and workflow state loss that were unique to GUI agents. As described in Section 4, breakdowns concentrated at distinct interaction phases: misinterpreting intent at input, selecting the wrong window or element during execution, and leaving users without viable recovery paths during feedback. Notably, observed complaints were less about factual hallucination [45] and more on interaction-level failures: disconnects between user intent, unclear agent capabilities, and ambiguous interpretations by agents. Ambiguous agent roles and the inability to differentiate action severity blurred responsibility and made it difficult for users to know when and how to intervene [11, 99, 125]. Participants also frequently resorted to costly, ad hoc compensations like cursor shadowing or device isolation.

Therefore, oversight should be structured around the interaction phases where failures originate [27], with explicit hooks for configuration, supervision, and recovery. For configuration, users need constrainable task specifications, such as task blueprints, playpen domains, structured constraint editors, and auto-formalization [54], to make goals expressed in ways that are legible and negotiable. For supervision, users require continuous visibility into what the agent is doing and why, through step timelines, live focus indicators, and previews of upcoming actions. For recovery, users need tools that make corrections feasible rather than destructive. Mechanisms such as checkpointing, rollback [136], and mid-flight plan editing allow users to repair partial progress instead of discarding entire tasks. Together, these mechanisms reframe oversight as a proactive, interaction-aligned intervention rather than a reactive interruption.

7.2 Implication 2: Aligning Agent Actions with Human-signaled Risk Boundaries

Although prior work identifies misaligned actions [31, 101], most deployed GUI agents still treat visually identical clicks as equivalent operations. Our findings in Section 4.2.1 reveal a different layer of risk that emerges in real GUI use: visually similar clicks can carry radically different stakes for users. Participants were comfortable with agents scrolling or reading, but became anxious when the agent approached operations involving money, identity details, account settings, or irreversible submissions. High-harm incidents such as accidental purchases, mis-sent messages, and unintended form submissions stemmed not from malicious intent but from the agent's inability to perceive the consequences. Consequently, users did not want continuous control, but wanted to intervene precisely when the agent neared sensitive actions, when ambiguity increased, or when its confidence seemed unjustified. Behavioral micro-signals such as cursor shadowing, hesitations, and manual overrides can offer rich evidence of when oversight is desired and how users prefer to intervene [20, 125, 126].

To address these challenges, agents should become consequence-aware [98]. One desirable direction is for operating systems and browsers to expose lightweight metadata describing risk level, reversibility, or required confirmation. Such metadata would give agents a principled base for understanding consequences rather than inferring. However, deploying such infrastructure requires substantial coordination across OS vendors, browsers, and web developers, and faces practical constraints around adoption, consistency, and resource overhead. Meanwhile, agents should infer risk through model- and interaction-level cues: interface semantics (e.g., "Pay," "Delete"), workflow structure (e.g., multi-step checkouts), and uncertainty signals learned through interaction. These approaches raise familiar challenges, such as reward hacking [41], where an agent optimizes for appearing safe rather than being safe, but they offer feasible near-term paths for dynamic consequence estimation without standardized metadata.

A key opportunity from our findings (Section 5.2.1) is personalized risk tiering. Agents can infer user-specific boundaries from where users pause, undo, or retake control. These signals form an emergent human "risk model" that can guide autonomy levels. Low-risk actions (scrolling, reading) may run automatically; medium-risk actions (navigation among ambiguous elements, text entry) may require previews or rationales; and high-risk actions (submissions, payments, identity or outbound-message operations) should always require explicit confirmation. Guardrail policies ("never submit forms containing financial identifiers," "never message outside approved contexts") can encode user- or organization-specific constraints within these tiers. Treating risk as contextual and user-dependent rather than an intrinsic property of the interface enables calibrated autonomy and clear responsibility boundaries, and provides foundation for agents to decide when to pause, request confirmation, or transfer control.

7.3 Implication 3: Provide Situated Transparency into Agent Intent, Perception, and Action

Explainability research emphasizes calibrated trust [69, 134], but most interfaces for agents still focus on explaining outputs rather than making the agent's perceptual field, access boundaries, and intentions legible. Our findings in Section 5.1.2 show that users often did not know which window the agent was acting on or what information it processed. These uncertainties led users to oscillate between over-trust and hyper vigilance, and contributed to ambiguous mental models of agent capability and responsibility. Prior work established that accurate mental models are essential for effective human–agent collaboration [11, 125, 128], but the mechanisms for forming such models remain unclear when agents manipulate GUI environments on users' behalf. When perception, scope, and intent are opaque, users cannot tell whether an error stems from misinterpreting instructions or violating policies. These concerns intensify when agents hold long-term access to sensitive data.

Interfaces should therefore provide situational transparency and scaffolding based on user mental models. Environment maps can visualize the tabs, windows, and applications the agent can access and highlight which surface it currently perceives as active. Provenance ribbons can tie explanations or rationales to specific screenshots or UI regions, revealing what the agent has actually "seen." Capability cards can summarize per-task permissions and data accesses, and allow users to revise boundaries mid-run. Boundary notifications can indicate when an attempted action is blocked by organizational or user-defined policies. Context-specific uncertainty cues, such as acknowledging ambiguous buttons, CAPTCHA verification challenges [130], or multiple visually similar elements can help users decide when close supervision is warranted. Over time, such mechanisms, especially when paired with safe spaces to practice (Section 7.6), can help users form accurate expectations about agents' capability boundary, reducing both over-reliance and unnecessary anxiety.

7.4 Implication 4: Design Graphical Interface Infrastructures That Treat Agents as Users

Technical work which benchmarks GUI agents in human-oriented interfaces highlights model limitations [56, 72, 97, 106, 139]. Our findings (Section 4.2.2) show that many failures arise because interfaces are optimized exclusively for human manipulation. Transient banners, icon-only controls, CAPTCHA or QR-based verification, multi-step authorization flows, and visually dense dashboards routinely disrupted agents. Agents struggled with non-standard layouts, lock-screen timeouts, and elements that disappeared before screenshot-based perception completed. Participants frequently compensated by pre-arranging the interface, such as resizing windows, clearing pop-ups, and simplifying panels.

At the same time, agents increasingly employ proactive strategies such as identifying atypical UI components or simulating interface states [111, 113]. Yet these efforts face practical constraints when the underlying interface exposes little machine-readable structure, since agents must rely entirely on noisy visual signals, making learned policies less stable and harder to generalize across real-world UI variability. Many users therefore expressed a preference for more direct integrations, such as API-level access for common workflows, rather than relying entirely on screenshot-based interaction (Section 6.1.1). This friction suggests rethinking interface design from the perspective of "agent as a user" [125].

Treating agents as users requires a dual-channel interface design. Web platforms can expose agent-friendly APIs or embedded endpoints for tasks such as downloading statements, applying filters, or performing batch operations. Declarative task contracts can describe the states and constraints of workflows such as checkout or password changes, enabling agents to recognize "where they are" and "which steps require human-only confirmation" in a multi-step process. UI components can pair human-facing visuals with machine-readable identifiers, role tags, and risk annotations, reducing grounding ambiguity. When tasks require human-only actions, such as completing a CAPTCHA or providing biometric authentication, interfaces can provide explicit hooks prompting the agent to request assistance rather than becoming stuck in loops. As with accessibility standards, agent-aware design compatibility can formalize how applications expose risk and semantics during human-agent collaboration.

7.5 Implication 5: Treat Computational Cost and Resource Usage as Design Considerations

Token consumption, screenshot frequency, and latency are typically reported as benchmark metrics [17, 63]. Our findings extend to show how these system measures corresponds to user burden (Sections 4.3.4 and 5.1.1). Agents frequently operated far slower than manual use, with screenshot based perception and repeated retries inflating token usage and introducing delays (Section 4.3.3). Runaway loops could also exhaust resources while completing only part of a task. Crucially, users repurposed cost constraints such as token limits and timeouts as ad hoc safety measures to preempt errant behavior. Thus, resource usage functions not merely as an economic constraint, but as a proxy for risk management and system reliability.

Agents should therefore integrate cost awareness as a core interaction and safety concern. Mechanisms such as explicit cost budgets and predictive summaries enable users to anticipate resource implications and cap consumption. Cost diagnostics can surface where resources are being consumed, for example, by identifying repeated misgrounding or oscillation between similar UI states. Predictive cost summaries before execution can allow users to anticipate resource implications and compare agent use to manual completion. Agents should adopt resource-efficient strategies such as capturing smaller regions of the screen, caching stable interpretations, reducing tool invocation frequency, and lowering perception frequency in stable contexts. When consumption escalation is detected, agents should warn users and propose corrective actions, such as simplifying the plan or switching to an low-cost integration when available. Making cost and performance part of the design rather than a hidden metric can reduce unexpected burdens and support predictable, accountable use.

7.6 Implication 6: Institutionalize Ecological Sandboxes and User-Centered Safety Patterns

Unlike safety-critical domains that mandate bounded testing [2], current GUI agents typically demand immediate, broad access. Existing sandbox efforts, including emerging agent modes like ChatGPT Agent [76] and research prototypes like AirGapAgent [5], typically mirror the live interface but offer users limited control over what the agent can see, click, or modify. These mechanisms are coarse and system-level rather than configurable for specific tasks or contexts. As a result, users in our study built their own ecological sandboxes (Section 6.1.2) by relying on virtual machines, burner accounts, synthetic data, restricted devices, and close shadowing. Many adopted a "zero trust" posture toward agents [10, 20], viewing dedicated sandboxes as essential for protecting real assets [118].

Platforms can relieve this burden by institutionalizing ecological sandboxes that allow agents to rehearse tasks in controlled and configurable environments. Such spaces would mirror real interface states while letting users configure the agent's capability, specify what it may access, and surface common failure modes without real-world risks. They also support phased deployment, beginning with synthetic or mirrored data, progressing to restricted read only interaction, and granting full write access only after the agent demonstrates reliable progress.

Furthermore, the informal practices we observed, such as synthetic rehearsal, interface manipulation, explicit scoping, and data minimization, can be transformed into formalized safety patterns. One-click "safe workspace" modes can automatically create constrained file systems, restrict network domains, and launch agents inside sandboxed browser profiles. Guided scoping wizards can help users specify what the agent may access or modify, generating reusable templates that encode permissions and risk limits. Tools for on-the-fly data masking and surrogate accounts allow users to test workflows without risking sensitive information. By integrating these capabilities, platforms can replace burdensome user workarounds with a consistent support for safe agent operation.

8 Limitations and Future Work

Our findings are subject to limitations that outline critical directions for future research.

First, regarding scope and generalizability, our investigation primarily centered on GUI agents within web browsing tasks. While the identified issues are likely relevant to broad automation contexts, the specific manifestations of these complaints will inevitably vary based on agent architecture, interface modalities, and sociotechnical contexts. The current findings provide a taxonomy, but comparative studies across different agent platforms, task complexities, and user populations are needed to validate the transferability of these mitigation strategies across the broad ecosystem of AI-powered GUI automation.

Second, regarding methodology, this study integrates broader perspectives from social media analysis with interviews situated within Chinese, tech-savvy early adopters. While the social media analysis captures concerns from an international user base, the interview findings are inevitably influenced by China's sociotechnical landscape. For instance, the prevalence of 'super-apps', mobile-centric web designs, and stringent identity verification protocols (e.g., QR code logins) in this region introduces unique execution frictions that may differ from Western-centric desktop workflows. Furthermore, their reported mitigation (e.g., VM, isolation) reflected a level of technical expertise that may not generalize to broad populations. Future work should extend to less tech-savvy users to understand how they use GUI agents.

9 Conclusions

This paper presents a systematic empirical investigation into the user-reported complaints accompanying the proliferation of LLM-based GUI agents in web browsing. Through a two-phase study, we characterize the phenomena of user-reported complaints towards GUI agents, specifically deficits in grounding abstract intent into concrete interface affordances and the inability to adapt to dynamic visual states. We analyze the influences of these phenomena, demonstrating how they diverge from text-based hallucinations to manifest as irreversible, erroneous actions that compromise security through persistent authentication and uncontrolled file system access. Finally, our examination of mitigation strategies reveals that users are currently forced to rely on ad-hoc workarounds, such as ecological sandboxing and cursor shadowing, to scaffold brittle agent behaviors. We conclude that future GUI agents design should transcend simple autonomous execution to prioritize consequence-aware architectures that facilitate interaction-aligned intervention and appropriate human oversight.

References

[1] Saaket Agashe, Kyle Wong, Vincent Tu, Jiachen Yang, Ang Li, and Xin Eric Wang. 2025. Agent s2: A compositional generalist-specialist framework for computer use agents. *arXiv preprint arXiv:2504.00906* (2025).

[2] Damiano Angeletti, Enrico Giunchiglia, Massimo Narizzano, Alessandra Puddu, and Salvatore Sabina. 2010. Using bounded model checking for coverage analysis of safety-critical software in an industrial setting. *Journal of Automated Reasoning* 45, 4 (2010), 397–414.

[3] Anthropic. 2025. Computer use (beta). https://docs.anthropic.com/en/docs/build-with-claude/computer-use/ Accessed: 2025-02-19.

[4] Sumit Asthana, Jane Im, Zhe Chen, and Nikola Banovic. 2024. " I know even if you don't tell me": Understanding Users' Privacy Preferences Regarding AI-based Inferences of Sensitive Information for Personalization. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–21.

[5] Eugene Bagdasarian, Ren Yi, Sahra Ghalebikesabi, Peter Kairouz, Marco Gruteser, Sewoong Oh, Borja Balle, and Daniel Ramage. 2024. Airgapagent: Protecting privacy-conscious conversational agents. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*. 3868–3882.

[6] Michael Bailey, David Dittrich, Erin Kenneally, and Doug Maughan. 2012. The menlo report. *IEEE Security & Privacy* 10, 2 (2012), 71–75.

[7] Tom L Beauchamp et al. 2008. The belmont report. *The Oxford textbook of clinical research ethics* (2008), 149–155.

[8] Virginia Braun and Victoria Clarke. 2019. Reflecting on reflexive thematic analysis. *Qualitative research in sport, exercise and health* 11, 4 (2019), 589–597.

[9] Virginia Braun and Victoria Clarke. 2024. Thematic analysis. In *Encyclopedia of quality of life and well-being research*. Springer, 7187–7193.

[10] Jed R Brubaker, Casey Fiesler, Michael Madaio, John Tang, and Richmond Y Wong. 2024. Generative AI Going Awry: Enabling Designers to Proactively Avoid It in CSCW Applications. In *Companion Publication of the 2024 Conference on Computer-Supported Cooperative Work and Social Computing*. 125–127.

[11] John M Carroll. 2003. *Making use: scenario-based design of human-computer interactions*. MIT press.

[12] Ada Chen, Yongjiang Wu, Junyuan Zhang, Jingyu Xiao, Shu Yang, Jen tse Huang, Kun Wang, Wenxuan Wang, and Shuai Wang. 2025. A Survey on the Safety and Security Threats of Computer-Using Agents: JARVIS or Ultron? doi:10.48550/arXiv.2505.10924 arXiv:2505.10924 [cs.CL]

[13] Chaoran Chen, Leyang Li, Luke Cao, Yanfang Ye, Tianshi Li, Yaxing Yao, and Toby Jia-jun Li. 2025. Why am i seeing this: Democratizing end user auditing for online content recommendations. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology*. 1–23.

[14] Chaoran Chen, Zhiping Zhang, Bingcan Guo, Shang Ma, Ibrahim Khalilov, Simret A Gebreegziabher, Yanfang Ye, Ziang Xiao, Yaxing Yao, Tianshi Li, et al. 2025. The Obvious Invisible Threat: LLM-Powered GUI Agents' Vulnerability to Fine-Print Injections. *arXiv preprint arXiv:2504.11281* (2025).

[15] Chaoran Chen, Zhiping Zhang, Ibrahim Khalilov, Bingcan Guo, Simret A Gebreegziabher, Yanfang Ye, Ziang Xiao, Yaxing Yao, Tianshi Li, and Toby Jia-Jun Li. 2025. Toward a Human-Centered Evaluation Framework for Trustworthy LLM-powered GUI Agents. *HEAL workshop of CHI 25* (2025).

[16] Dongping Chen, Yue Huang, Siyuan Wu, Jingyu Tang, Huichi Zhou, Qihui Zhang, Zhigang He, Yilin Bai, Chujie Gao, Liuyi Chen, et al. 2024. GUI-World: A Video Benchmark and Dataset for Multimodal GUI-oriented Understanding. In *The Thirteenth International Conference on Learning Representations*.

[17] Gongwei Chen, Xurui Zhou, Rui Shao, Yibo Lyu, Kaiwen Zhou, Shuai Wang, Wentao Li, Yinchuan Li, Zhongang Qi, and Liqiang Nie. 2025. Less is more: Empowering gui agent with context-aware simplification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5901–5911.

[18] Weihao Chen, Xiaoyu Liu, Jiacheng Zhang, Ian Iong Lam, Zhicheng Huang, Rui Dong, Xinyu Wang, and Tianyi Zhang. 2023. Miwa: Mixed-initiative web automation for better user control and confidence. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–15.

[19] Wei-Hao Chen, Weixi Tong, Amanda Case, and Tianyi Zhang. 2025. Dango: A Mixed-Initiative Data Wrangling System using Large Language Model. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–28.

[20] Yu-Ting Chen, Hsin-Yi Sandy Tsai, and Chien Wen Yuan. 2024. Exploring How Users Attribute Responsibilities Across Different Stakeholders in Human-AI Interaction. In *Companion Publication of the 2024 Conference on Computer-Supported Cooperative Work and Social Computing*. 202–208.

[21] Zichen Chen, Jiaao Chen, Jianda Chen, and Misha Sra. 2025. Position: Standard Benchmarks Fail–LLM Agents Present Overlooked Risks for Financial Applications. *arXiv preprint arXiv:2502.15865* (2025).

[22] Kanzhi Cheng, Qiushi Sun, Yougang Chu, Fangzhi Xu, Yantao Li, Jianbing Zhang, and Zhiyong Wu. 2024. SeeClick: Harnessing GUI Grounding for Advanced Visual GUI Agents. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*. doi:10.18653/v1/2024.acl-long.505

[23] Anh-Hoang Dang, Vu Tran, and Le-Minh Nguyen. 2025. Survey and analysis of hallucinations in large language models: attribution to prompting strategies or model behavior. *Frontiers in Artificial Intelligence* 8 (2025), 1622292. doi:10.3389/frai.2025.1622292

[24] Antonella De Angeli, Sheryl Brahnam, Peter Wallis, and Alan Dix. 2006. Misuse and abuse of interactive technologies. In *CHI'06 Extended Abstracts on Human Factors in Computing Systems*. 1647–1650.

[25] Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabrizio Petroni, Antonio Scala, Guido Caldarelli, H. Eugene Stanley, and Walter Quattrociocchi. 2016. The spreading of misinformation online. *Proceedings of the National Academy of Sciences* 113, 3 (2016), 554–559.

[26] Zehang Deng, Yongjian Guo, Changzhou Han, Wanlun Ma, Junwu Xiong, Sheng Wen, and Yang Xiang. 2025. Ai agents under threat: A survey of key security challenges and future pathways. *Comput. Surveys* 57, 7 (2025), 1–36.

[27] Cedric Faas, Sophie Kerstan, Richard Uth, Markus Langer, and Anna Maria Feit. 2025. Design Considerations for Human Oversight of AI: Insights from

Co-Design Workshops and Work Design Theory. *arXiv preprint arXiv:2510.19512* (2025).

[28] Don Fallis. 2015. What is disinformation? *Library trends* 63, 3 (2015), 401–426.

[29] Xiaocong Fan, Sooyoung Oh, Michael McNeese, John Yen, Haydee Cuevas, Laura Strater, and Mica R. Endsley. 2008. The influence of agent reliability on trust in human-agent collaboration. In *Proceedings of the 15th European Conference on Cognitive Ergonomics: The Ergonomics of Cool Interaction* (Funchal, Portugal) *(ECCE '08)*. Association for Computing Machinery, New York, NY, USA, Article 7, 8 pages. doi:10.1145/1473018.1473028

[30] Haishuo Fang, Xiaodan Zhu, and Iryna Gurevych. 2025. Preemptive Detection and Correction of Misaligned Actions in LLM Agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Suzhou, China, 222–244. doi:10.18653/v1/2025.emnlp-main.12

[31] Haishuo Fang, Xiaodan Zhu, and Iryna Gurevych. 2025. Preemptive detection and correction of misaligned actions in llm agents. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*. 222–244.

[32] Kevin Feng, Tae Soo Kim, Rock Yuren Pang, Faria Huq, Tal August, and Amy X Zhang. 2025. On the Regulatory Potential of User Interfaces for AI Agent Governance. In *NeurIPS 2025 Workshop on Regulatable ML*.

[33] Casey Fiesler, Michael Zimmer, Nicholas Proferes, Sarah Gilbert, and Naiyan Jones. 2024. Remember the human: A systematic review of ethical considerations in reddit research. *Proceedings of the ACM on Human-Computer Interaction* 8, GROUP (2024), 1–33.

[34] Difei Gao, Lei Ji, Zechen Bai, Mingyu Ouyang, Peiran Li, Dongxing Mao, Qinchen Wu, Weichen Zhang, Peiyi Wang, Xiangwu Guo, et al. 2024. AssistGUI: Task-Oriented PC Graphical User Interface Automation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 13289–13298.

[35] Boyu Gou, Zanming Huang, Yuting Ning, Yu Gu, Michael Lin, Weijian Qi, Andrei Kopanev, Botao Yu, Bernal Jiménez Gutiérrez, Yiheng Shu, et al. 2025. Mind2Web 2: Evaluating Agentic Search with Agent-as-a-Judge. *arXiv preprint arXiv:2506.21506* (2025).

[36] Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. 2024. Navigating the Digital World as Humans Do: Universal Visual Grounding for GUI Agents. In *The Thirteenth International Conference on Learning Representations*.

[37] Nitesh Goyal, Minsuk Chang, and Michael Terry. 2024. Designing for Human-Agent Alignment: Understanding what humans want from their agents. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*. 1–6.

[38] Wei Hao, Van Tran, Vincent Rideout, Zixi Wang, AnMei Dasbach-Prisk, MH Afifi, Junfeng Yang, Ethan Katz-Bassett, Grant Ho, and Asaf Cidon. 2025. Do spammers dream of electric sheep? characterizing the prevalence of llm-generated malicious emails. In *Proceedings of the 2025 ACM Internet Measurement Conference*. 192–203.

[39] Hongliang He, Wenlin Yao, Kaixin Ma, Wenhao Yu, Yong Dai, Hongming Zhang, Zhenzhong Lan, and Dong Yu. 2024. WebVoyager: Building an End-to-End Web Agent with Large Multimodal Models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 6864–6890.

[40] Ming-Tung Hong, Jesse Josua Benjamin, and Claudia Müller-Birn. 2018. Coordinating agents: Promoting shared situational awareness in collaborative sensemaking. In *Companion of the 2018 ACM Conference on Computer Supported Cooperative Work and Social Computing*. 217–220.

[41] Tiechuan Hu, Wenbo Zhu, and Yuqi Yan. 2025. Reward Hacking in Reinforcement Learning and RLHF: A Multidisciplinary Examination of Vulnerabilities, Mitigation Strategies, and Alignment Challenges. In *2025 5th Intelligent Cybersecurity Conference (ICSC)*. IEEE, 272–275.

[42] Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh Tuan Luu, Junxian He, Pang Wei W Koh, and Bryan Hooi. 2024. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in llms. *Advances in Neural Information Processing Systems* 37 (2024), 24181–24215.

[43] Tian Huang, Chun Yu, Weinan Shi, Zijian Peng, David Yang, Weiqi Sun, and Yuanchun Shi. [n. d.]. Prompt2Task: Automating UI Tasks on Smartphones from Textual Prompts. *ACM Transactions on Computer-Human Interaction* ([n. d.]).

[44] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yvette Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of Hallucination in Natural Language Generation. *Comput. Surveys* (2023). https://arxiv.org/abs/2202.03629 arXiv:2202.03629.

[45] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *Comput. Surveys* 55, 12 (2023), 1–38.

[46] Haitao Jia, Ming He, Zimo Yin, Likang Wu, Jianping Fan, and Jitao Sang. 2025. ReInAgent: A Context-Aware GUI Agent Enabling Human-in-the-Loop Mobile Task Navigation. *arXiv preprint arXiv:2510.07988* (2025). doi:10.48550/arXiv.2510.07988

[47] Shian Jia, Xinbo Wang, Mingli Song, and Gang Chen. 2024. Agent Centric Operating System–a Comprehensive Review and Outlook for Operating System. *arXiv preprint arXiv:2411.17710* (2024).

[48] Wenjia Jiang, Yangyang Zhuang, Chenxi Song, Xu Yang, Joey Tianyi Zhou, and Chi Zhang. 2025. Appagentx: Evolving gui agents as proficient smartphone users. *arXiv preprint arXiv:2503.02268* (2025).

[49] Malte Josten and Torben Weis. 2025. Large Language Models as a Cyber Threat: Towards Countering LLM-based Spam Attacks. In *2025 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. IEEE Computer Society, 606–607.

[50] Samuel Judson, Matthew Elacqua, Filip Cano, Timos Antonopoulos, Bettina Könighofer, Scott J Shapiro, and Ruzica Piskac. 2024. soid: A Tool for Legal Accountability for Automated Decision Making. In *International Conference on Computer Aided Verification*. Springer, 233–246.

[51] Hyunggu Jung, Woosuk Seo, Seokwoo Song, and Sungmin Na. 2023. Toward value scenario generation through large language models. In *Companion Publication of the 2023 Conference on Computer Supported Cooperative Work and Social Computing*. 212–220.

[52] Jacek A Kopec and John M Esdaile. 1990. Bias in case-control studies. A review. *Journal of epidemiology and community health* 44, 3 (1990), 179.

[53] Thomas Kuntz, Agatha Duzan, Hao Zhao, Francesco Croce, J. Zico Kolter, Nicolas Flammarion, and Maksym Andriushchenko. 2025. OS-Harm: A Benchmark for Measuring Safety of Computer Use Agents. doi:10.48550/arXiv.2506.14866 arXiv:2506.14866 [cs.SE]

[54] Jungjae Lee, Dongjae Lee, Chihun Choi, Youngmin Im, Jaeyoung Wi, Kihong Heo, Sangeun Oh, Sunjae Lee, and Insik Shin. 2025. Safeguarding mobile gui agent via logic-based action verification. *arXiv preprint arXiv:2503.18492* (2025).

[55] Kaixin Li, Ziyang Meng, Hongzhan Lin, Ziyang Luo, Yuchen Tian, Jing Ma, Zhiyong Huang, and Tat-Seng Chua. 2025. ScreenSpot-Pro: GUI Grounding for Professional High-Resolution Computer Use. In *Workshop on Reasoning and Planning for Large Language Models*. https://openreview.net/forum?id=XaKNDIAHas

[56] Wei Li, William E Bishop, Alice Li, Christopher Rawles, Folawiyo Campbell-Ajala, Divya Tyamagundlu, and Oriana Riva. 2024. On the effects of data scale on ui control agents. *Advances in Neural Information Processing Systems* 37 (2024), 92130–92154.

[57] Youwei Li, Yangyang Li, and Yangzhao Yang. 2024. Test-Agent: A Multimodal App Automation Testing Framework Based on the Large Language Model. In *2024 IEEE 4th International Conference on Digital Twins and Parallel Intelligence (DTPI)*. IEEE, 609–614.

[58] Yuanchun Li, Hao Wen, Weijun Wang, Xiangyu Li, Yizhen Yuan, Guohong Liu, Jiacheng Liu, Wenxing Xu, Xiang Wang, Yi Sun, et al. 2024. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv preprint arXiv:2401.05459* (2024).

[59] Siyuan Liang, Tianmeng Fang, Zhe Liu, Aishan Liu, Yan Xiao, Jinyuan He, Ee-Chien Chang, and Xiaochun Cao. 2025. SafeMobile: Chain-level Jailbreak Detection and Automated Evaluation for Multimodal Mobile Agents. *arXiv preprint arXiv:2507.00841* (2025).

[60] Zeyi Liao, Lingbo Mo, Chejian Xu, Mintong Kang, Jiawei Zhang, Chaowei Xiao, Yuan Tian, Bo Li, and Huan Sun. 2024. EIA: Environmental injection attack on generalist web agents for privacy leakage. In *The Thirteenth International Conference on Learning Representations*.

[61] Zhixin Lin, Jungang Li, Shidong Pan, Yibo Shi, Yue Yao, and Dongliang Xu. 2025. Mind the third eye! benchmarking privacy awareness in mllm-powered smartphone agents. *arXiv preprint arXiv:2508.19493* (2025).

[62] Guangyi Liu, Pengxiang Zhao, Liang Liu, Zhiming Chen, Yuxiang Chai, Shuai Ren, Hao Wang, Shibo He, and Wenchao Meng. 2025. Learnact: Few-shot mobile gui agent with a unified demonstration benchmark. *arXiv preprint arXiv:2504.13805* (2025).

[63] Zikang Liu, Junyi Li, Wayne Xin Zhao, Dawei Gao, Yaliang Li, and Ji-rong Wen. 2025. PAL-UI: Planning with Active Look-back for Vision-Based GUI Agents. *arXiv preprint arXiv:2510.00413* (2025).

[64] Xing Han Lù, Gaurav Kamath, Marius Mosbach, and Siva Reddy. 2025. Build the web for agents, not agents for the web. *arXiv preprint arXiv:2506.10953* (2025).

[65] Kai Lukoff, Ulrik Lyngs, Himanshu Zade, J Vera Liao, James Choi, Kaiyue Fan, Sean A Munson, and Alexis Hiniker. 2021. How the design of youtube influences user sense of agency. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–17.

[66] Rongjun Ma, Caterina Maidhof, Juan Carlos Carrillo, Janne Lindqvist, and Jose Such. 2025. Privacy perceptions of custom gpts by users and creators. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–18.

[67] Diogo Marques, Ildar Muslukhov, Tiago Guerreiro, Luís Carriço, and Konstantin Beznosov. 2016. Snooping on mobile phones: Prevalence and trends. In *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. 159–174.

[68] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. 2019. Reliability and inter-rater reliability in qualitative research: Norms and guidelines for CSCW and HCI practice. *Proceedings of the ACM on human-computer interaction* 3, CSCW (2019), 1–23.

[69] Siddharth Mehrotra, Carolina Centeio Jorge, Catholijn M Jonker, and Myrthe L Tielman. 2024. Integrity-based explanations for fostering appropriate trust in AI agents. *ACM Transactions on Interactive Intelligent Systems* 14, 1 (2024), 1–36.

[70] Ildar Muslukhov, Yazan Boshmaf, Cynthia Kuo, Jonathan Lester, and Konstantin Beznosov. 2013. Know your enemy: the risk of unauthorized access in smartphones by insiders. In *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services.* 271–280.

[71] Dang Nguyen, Jian Chen, Yu Wang, Gang Wu, Namyong Park, Zhengmian Hu, Hanjia Lyu, Junda Wu, Ryan Aponte, Yu Xia, et al. 2024. Gui agents: A survey. *arXiv preprint arXiv:2412.13501* (2024).

[72] Dang Nguyen, Jian Chen, Yu Wang, Gang Wu, Namyong Park, Zhengmian Hu, Hanjia Lyu, Junda Wu, Ryan Aponte, Yu Xia, et al. 2025. Gui agents: A survey. In *Findings of the Association for Computational Linguistics: ACL 2025.* 22522–22538.

[73] Liangbo Ning, Ziran Liang, Zhuohang Jiang, Haohao Qu, Yujuan Ding, Wenqi Fan, Xiao-yong Wei, Shanru Lin, Hui Liu, Philip S Yu, et al. 2025. A survey of webagents: Towards next-generation ai agents for web automation with large foundation models. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2.* 6140–6150.

[74] Michelle OâÂ Â^{2}Reilly and Nikki Kiyimba. 2015. Advanced qualitative research: A guide to using theory. (2015).

[75] Virpi Oksman. 2010. *The mobile phone-A medium in itself.* VTT.

[76] OpenAI. 2024. Introducing ChatGPT Agents. https://openai.com/index/introducing-chatgpt-agent/. Accessed: 2025-01-29.

[77] OpenAI. 2025. Introducing Operator. https://openai.com/index/introducing-operator/ Accessed: 2025-02-19.

[78] Michelle O'Reilly, Nikki Kiyimba, and Alison Drewett. 2021. Mixing qualitative methods versus methodologies: A critical reflection on communication and power in inpatient care. *Counselling and psychotherapy research* 21, 1 (2021), 66–76.

[79] Vardaan Pahuja, Yadong Lu, Corby Rosset, Boyu Gou, Arindam Mitra, Spencer Whitehead, Yu Su, and Ahmed Hassan. 2025. Explorer: Scaling exploration-driven web trajectory synthesis for multimodal web agents. In *Findings of the Association for Computational Linguistics: ACL 2025.* 6300–6323.

[80] Alexander Pan, Erik Jones, Meena Jagadeesan, and Jacob Steinhardt. 2024. Feedback loops with language models drive in-context reward hacking. In *Proceedings of the 41st International Conference on Machine Learning.* 39154–39200.

[81] Ayush Pandey, Jai Bardhan, Ishita Jain, Ramya S Hebbalaguppe, Rohan Raju Dhanakshirur, and Lovekesh Vig. 2025. Refine and Align: Confidence Calibration through Multi-Agent Interaction in VQA. *arXiv preprint arXiv:2511.11169* (2025).

[82] Rock Yuren Pang, Hope Schroeder, Kynnedy Simone Smith, Solon Barocas, Ziang Xiao, Emily Tseng, and Danielle Bragg. 2025. Understanding the LLM-ification of CHI: Unpacking the Impact of LLMs at CHI through a Systematic Literature Review. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems.* 1–20.

[83] Georgios Papoudakis, Thomas Coste, Jun Wang, Kun Shao, et al. 2025. Succeed or Learn Slowly: Sample Efficient Off-Policy Reinforcement Learning for Mobile App Control. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems.*

[84] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. In *Proceedings of the 36th annual acm symposium on user interface software and technology.* 1–22.

[85] Samir Passi. 2025. Agentic AI has a Human Oversight Problem. *Available at SSRN 5529058* (2025).

[86] Nixalkumar Patel and Heta Chauhan. 2025. Agentic AI and the Future of Work: Transforming Labor Markets, Economic Structures, and Workforce Development. In *The Power of Agentic AI: Redefining Human Life and Decision-Making: In Industry 6.0.* Springer, 205–227.

[87] Nicholas Proferes, Naiyan Jones, Sarah Gilbert, Casey Fiesler, and Michael Zimmer. 2021. Studying reddit: A systematic overview of disciplines, approaches, methods, and ethics. *Social Media+ Society* 7, 2 (2021), 20563051211019004.

[88] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, et al. 2024. Chatdev: Communicative agents for software development. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 15174–15186.

[89] Cheng Qian, Zuxin Liu, Akshara Prabhakar, Zhiwei Liu, Jianguo Zhang, Haolin Chen, Heng Ji, Weiran Yao, Shelby Heinecke, Silvio Savarese, Caiming Xiong, and Huan Wang. 2025. UserBench: An Interactive Gym Environment for User-Centric Agents. *arXiv preprint arXiv:2507.22034* (2025). doi:10.48550/arXiv.2507.22034

[90] Subhey Sadi Rahman, Md. Adnanul Islam, Md. Mahbub Alam, Musarrat Zeba, Md. Abdur Rahman, Sadia Sultana Chowa, Mohaimenul Azam Khan Raiaan, and Sami Azam. 2025. Hallucination to Truth: A Review of Fact-Checking and Factuality Evaluation in Large Language Models. (2025). doi:10.48550/arXiv.2508.03860 arXiv:2508.03860.

[91] Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William E Bishop, Wei Li, Folawiyo Campbell-Ajala, et al. 2024. AndroidWorld: A Dynamic Benchmarking Environment for Autonomous Agents. In *The Thirteenth International Conference on Learning Representations.*

[92] Christopher Rawles, Sarah Clinckemaillie, Yifan Chang, Jonathan Waltz, Gabrielle Lau, Marybeth Fair, Alice Li, William E Bishop, Wei Li, Folawiyo Campbell-Ajala, et al. 2024. AndroidWorld: A Dynamic Benchmarking Environment for Autonomous Agents. In *The Thirteenth International Conference on Learning Representations.*

[93] Summer Rebensky, Kendall Carmody, Cherrise Ficke, Daniel Nguyen, Meredith Carroll, Jessica Wildman, and Amanda Thayer. 2021. Whoops! Something went wrong: Errors, trust, and trust repair strategies in human agent teaming. In *International Conference on Human-Computer Interaction.* Springer, 95–106.

[94] Zeyu Rong, Tianxi Ji, Jiazhao Zhang, Tong Qu, Jingling Li, and Qiang Ma. 2025. RecAgent: Uncertainty-Aware GUI Agent. *arXiv preprint arXiv:2508.04025* (2025). doi:10.48550/arXiv.2508.04025

[95] Yangjun Ruan, Honghua Dong, Andrew Wang, Silviu Pitis, Yongchao Zhou, Jimmy Ba, Yann Dubois, Chris J Maddison, and Tatsunori Hashimoto. 2023. Identifying the risks of lm agents with an lm-emulated sandbox. *arXiv preprint arXiv:2309.15817* (2023).

[96] Mark Russinovich, Ahmed Salem, and Ronen Eldan. 2025. Great, now write an article about that: The crescendo {Multi-Turn}{LLM} jailbreak attack. In *34th USENIX Security Symposium (USENIX Security 25).* 2421–2440.

[97] Huawen Shen, Chang Liu, Gengluo Li, Xinlong Wang, Yu Zhou, Can Ma, and Xiangyang Ji. 2024. Falcon-ui: Understanding gui before following user instructions. *arXiv preprint arXiv:2412.09362* (2024).

[98] Yucheng Shi, Wenhao Yu, Wenlin Yao, Wenhu Chen, and Ninghao Liu. 2025. Towards trustworthy gui agents: A survey. *arXiv preprint arXiv:2503.23434* (2025).

[99] Ben Shneiderman. 2002. Promoting universal usability with multi-layer interface design. *ACM SIGCAPH computers and the physically handicapped* 73-74 (2002), 1–8.

[100] Junhao Su, Yuanliang Wan, Junwei Yang, Hengyu Shi, Tianyang Han, Junfeng Luo, and Yurui Qiu. 2025. Failure Makes the Agent Stronger: Enhancing Accuracy through Structured Reflection for Reliable Tool Interactions. *arXiv preprint arXiv:2509.18847* (2025).

[101] Qiushi Sun, Mukai Li, Zhoumianze Liu, Zhihui Xie, Fangzhi Xu, Zhangyue Yin, Kanzhi Cheng, Zehao Li, Zichen Ding, Qi Liu, et al. 2025. OS-Sentinel: Towards Safety-Enhanced Mobile GUI Agents via Hybrid Validation in Realistic Workflows. *arXiv preprint arXiv:2510.24411* (2025).

[102] Xiangru Tang, Qiao Jin, Kunlun Zhu, Tongxin Yuan, Yichi Zhang, Wangchunshu Zhou, Meng Qu, Yilun Zhao, Jian Tang, Zhuosheng Zhang, et al. 2024. Prioritizing safeguarding over autonomy: Risks of llm agents for science. *arXiv preprint arXiv:2402.04247* (2024).

[103] Xingjian Tao, Yiwei Wang, Yujun Cai, Zhicheng Yang, and Jing Tang. 2025. Understanding GUI Agent Localization Biases through Logit Sharpness. In *Findings of the Association for Computational Linguistics: EMNLP 2025.* Association for Computational Linguistics, Suzhou, China, 23361–23374. doi:10.18653/v1/2025.findings-emnlp.1268

[104] Jonathan A Tran, Katie S Yang, Katie Davis, and Alexis Hiniker. 2019. Modeling the engagement-disengagement cycle of compulsive phone use. In *Proceedings of the 2019 CHI conference on human factors in computing systems.* 1–14.

[105] Jessica Vitak, Michael Zimmer, Anna Lenhart, Sunyup Park, Richmond Y. Wong, and Yaxing Yao. 2021. Designing for data awareness: addressing privacy and security concerns about "smart" technologies. In *Companion Publication of the 2021 Conference on Computer Supported Cooperative Work and Social Computing.* 364–367.

[106] Shuai Wang, Weiwen Liu, Jingxuan Chen, Yuqi Zhou, Weinan Gan, Xingshan Zeng, Yuhan Che, Shuai Yu, Xinlong Hao, Kun Shao, et al. 2024. Gui agents with foundation models: A comprehensive survey. *arXiv preprint arXiv:2411.04890* (2024).

[107] Alexander Wei, Nika Haghtalab, and Jacob Steinhardt. 2023. Jailbroken: How does llm safety training fail? *Advances in Neural Information Processing Systems* 36 (2023), 80079–80110.

[108] Marcus Williams, Micah Carroll, Adhyyan Narang, Constantin Weisser, Brendan Murphy, and Anca Dragan. 2025. On Targeted Manipulation and Deception when Optimizing LLMs for User Feedback. In *The Thirteenth International Conference on Learning Representations.*

[109] Yuhao Wu, Franziska Roesner, Tadayoshi Kohno, Ning Zhang, and Umar Iqbal. 2025. IsolateGPT: An Execution Isolation Architecture for LLM-Based Agentic Systems. In *NDSS.*

[110] Zhe Wu, Hongjin Lu, Junliang Xing, Changhao Zhang, Yin Zhu, Yuhao Yang, Yuheng Jing, Kai Li, Kun Shao, Jianye Hao, et al. 2025. Hi-Agent: Hierarchical Vision-Language Agents for Mobile Device Control. *arXiv preprint arXiv:2510.14388* (2025).

[111] Jiannan Xiang, Yun Zhu, Lei Shu, Maria Wang, Lijun Yu, Gabriel Barcik, James Lyon, Srinivas Sunkara, and Jindong Chen. 2025. UISim: An Interactive

Image-Based UI Simulator for Dynamic Mobile Environments. *arXiv preprint arXiv:2509.21733* (2025).

[112] Hongwei Xiao, Yongqi Sun, Zhenghao Duan, Yunxiang Huo, Jingze Liu, Mingyu Luo, Yanhui Li, and Yingchao Zhang. 2024. A Study of Model Iterations of Fitts' Law and Its Application to Human–Computer Interactions. *Applied Sciences* 14, 16 (2024), 7386. doi:10.3390/app14167386

[113] Bin Xie, Rui Shao, Gongwei Chen, Kaiwen Zhou, Yinchuan Li, Jie Liu, Min Zhang, and Liqiang Nie. 2025. Gui-explorer: Autonomous exploration and mining of transition-aware knowledge for gui agent. *arXiv preprint arXiv:2505.16827* (2025).

[114] Tianci Xue, Weijian Qi, Tianneng Shi, Chan Hee Song, Boyu Gou, Dawn Song, Huan Sun, and Yu Su. 2025. An illusion of progress? assessing the current state of web agents. *arXiv preprint arXiv:2504.01382* (2025).

[115] Jingyi Yang, Shuai Shao, Dongrui Liu, and Jing Shao. 2025. RiOSWorld: Benchmarking the Risk of Multimodal Computer-Use Agents. arXiv:2506.00618 [cs.AI] https://arxiv.org/abs/2506.00618

[116] Jingqi Yang, Zhilong Song, Jiawei Chen, Mingli Song, Sheng Zhou, Xiaogang Ouyang, Chun Chen, Can Wang, et al. 2025. GUI-Robust: A Comprehensive Dataset for Testing GUI Agent Robustness in Real-World Anomalies. *arXiv preprint arXiv:2506.14477* (2025).

[117] Pei Yang, Hai Ci, and Mike Zheng Shou. 2025. macOSWorld: A Multilingual Interactive Benchmark for GUI Agents. *arXiv preprint arXiv:2506.04135* (2025).

[118] Jingzhou Ye, Yao Li, Wenting Zou, and Xueqiang Wang. 2025. From Awareness to Action: The Effects of Experiential Learning on Educating Users about Dark Patterns. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–22.

[119] Ori Yoran, Samuel Amouyal, Chaitanya Malaviya, Ben Bogin, Ofir Press, and Jonathan Berant. 2024. AssistantBench: Can Web Agents Solve Realistic and Time-Consuming Tasks?. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. 8938–8968.

[120] Chaoyun Zhang, Shilin He, Liqun Li, Si Qin, Yu Kang, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. 2025. Api agents vs. gui agents: Divergence and convergence. *arXiv preprint arXiv:2503.11069* (2025).

[121] Chaoyun Zhang, Shilin He, Jiaxu Qian, Bowen Li, Liqun Li, Si Qin, Yu Kang, Minghua Ma, Guyue Liu, Qingwei Lin, et al. 2024. Large language model-brained gui agents: A survey. *arXiv preprint arXiv:2411.18279* (2024).

[122] Chi Zhang, Zhao Yang, Jiaxuan Liu, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. 2023. AppAgent: Multimodal Agents as Smartphone Users. arXiv:2312.13771 [cs.CV] https://arxiv.org/abs/2312.13771

[123] Chi Zhang, Zhao Yang, Jiaxuan Liu, Yanda Li, Yucheng Han, Xin Chen, Zebiao Huang, Bin Fu, and Gang Yu. 2025. Appagent: Multimodal agents as smartphone users. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–20.

[124] Kaiyuan Zhang, Zian Su, Pin-Yu Chen, Elisa Bertino, Xiangyu Zhang, and Ninghui Li. 2025. LLM Agents Should Employ Security Principles. *arXiv preprint arXiv:2505.24019* (2025).

[125] Mingyuan Zhang, Zhaolin Cheng, Sheung Ting Ramona Shiu, Jiacheng Liang, Cong Fang, Zhengtao Ma, Le Fang, and Stephen Jia Wang. 2023. Towards Human-Centred AI-Co-Creation: A Three-Level Framework for Effective Collaboration between Human and AI. In *Companion Publication of the 2023 Conference on Computer Supported Cooperative Work and Social Computing*. 312–316.

[126] Muru Zhang, Ofir Press, William Merrill, Alisa Liu, and Noah A Smith. 2023. How language model hallucinations can snowball. *arXiv preprint arXiv:2305.13534* (2023).

[127] Shuning Zhang, Ying Ma, Jingruo Chen, Simin Li, Xin Yi, and Hewu Li. 2025. Towards Aligning Personalized AI Agents with Users' Privacy Preference. In *Proceedings of the 2025 Workshop on Human-Centered AI Privacy and Security*. 33–42.

[128] Shuning Zhang, Hui Wang, and Xin Yi. 2025. Exploring collaboration patterns and strategies in human-ai co-creation through the lens of agency: A scoping review of the top-tier hci literature. *Proceedings of the ACM on Human-Computer Interaction* 9, 7 (2025), 1–43.

[129] Shuning Zhang, Lyumanshan Ye, Xin Yi, Jingyu Tang, Bo Shui, Haobin Xing, Pengfei Liu, and Hewu Li. 2024. " Ghost of the past": identifying and resolving privacy leakage from LLM's memory through proactive user interaction. *arXiv preprint arXiv:2410.14931* (2024).

[130] Shuning Zhang, Xin Yi, Shixuan Li, Haobin Xing, and Hewu Li. 2025. Priv-CAPTCHA: Interactive CAPTCHA to Facilitate Effective Comprehension of APP Privacy Policy. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–20.

[131] Shuning Zhang, Xin Yi, Haobin Xing, Lyumanshan Ye, Yongquan Hu, and Hewu Li. 2024. Adanonymizer: Interactively Navigating and Balancing the Duality of Privacy and Output Performance in Human-LLM Interaction. *arXiv preprint arXiv:2410.15044* (2024).

[132] Wan Zhang and Jing Zhang. 2025. Hallucination Mitigation for Retrieval-Augmented Large Language Models: A Review. *Mathematics* 13, 5 (2025), 856. doi:10.3390/math13050856

[133] Xuxin Zhang, Shuchang Liu, Shuai Wang, Cunxiang Wang, Qipeng Guo, Zhiyong Wu, Deyi Xiong, and Yue Zhang. 2024. Ask-Before-Plan: Proactive Language Agents for Real-World Planning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics, Miami, Florida, USA, 10836–10863. doi:10.18653/v1/2024.findings-emnlp.636

[134] Yunfeng Zhang, Q Vera Liao, and Rachel KE Bellamy. 2020. Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 295–305.

[135] Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2024. A survey on the memory mechanism of large language model based agents. *arXiv preprint arXiv:2404.13501* (2024).

[136] Zhisong Zhang, Tianqing Fang, Kaixin Ma, Wenhao Yu, Hongming Zhang, Haitao Mi, and Dong Yu. 2025. Enhancing web agents with explicit rollback mechanisms. *arXiv preprint arXiv:2504.11788* (2025).

[137] Zhiping Zhang, Michelle Jia, Hao-Ping Lee, Bingsheng Yao, Sauvik Das, Ada Lerner, Dakuo Wang, and Tianshi Li. 2024. "It's a Fair Game", or Is It? Examining How Users Navigate Disclosure Risks and Benefits When Using LLM-Based Conversational Agents. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–26.

[138] Zhuohao (Jerry) Zhang, Eldon Schoop, Jeffrey Nichols, Anuj Mahajan, and Amanda Swearngin. 2025. From Interaction to Impact: Towards Safer AI Agent Through Understanding and Evaluating Mobile UI Operation Impacts. In *Proceedings of the 30th International Conference on Intelligent User Interfaces (IUI '25)*. Association for Computing Machinery, New York, NY, USA, 727–744. doi:10.1145/3708359.3712153

[139] Kangjia Zhao, Jiahui Song, Leigang Sha, Haozhan Shen, Zhi Chen, Tiancheng Zhao, Xiubo Liang, and Jianwei Yin. 2024. Gui testing arena: A unified benchmark for advancing autonomous gui testing agent. *arXiv preprint arXiv:2412.18426* (2024).

[140] Boyuan Zheng, Boyu Gou, Jihyung Kil, Huan Sun, and Yu Su. 2024. GPT-4V (ision) is a Generalist Web Agent, if Grounded. In *International Conference on Machine Learning*. PMLR, 61349–61385.

[141] Shuyan Zhou, Frank F Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, et al. 2023. Webarena: A realistic web environment for building autonomous agents. *arXiv preprint arXiv:2307.13854* (2023).

[142] Yuqi Zhu, Shuofei Qiao, Yixin Ou, Shumin Deng, Shiwei Lyu, Yue Shen, Lei Liang, Jinjie Gu, Huajun Chen, and Ningyu Zhang. 2025. Knowagent: Knowledge-augmented planning for llm-based agents. In *Findings of the Association for Computational Linguistics: NAACL 2025*. 3709–3732.

A Details of Social Media Analysis

Our social media analysis covered posts from different subreddits, with 44 from r/OpenAI, 42 from r/ClaudeAI, 21 from r/singularity, 18 from r/LocalLLaMA, 14 from r/AI_Agents, 8 from r/ArtificialIntelligence, 6 from r/ChatGPT, 5 from r/ChatGPTCoding, 5 from r/Anthropic, and others from other subreddits.

B Semi-structured Interview Protocol

This protocol is designed for semi-structured interviews, allowing for both predefined questions and follow-up questions based on participant responses. Examples are only used when participants want further clarification on the questions.

B.1 Section 1: Basic Information

1. Could you describe your first experience using a GUI Agent, or your most memorable experience with a GUI Agent? What prompted you to start using GUI Agents?

2. In what situations do you typically use GUI Agents?

3. Could you describe a recent task you completed using a GUI Agent in detail?

4. Have you ever used GUI Agents for browser automation tasks? If so, could you describe one of your most recent experiences?

5. What different GUI Agents have you used? How would you compare them in terms of the following factors: functionality, usability, privacy and security concerns, and efficiency?

B.2 Section 2: Unexpected Behaviors and Impacts

6. Have you ever encountered any unexpected behavior from a GUI Agent? If yes, could you describe the most memorable instance? (e.g., did the Agent perform tasks you did not request, or execute tasks in an unexpected way?)

7. Have you ever experienced a situation where a GUI Agent was unable to complete a task? If yes, could you describe the most memorable instance in detail?

8. Have you ever received incorrect or misleading information from a GUI Agent? If yes, could you provide some specific examples?

9. Have you encountered any privacy or security issues related to GUI Agents? (e.g., did an Agent access your personal information without authorization, or did its behavior result in data leakage?)

10. Which of these unexpected behaviors had the most significant impact on you? What were the consequences? (e.g., time loss, financial loss, privacy breaches, security risks, decreased user experience)

11. Did you later discover the reason for the unexpected behavior of the GUI Agent? Or do you still not know the cause? If you discovered it, what was the reason, and how did you find out? Was it due to configuration errors, limited model capabilities, insufficient context understanding, or system complexity?

B.3 Section 3: User Coping Strategies

12. What measures have you taken to prevent or respond to these unexpected behaviors when using GUI Agents? (e.g., limiting Agent permissions, manually checking Agent operations, using virtual machines or containers to run Agents)

13. Do you think your coping strategies were effective? If yes, to what extent?

14. What problems do you still encounter after the strategies you adopted?

15. Do you know why these strategies were ineffective or why problems persisted? If not, do you have any guesses?

B.4 Section 4: User Expectations for Developers

16. What functions or features do you think GUI Agents should offer to reduce unexpected behaviors?

17. How could they help you better control the Agent's actions and prevent unexpected behaviors?

B.5 Section 5: Future Outlook

18. What do you think is the future trend of GUI Agents?

19. How do you think GUI Agents will change our work and lifestyle?

20. What are your expectations for the future development of GUI Agents? What new functions would you like GUI Agents to achieve?

21. Is there anything else you would like to add?