The Perils of Agency: How Developers Perceive, Prioritize, and Address Risks in Agentic AI Products

Hao-Ping (Hank) Lee[1*], **Jessica He**[2], **David Piorkowski**[3*], **Thomas Serban von Davier**[1], **Jodi Forlizzi**[1], **Sauvik Das**[1]

[1]Carnegie Mellon University
[2]IBM Research
[3]Alinia

Abstract

Agentic AI systems act autonomously, use tools, adapt to context, and operate in complex real-world environments. However, these same characteristics can create or exacerbate product risks. We studied how industry developers (n=35) perceive, prioritize, and address the risks in their agentic AI products. We found that developers' perceptions of risk were closely tied to the qualities that made the product agentic, such as autonomy, tool use, and usage in a real-world context. Developers prioritized product and business risks before considering downstream societal risks like job displacement and end-user privacy. This prioritization also impacted developers' ability and motivation to mitigate agentic risks. Finally, developers lacked mature controls for containing agentic risks, often relying on constraining the same characteristics that make agents useful: e.g., autonomy and goal complexity. These findings reveal a capability vs. risk control tension in agentic AI development: developers need to address risks that emerge from agentic capabilities, yet they currently have limited support for doing so without constraining agentic functionality.

Introduction

Across academia, industry, and government, AI agents are framed as a new paradigm for software systems and technology governance because they can act autonomously with limited oversight, interact with and affect their environments, and operate across diverse roles, contexts, and tasks (Kasirzadeh and Gabriel 2025; Chan et al. 2023). These same capabilities also create new and exacerbate known risks of generative AI systems: e.g., agents can violate policies or take unauthorized actions (Hart 2026a), misuse tools or environment resources that leak sensitive information or damage connected systems (Roth 2026), and make actions harder to monitor and hold accountable (Hart 2026b).

Despite the growing excitement and concern around agentic AI, we know little about how developers who build user-facing agentic AI products perceive these emergent risks, decide which risks to prioritize, and implement controls to mitigate these risks. Understanding these points is essential because while emerging governance frameworks for agentic AI call for centering principles like meaningful human control, accountability, and monitoring (Infocomm Media Development Authority 2026; World Economic Forum and Capgemini 2025), prior work has documented a substantial "gap between principle and practice" in developing human-centered and responsible AI (Winfield and Jirotka 2018; Shneiderman 2020).

To understand how developers perceive, prioritize, and mitigate agentic AI risks, we draw on the Security and Privacy Acceptance Framework (SPAF). SPAF was developed to explain end-users' adoption of security and privacy best practices (Das et al. 2022), and recent work has adapted its three barriers—*awareness* of harms, *motivation* to act, and *ability* to address the harms—to study how *practitioners* incorporate human-centered principles into AI product development (Lee et al. 2024a). We use SPAF in a similar, developer-oriented way in the context of agentic AI product development. Specifically, we ask three research questions:

RQ1 To what extent does agentic AI shape developers' perceived risks of their products? (Awareness)

RQ2 How do developers prioritize addressing emergent agentic AI risks of their products? (Motivation)

RQ3 What constitutes addressing agentic AI risks for developers, and what affects their ability to do so? (Ability)

To answer these questions, we conducted semi-structured interviews with $N = 35$ developers from different product teams at IBM. All participants had engaged in addressing agentic AI risks for their user-facing agentic AI products in some capacity. In line with the recent agentic AI literature, we define agentic AI products as products that employ foundation models (e.g., LLMs or large multimodal models) with access to tools (e.g., APIs, external services, computational resources), and have *autonomy* to *act in an environment* based on set goals, *decide which actions to perform*, and *execute multi-step actions to pursue goals* for end users (Pan et al. 2026; Kasirzadeh and Gabriel 2025; Wang et al. 2024a; Acharya, Kuppan, and Divya 2025; Miehling et al. 2025).

We found that developers focused more on risks that were directly proximate to product objectives for their AI agents (e.g., risks related to reliability and performance) and less on broader downstream risks (e.g., risks related to job displacement or privacy). Developers were particularly aware of risks related to agent policy violations, compounding

[*]Work done while at IBM Research.

failures, or unclear action accountability (**RQ1**). Developers prioritized mitigating risks through an implicit model of business success: risks became more actionable when they threatened agent performance, user adoption, production safety, or client needs. Conversely, developers were more hesitant about mitigations that introduced opportunity costs, reduced product value, conflicted with development timelines, or fell outside of their perceived responsibilities (**RQ2**). Developers implemented a range of mitigation strategies and controls, such as human oversight, action constraints, access control, validation, and data sanitization. Yet these controls often constrained the same properties that made the systems agentic in the first place: autonomy, adaptability, goal complexity, and environment complexity. Developers also lacked reliable assessment methods to evaluate whether these mitigations were effective (**RQ3**). In sum, our findings reveal a central tension in agentic AI product development: industry efforts to make products *more agentic* create new risks, which developers then attempt to mitigate by making products *less agentic*.

Our work makes the following contributions:

- We provide an empirical account of how developers perceive, prioritize, and mitigate risks entailed by agentic AI products.
- We show how developers' *risk awareness* is shaped by product-proximate agentic AI functionality, what *motivates and inhibits* risk mitigation work, and what *affects developers' ability* to implement risk controls.
- We extend the human-centered AI (HAI) literature on the principle-practice gap by showing how this gap manifests in agentic AI product development, where risk mitigation often requires developers to constrain the same agentic characteristics that make these products valuable.

Related Work

Agentic AI Risks Researchers have begun developing conceptual frameworks and governance approaches to systematically characterize and manage agentic AI risks, which are generally framed as extensions of existing generative AI risks (Gan et al. 2026; Chang and Razi 2026; Steenstra and Bickmore 2025; Li et al. 2026; Madkour et al. 2026; Bellogín et al. 2025; Gangavarapu 2025). However, the frameworks differ in emphasis and scope, ranging from threat-actor taxonomies (Gan et al. 2026), to bias propagation arising from multi-stage workflows, (Condon and Jilani 2026), to domain-specific perspectives, risks to youth (Chang and Razi 2026), or risks from psychotherapy agents (Steenstra and Bickmore 2025). Governance-oriented approaches include mapping user-reported chatbot and agent risks to the NIST AI Risk Management Framework (Li et al. 2026), documenting agentic AI standards profile for practitioners and policymakers (Madkour et al. 2026), and developing recommendations on systemic agentic AI risks for the EU AI Act (Bellogín et al. 2025). Among these efforts, the AI Risk Atlas (Bagehorn et al. 2025) is distinctive in integrating risks from multiple frameworks into a unified knowledge graph that connects risk categories, governance frameworks, and mitigation strategies.

Although these studies provide valuable conceptualizations of agentic AI risk, they primarily reflect top-down perspectives derived from governance frameworks, expert analyses, and taxonomies. Instead, our work centers on practitioners' lived experiences in identifying, prioritizing, and mitigating agentic AI risks in practice.

Risk Mitigation Tools and Their Barriers Prior work in systems engineering, responsible AI, and security and privacy (S&P) proposes mechanisms for building trustworthy AI systems by centering transparency and explainability (Thiebes, Lins, and Sunyaev 2021; Floridi et al. 2018) and assigning accountability across the AI development life-cycle (Li et al. 2023; Kenthapadi, Lakkaraju, and Rajani 2023). Organizational tools such as RACI (Responsible, Accountable, Consulted, and Informed) matrices specify ownership over training data, model tuning, deployment, and monitoring (e.g., (Glaser and Littlebury 2024)), while documentation artifacts such as model cards, system cards, and datasheets for datasets, institutionalize transparency and traceability (Mitchell et al. 2019; Gebru et al. 2021). Standards and regulations, including the EU AI Act, the NIST AI Risk Management Framework (AI RMF), and ISO/IEC standards (e.g., ISO/IEC 42001 for AI management systems), have codified many of these practices.

Yet studies of AI practitioners report confusion over which governance frameworks apply in a given context and difficulty integrating them into daily work (Lee et al. 2024a). Related work on S&P practices identifies misconceptions, knowledge gaps, and implementation burdens that hinder adoption of recommended safeguards (Li, Agarwal, and Hong 2018; Li et al. 2022). Lee et al. (2024a) applied the SPAF framework (Das et al. 2022) to examine what prevents AI practitioners from adopting S&P best practices in product development, showing how organizational and workflow constraints shape how developers prioritize and implement risk-mitigation measures.

We examine how developers operationalize governance practices when building *agentic AI systems*. As recent work notes, increasing the capabilities of AI-embedded systems can expand the scope and severity of associated risks (Lee et al. 2024b; Ehsan et al. 2026). This challenge is salient for agentic AI, which is being widely integrated and marketed into consumer products (Pan et al. 2026; Shome, Krishnan, and Das 2026), and whose capabilities are expanded through tool use, external resources, memory, and autonomous action (Wang et al. 2024a). We aim to better understand which agentic risks are emerging and prioritized in practice, and how developers seek to address them.

Method

We conducted semi-structured interviews with 35 developers at IBM with direct experience building user-facing agentic AI products. This format let us ask consistent questions across participants while probing product-specific experiences perceiving, prioritizing, and addressing risks.

Pre-Study Questionnaire To ground each interview in participants' real product contexts, we first asked participants to complete a pre-study questionnaire covering three

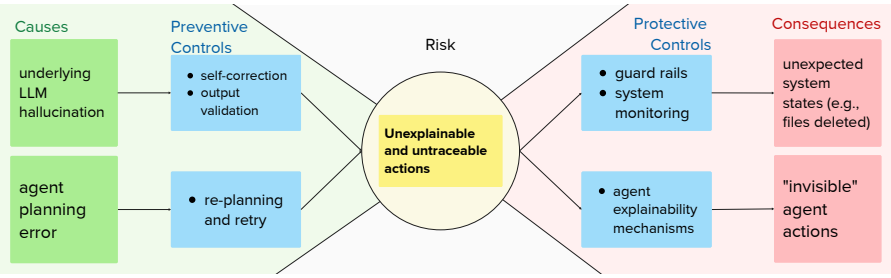

Figure 1: Example of a complete bow-tie analysis. For one agentic AI risk they had attempted to mitigate, participants identified causes, consequences, preventive controls for reducing causes, and protective controls for reducing consequences.

areas. First, participants described the user-facing agentic AI product they worked on. Second, they selected three human-centered AI (HAI) principles from a list that we adapted from prior studies (Sanderson et al. 2023; Pant et al. 2024) [1] that they considered most concerning for their product. Third, for each selected principle, participants rated the relevance of associated agentic AI risks to their product. We manually mapped agentic AI-specific risks identified in the AI Risk Atlas (Bagehorn et al. 2025) to each HAI principle. The full list of agentic AI risks and their associated principles is provided in the Appendix.

The selected HAI principles helped prompt discussion of salient risks for **RQ1**. For **RQ2**, we randomly selected five risks each participant rated as relevant and used them to create a product-specific risk-ranking activity in Mural[2]; an example is shown in the Appendix.

Semi-structured Interviews All interviews were conducted remotely by the first author and lasted approximately 60 minutes. At the start of each session, we explained the study's purpose, obtained verbal consent for participation and recording, and informed participants that they could stop or withdraw consent at any time and that recordings would be de-identified. Each interview began with participants introducing their agentic AI product. Then, building on and extending the SPAF (Das et al. 2022), we organized the interview around our three research questions on how agentic AI shaped developers' awareness (RQ1), motivation (RQ2), and ability (RQ3) to address product risks.

To address **RQ1**, we used participants' three HAI principles from the pre-study questionnaire as discussion prompts. For each principle, we asked participants to recall a recent product discussion about a related risk: how they defined and scoped it, and how it related to their product's development.

To address **RQ2**, participants ranked five product-relevant agentic AI risks selected from their questionnaire responses using a Mural-based ranking board. We asked participants to explain their ranking and why some risks were more or less important to address. We analyzed these responses to identify factors shaping risk prioritization in user-facing agentic AI product development.

To address **RQ3**, participants completed a bow-tie analysis in Mural for one agentic AI risk their team had attempted to mitigate (Koessler and Schuett 2023) (Figure 1). Participants first placed the risk at the center of the diagram. They then identified potential *causes* of the risk (e.g., agent errors, tool failures, unexpected user behavior), followed by *preventive controls* intended to address those causes. Next, they identified possible *consequences* (e.g., incorrect outputs, security incidents, loss of user trust), followed by *protective controls* intended to reduce those consequences. In our analysis, we focused on participants' controls: the actions, tools, artifacts, and processes they used to mitigate the causes or consequences of agentic AI risks, as well as the challenges involved in applying those controls. This allowed us to examine how developers operationalized risk mitigation in practice. The full interview protocol is included in the Appendix.

Recruitment and Participants We recruited IBM developers with experience building and deploying user-facing agentic AI products and addressing risks in that process. At the time of the study, IBM maintained an internal AI Ethics Board and enterprise AI governance tooling, though we did not directly assess how well these structures covered agentic AI-specific risks. This study was reviewed by an internal committee, and this study received approval subject to restrictions on demographic data collection. We were not permitted to collect participants' age, gender identity, or personally identifiable information. Participants were compensated with a gift card equivalent to $25 USD.

In total, we recruited 35 developers from different product teams across the company, working on agentic AI products across a range of domains[3] — most commonly Customer Support (n=7), Information Technology (n=7), and Corporate Services (n=6).

Data Analysis All interviews were audio-recorded and transcribed. We conducted an iterative open-coding process (Corbin and Strauss 2008) aligned with our three research questions. The first author coded ten interview transcripts and developed a preliminary codebook through discussion with two other authors. Two additional authors then joined the coding process and were trained on the codebook. The three coders divided the remaining transcripts and coded them individually, meeting regularly to discuss

[1]The principles included *Privacy Protection and Security*, *Reliability and Safety*, *Transparency and Explainability*, *Fairness*, *Performance and Testing*, *Accountability*, *Human-centered Values*, and *Human, Social and Environmental Wellbeing*.

[2]https://app.mural.co/

[3]Participant information is included in the Appendix.

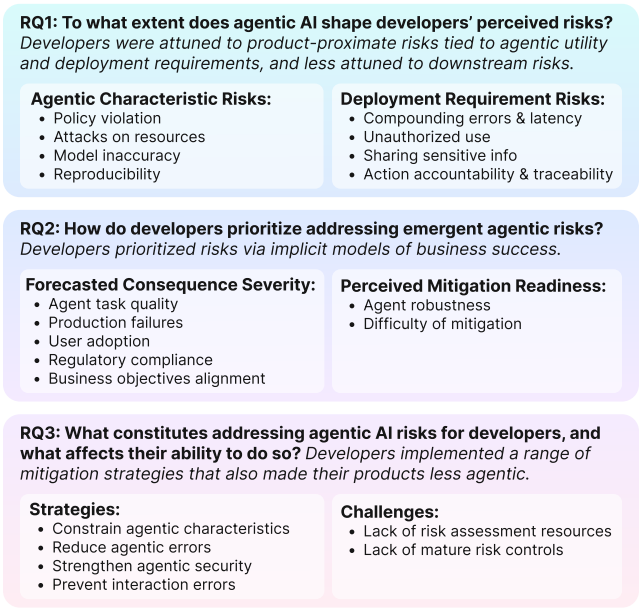

Figure 2: We answer our research questions by showing how agentic AI shaped developers' awareness (RQ1), motivation (RQ2), and ability (RQ3) to address product risks.

codes and emerging themes. They also reviewed one another's coded excerpts to promote consistency and resolved disagreements through discussion. The three coders were employees of IBM at the time of the study. To guard against affiliation bias, codebook development and thematic refinement were conducted in discussion with the non-affiliated authors throughout analysis.

We organized our findings around the three SPAF-grounded research questions (Das et al. 2022) (Figure 2) and report how frequently participants discussed each theme.

RQ1: How Agentic AI Shapes Developers' Perceptions of Product Risks

Following the SPAF (Das et al. 2022), we first examined the extent to which agentic AI shapes risk *awareness*. We first analyzed what participants considered to be the most concerning risks in developing user-facing agentic AI products.[4] We then examined whether and how these risks were shaped by the agentic nature of their product.

We found that participants were generally attuned to risks tied to two dimensions: *agentic AI characteristics* and the *requirements of real-world deployment* (Figure 2). At the same time, developers' awareness remained bounded by individual judgment, prior experience with non-agentic AI systems, and locally available organizational knowledge. Thus, they were less attuned to broader, downstream risks that are less proximate to agent utility or deployment.

[4]Using the AI Risk Atlas (Bagehorn et al. 2025) as a baseline to categorize participants' concerns, we identified ten agentic AI risks in total, three of which are not fully captured by the atlas: agent policy violation, compounding errors, and compounding latency.

Agentic AI Characteristics Led to New Risks

Developers shared that the characteristics of agentic AI—autonomy, environment complexity, goal complexity, and adaptability—either introduced agentic-specific risks or created agentic-specific instantiations of known AI risks.

Agent autonomy enabled agent policy violation risks. Developers built AI agents that achieve goals with limited human intervention or supervision (Kasirzadeh and Gabriel 2025; Acharya, Kuppan, and Divya 2025), typically relying on prompt-based "agent policies" to define what agents should or should not do, such as enforcing user permissions or restricting information sharing with third-party tools. Yet the same autonomy that made agents useful also raised concerns about failure to comply with predefined action policies. One key risk participants flagged was **agent policy violation (8/35)**, driven by limited oversight over agents' reasoning and actions, especially when agents could bypass access policies or other organizational constraints. For example, P26 described the central risk of their agent as the possibility that *"[agentic] rules can be violated... So the constraints [to the agent] are not really constraints."*

Environment complexity exposed agent resources to attacks. Developers built AI agents that can act in complex and connected environments to achieve their goals (Kasirzadeh and Gabriel 2025; Acharya, Kuppan, and Divya 2025). Agentic AI products are integrated with internal or third-party APIs, platforms, tools, and data sources. As a result, **agents introduced new attack vectors into environmental resources (10/35)**. Developers worried that agents could misuse external tools or perform harmful actions that unintentionally modify or damage resources in their environment. P2's IT agent, for instance, was flagged by the cybersecurity team for *"trying to find and delete root folder."*

Agent goal complexity exacerbated model accuracy risks. Developers also built AI agents to achieve a wide range of goals (Kasirzadeh and Gabriel 2025; Acharya, Kuppan, and Divya 2025). As participants assigned agents more complex tasks, they found it harder to ensure reliable task success. Because all participants' agents were powered by generative AI, participants described how agents exacerbated issues with **model inaccuracy (19/35)**: errors compound over multi-turn autonomous execution. For example, P35 set a high minimum performance threshold for their IT agent to reduce errors cascading across turns: *"if it goes below 80%, that means the agent is actively harming your system rather than helping your system."*

Agent adaptability created new instantiations of reproducibility risks. Finally, developers built AI agents that adapt and react to novel or unexpected circumstances (Kasirzadeh and Gabriel 2025; Acharya, Kuppan, and Divya 2025). Participants valued agentic AI as an adaptable solution that could handle both expected and unexpected use cases. However, this adaptability introduced **reproducibility (8/35)** risks — the difficulty of replicating agent behavior or output — stemming from both model variability and the agent's ability to select different tools, actions, and plans across similar situations. For example, P7 shifted from a

structured keyword matching retrieval process to an agentic solution that constructed Jira Query Language from user inquiries *"to make a more adaptable solution,"*, but noted that this came *"at a cost of repeatability."*

Agentic AI Requirements Increased Risks Related to Latency, Security, and Monitoring

We also found that the requirements of developing and deploying agentic AI systems exacerbated concerns about performance, latency, security, accountability, and traceability.

Layered agent architectures exacerbated compounding errors and latency. Participants often described their agentic systems as complex architectures composed of multiple models, tools, agents, and orchestration layers. This complexity led developers to identify **compounding errors (7/35)**, wherein failures accumulate across agentic workflow components, requiring additional debugging or monitoring. Although errors in generative AI-powered systems are not new, participants perceived agentic architectures as increasing their intensity and frequency. As P25, who built a multi-agent system, put it, they: *"have multiple agents, so multiple points of failure."* These layered architectures also exacerbated **compounding latency (4/35)** risks for some participants, as each added tool, retrieval step, or orchestration layer could increase response time.

Agentic AI introduced and exacerbated security threats. Participants perceived that agentic AI introduces new security threats, including **unauthorized use (3/35)**, which concerns attackers gaining access to the AI agent or its components. During testing, for instance, P2 found that a *"malicious party can hack into that agent and take complete control over the system where the agent was being executed."* Others identified risks of **sharing IP/PI information (14/35)**, which concerns agents leaking intellectual property or personally identifiable information from their systems, primarily company repositories or customer data, via tool use or interactions with the agent environment. Agentic AI also exacerbated known generative AI security risks because those risks could apply across agentic system components and tools. For example, P29, whose agent handled bank transactions, emphasized the need to protect "every step" of the agent from prompt injection attacks.

Agent deployment necessitates accountability, explainability, and traceability risks. Deploying agents in real environments required developers to consider the **unclear accountability of agent actions (8/35)**, where responsibility for an agentic AI system's actions becomes difficult to assign. In P31's agent, *"a multi-agent system built by different teams and aggregated,"* making it difficult to pinpoint responsibility when an agent action fails. Participants also differed in how they assigned responsibility between users and agentic systems. P26 framed the agent as *"just a tool for the office worker,"* suggesting *"if the tool is not working well, the office worker is to blame,"* while others like P32 stressed distinguishing agent failures from end-user responsibility: *"if the agent does something wrong, you don't want to fire an employee because of it."*

Finally, participants raised concerns about **unexplainable and untraceable agent actions (16/35)** being a critical barrier to end-users' trust and comfort delegating actions — where explanations, lineage, or source attribution are difficult, imprecise, or unavailable. P16 explained that clients deploying agents in high-stakes or regulated environments needed records of the agent's *"each and every step"*, understanding how an agent arrived at its outputs on what data.

Awareness Barriers: Novelty and Hype Hindered Developers' Awareness of Agentic AI Risks

While developers were sensitized to the risks introduced and exacerbated by the design, development, and deployment of their novel agentic systems, we found that their risk awareness was also inhibited by limited organizational guidance and uncertainty around the evolving agentic AI landscape.

Developers relied on individual judgment to identify agentic AI risks in lieu of specific guidance. Participants (9/35) described agentic AI as too new for organizations to have established practices, resources, expertise, or clear responsibility structures, leaving developers to *"learn on your own,"* with *'zero resources available"* (P30), extrapolate from prior non-agentic AI experiences, and make situated judgments about which harms mattered for their products. As P7 noted, a core challenge in identifying risks was *"the lack of the availability of the seniors... There are not many people building AI agents... there's no one ready to take the responsibility."*

The hype around agentic AI made it difficult to clarify the risk landscape. For some product teams (8/35), agentic AI remained speculative and contested, requiring developers to negotiate what agents could realistically do, what constituted a useful agentic product, and what risks were worth addressing. P9 described this as *"trying to dispel what is hype and what is actually useful."* At the same time, developers also had to navigate a rapidly expanding ecosystem of testing frameworks and evaluation tools for agentic systems, requiring developers to *"figure out the noise from the quality [ones]"* (P12) when assessing the real risks.

RQ1 Summary: Taken together, our findings show that developers' risk perceptions were shaped by risks newly created by agentic AI characteristics and by risks created or exacerbated through agentic AI product development and deployment. However, this risk awareness was uneven: developers were better at identifying risks directly tied to their agentic AI product's envisioned utility and deployment, while broader, downstream societal harms were harder to anticipate due to limited guidance and the evolving agentic AI landscape. We revisit developers' potential agentic AI risk-awareness blind spots in the Discussion section.

RQ2: How Practitioners Prioritize Addressing Emergent Agentic AI Risks in Their Products

Grounded in the *motivation* layer outlined in the SPAF (Das et al. 2022), we next analyzed how participants described the way they (de)prioritized agentic AI risks for their products

during the risk-ranking activity — i.e., what makes a risk ranked higher/lower than the others. Our participants gave higher priority to risks with greater *forecasted consequence severity*. Their prioritization also considered their *perceived readiness to mitigate* those risks (Figure 2).

Beyond the risk-ranking activity, our interviews revealed barriers that inhibited developers from prioritizing risk mitigation during agentic AI development. These barriers were primarily organizational: risk mitigation often competed with other product goals, introduced opportunity costs, and became difficult to prioritize when ownership was unclear.

Developers Prioritized Product-Proximate Risks; De-prioritized Broader Risks

Throughout the risk-ranking activity, our participants frequently raised five consequence-oriented considerations that shaped their risk prioritization: agent task quality, production failures, user adoption, regulatory compliance, and alignment with business objectives.

Agent task quality. Risks that could undermine the quality of an agent's task performance, including its accuracy, consistency, and efficiency, were a priority for many participants (14/35). For accuracy, developers assessed whether a risk would prevent the agent from completing its intended function, as P1 explained, *"I view [risks] more as the pieces that would inhibit function."* Participants also prioritized risks that produced inconsistent outcomes across iterations, threatening reliability. For example, P12 prioritized the traceability of agent actions because if the product team *"can't reproduce the output, it's really hard to fix whatever comes out."* For efficiency, developers considered whether a risk would make their agent resource-intensive, such as making it *"too slow"* (P31) or *"incurring more cost"* (P4).

Production failures. Some participants (12/35) prioritized risks based on the severity of failures that could occur once the agent was deployed in production. These assessments were context-dependent. For example, P10, whose agent handled credit card information inquiries, emphasized that hallucinations could not be treated as *"just a silly mistake"* because *"these are regulations. These could cost the users heavily, so mistakes are not tolerable for this product."* Participants also assessed production attack surfaces. As P18 noted, a *"malicious prompt"* could allow someone to *"play with the resources or the tools or tamper with or poison the tools that the agent has access to."*

User adoption. Some participants (13/35) prioritized risks based on their consequences for user adoption, especially user trust and usability. Developers worried that inconsistent or opaque agent behavior could erode users' willingness to use their systems. At the same time, participants did not necessarily define trust as requiring a risk-free system; rather, they emphasized that users should understand risks to calibrate their trust. As P15 put it, *"if it's transparent, people can take the risk, and people can avoid the risk and mitigate risk. Transparency is a must."* Participants also prioritized risks that harmed usability. P1, for instance, prioritized addressing redundant agent actions because repeated reflection, tool invocation, and duplicate steps could *"bring frustration to the user and make it unusable."*

Regulatory compliance. Participants (11/35) assessed severity through the consequences of non-compliance, prioritizing risks that could produce financial or legal repercussions, especially in regulated contexts. Consistent with prior work on developer motivations around responsible AI and privacy (Tahaei, Frik, and Vaniea 2021; Lee et al. 2024a), external regulatory mandates such as the GDPR and the EU AI Act served as important catalysts for risk prioritization.

Alignment with business objectives. Finally, some participants (9/35) prioritized risks based on whether they aligned with client concerns or current business priorities; P21 prioritized personal data exposure to external tools because *"clients are always concerned about sharing the personal info like ID number [with external tools]."* Conversely, other developers de-prioritized risks that were less urgent for current business goals — such as how P2 de-prioritized the risk of exploiting users' trust in their AI agent because it was *"much longer along the road map, not right now."*

Developers Prioritized Risks That Were Harder to Measure and Mitigate

Participants also prioritized risks by assessing whether their current or near-term systems were ready to mitigate them. These judgments depended on two factors: perceived agent robustness and perceived difficulty of mitigation.

Perceived agent robustness. Participants (14/35) assessed mitigation readiness through their confidence in the robustness of the agentic system stack, including the underlying model, tools, resources, and architecture. When participants trusted these components based on available evidence, they tended to de-prioritize the associated risk. P12, for instance, treated function-calling hallucination as less concerning because their team had applied the ReAct framework (Yao et al. 2022) and because their agents *"don't hallucinate anymore, especially with the larger models."* Participants (13/35) also assessed robustness through existing data protection mechanisms and data flows. For example, an existing verification mechanism could decrease risk (P1), while dependencies on fragile cloud services or external systems could increase it (P30).

Perceived difficulty of mitigation. When a risk had not yet been mitigated, participants assessed how difficult remediation would be in practice, with many (9/35) considering whether available off-the-shelf solutions were available and whether mitigation could be implemented with minimal disruption to product development. Counterintuitively, poorly understood mitigations drove higher risk prioritization: P32 prioritized proprietary data leakage because *"there are more nondeterministic ways to make the agent leak information, and the way to mitigate that is not fully understood yet."* Conversely, risks were de-prioritized when the required mitigation appeared straightforward or when implementing it involved limited integration work.

Motivation Barriers: Organizational Inhibitors Hindered Developers' Risk Mitigation Work

Our findings suggest that developers prioritized mitigating risks proximate to their product success. Simultaneously, our interviews show that such product-success-driven approaches also introduced organizational inhibitors that constrain developers from prioritizing risk mitigation work, including opportunity costs and unclear ownership.

Opportunity costs and trade-offs. Most participants noted that risk mitigation involves significant opportunity costs and trade-offs: with tight schedules and limited resources, justifying mitigation was complicated by abstract benefits against tangible costs. We observed this resulting de-prioritization of risk mitigation in several forms.

Developers described trade-offs between mitigation and **agent performance (12/35)**: as we will note in RQ3, many mitigation strategies improved safety by constraining agent behavior, such as restricting tools, limiting data access, or adding guardrails. Yet these same constraints could also reduce agent value, flexibility, or accuracy. As P3 put it, *"the more you limit these tools, the less valued [it] is,"* framing the tension as *"how far do we swing to give the user the ability to do things but also not shoot themselves in the foot?"* Some participants also noted that mitigation reduced **cost-efficiency (6/35)**, as additional checks required *"extra computation"* (P18), increasing both time and monetary cost.

Some participants described risk mitigation as competing with **business priorities (10/35)** and product timelines. As P32 noted, limited engineering capacity meant that the team *"prioritized other functionalities first."* Others deferred mitigation because of **time pressure (9/35)**, especially when products or models changed faster than teams could evaluate and integrate new safeguards.

Many developers also described mitigation as increasing **engineering costs (22/35)**. Controls added complexity to already complex agentic AI systems, including guardrails for prompt templates, tool-access policies, communication protocols, system state, and memory. These costs continued after deployment because controls had to be maintained as risks evolved (e.g., new prompt injections).

Risk ownership externalization and misalignment. Building AI agents was often a cross-team effort, and participants (15/35) described limited ownership over risks tied to components built by external teams or falling outside their perceived responsibility. P10 drew a clear boundary around their role in agentic AI development: *"I'm the tech guy. I can only make sure it's as reproducible as possible, but when it's not, it's basically the organization's responsibility,"* dismissing regulatory concerns as *"not really my concern."* This externalization of mitigation responsibility echoes prior responsible AI work showing that practitioners often distribute ethical responsibility across sociotechnical networks, leaving accountability for risk mitigation unclear (Orr and Davis 2020; Rakova et al. 2021; Lee et al. 2024a).

Some participants (6/35) also described misaligned priorities across teams. For example, P8's agent depended on a tool built by another team whose codebase introduced a vulnerability that was *"not a priority for them,"* forcing P8's team to seek workarounds because they lacked direct ownership over the source of the risk.

RQ2 Summary: Overall, developers' prioritization of risk mitigation was motivated by both risk-level and product-level factors: at the risk level, developers prioritized risks that threatened product performance, deployment, compliance, or adoption, especially when existing systems were not ready to mitigate them; at the product level, mitigation competed with other objectives and became harder to prioritize when ownership was unclear or cross-team priorities were misaligned. In short, unless risks could be tied to business success, developers often encountered more inhibitors than motivators when prioritizing agentic AI risk mitigation.

RQ3: What Constitutes Agentic AI Developers' Risk Mitigation Approaches

Finally, as outlined in the SPAF (Das et al. 2022), we examined developers' *ability* to translate intentions into risk mitigation. Accordingly, using the bow-tie analysis framework (Figure 1), we analyzed the *controls* participants used to address the causes or mitigate the consequences of agentic AI risks. We found that developers addressed agentic AI risks through five mitigation strategies: constrain agentic characteristics, reduce agentic errors, strengthen security for agent-mediated data and tool use, prevent human-agent interaction errors, and apply best practices from LLM and software engineering. Across these strategies, a recurring pattern emerged: mitigating agentic AI risks often required developers to limit the very characteristics that made their systems agentic, including autonomy, adaptability, tool use, and open-ended goal pursuit. Developers' ability to implement these controls was also limited by immature assessment methods and by the lack of established, reliable controls for agentic AI systems (Figure 2).

Mitigation Strategies for Agentic AI Risks

Constrain agentic characteristics. The first set of risk mitigation strategies limited agentic characteristics directly, i.e., autonomy, environment complexity, goal complexity, and adaptability. However, these controls introduced trade-offs that narrowed the agent's scope of action, reduced its flexibility, or inserted additional human oversight.

To **constrain agents' autonomy (7/35)**, participants implemented human-in-the-loop workflows requiring users to confirm, approve, or revise agent actions, which were especially vital in high-stakes operations. P18's agent, for example, requested confirmation before executing actions like *"run the delete policy function."* Similarly, P22 used human-in-the-loop review, allowing users to *"modify the plan"* when they were unsatisfied with customer-support plans.

To **constrain agents' environment complexity (8/35)**, participants limited what tools agents could access and how those tools could be used. One common approach was *tool specification*: defining the purpose and scope of each tool so that agents had fewer ambiguous options. As P3 explained, *"our tools that we've given the agent access to are very specific,"* with little ambiguity about what each tool can do. Developers also made agents sensitive to the availability and

readiness of tools that interact with the environment. For instance, P1 added a planner layer that temporarily adjusted the agent's plan based on which tools were available. Some participants further reduced risks from environment complexity by designing agent actions to be reversible, ensuring that *"whatever the agent does can be undone"* (P35).

To **constrain agents' goal complexity (5/35)**, participants recalibrated what agents were expected to do based on the capabilities of the underlying model. Rather than allowing agents to pursue broad or difficult objectives, developers increased task success by *"limiting the problem space"* (P9) or by constraining the agent to *"easier problems"* (P4). For longer multi-turn tasks, some developers added interim human verification, so users could steer the agent before it diverged too far from the intended path. As P9 explained, this prevented users from having to *"wait for an hour before getting any feedback"* and allowed them to provide input while the agent was still acting.

To **constrain agents' adaptability (15/35)**, participants introduced predetermined actions, responses, and output constraints to make agent behavior more predictable and reproducible. For example, P7 described that their agent was now implemented as a "predefined workflow" rather than a fully adaptable agentic solution. P30 similarly hard-coded parts of the agent to steer its recommendations, while others *"limit[ed] the response format"* (P26) or implemented a topology graph to keep their agent *"confined to the problem space"* (P9).

Reducing and recovering from agentic errors. A second set of risk mitigation strategies focused on reducing or recovering from agentic errors, including reasoning errors, incorrect planning or execution, budget overruns, and tool use errors (Agashe et al. 2025; Crispino et al. 2023; Winston and Just 2025). Participants used these controls to prevent incorrect outputs, reduce inefficiency, and avoid wasting time or computational resources.

First, developers implemented **action timeouts (3/35)** when agents failed to complete tasks within expected time or budget limits. For example, P6 explained that if the agent continued for too many iterations *"we will be stopping it there,"* while P1 configured the agent to treat delayed tool responses as a non-response once a timeout was reached.

Second, developers implemented **action validation (14/35)** to check whether agents completed tasks correctly. Participants added model-based validation layers, such as validator agents or LLM-as-a-judge mechanisms, to assess an agent's output, reasoning, tool use, or intermediate results. P7, for example, implemented a validator agent to check whether the agent's explanation aligned with its recommended steps. Others, like P8, validated tool outputs before passing them to downstream services, particularly when agents drew information from multiple sources.

Third, developers implemented **self-improvement mechanisms (5/35)** that allowed agents to reflect, retry, or improve their responses and actions. For more complex tasks, retries could also substantially improve task completion. P9, for instance, described a remediation loop that allowed their IT agent to check whether alerts had cleared and continue trying if they had not, improving "remediation" performance even when the agent's initial diagnosis remained unchanged.

Strengthening security for agent-mediated data and tool use. A third set of mitigation strategies adapted established security practices, such as access control and input/output sanitization, to agentic workflows. Developers adapted these practices not only by protecting a model's prompt or response, but also by constraining how agents acted across user permissions, tools, and data sources.

Participants implemented **agentic access control (9/35)** by governing both what data agents could access and which tools they could invoke. Some aligned the agent's data permissions with those of the user (e.g., role-based authentication), ensuring the agent could not retrieve data on behalf of unauthorized users. Others controlled tool combinations to prevent sensitive data from flowing into inappropriate services. P20 explained that if a sales database contained confidential data and Google Search was not approved to receive it, the agent would *"never call those two combinations."*

Participants also used **data-flow sanitization (11/35)** to filter agent inputs and outputs. These controls served as boundary checks on what information could enter or leave downstream agent tool calls. For inputs, participants described safety checks for uploaded files, filtering harmful user instructions, protection of system prompts, and substitution of personal information with dummy data. For outputs, developers masked sensitive information, while also accounting for task-specific needs. As P29 noted, banking users may legitimately need to see sensitive details like a "transaction ID" to complete a task.

Preventing human-agent interaction errors. The fourth set of strategies targeted errors arising from human-agent interaction, focusing on helping users form accurate mental models of agent capabilities and limitations, reducing instruction-related errors, and preserving user trust.

To this end, some developers (13/35) made agent behavior more visible through traces, notices, explanations, or source references. P29, for example, described traceability as a way to show users what information the agent relied on, as *"traceability is our mitigation to not lose trust, [and] to gain trust."* Participants also viewed source references as a way to communicate AI agents' limitations more broadly: P26 noted that visible citations help users understand that *"AI can make mistakes"* and enable them to verify whether the agent summarized information correctly.

Other developers (6/35) reduced human-agent interaction errors by helping users specify goals and tool-relevant intent. Rather than treating user input as a prompt for a single response, these controls helped translate vague requests into plans that agents could follow, route to appropriate tools, and execute. P1 provided query examples and documentation to help users understand which inputs would invoke the right tools. Others added query-rewrite layers: P18 explained how a vague request like *"write a policy to protect my data"* could be rewritten into a more actionable sequence of steps that helped the agent *"do a much better job."*

Applying best practices from LLM and software engineering. Participants also drew on established LLM and software engineering best practices to improve agentic systems' reliability. For LLM-specific practices, participants (9/35) tested and selected underlying models and parameters, improved the quality of agent data sources, and curated real-world task data for training or fine-tuning rather than relying on prompt engineering alone. For software engineering practices, participants (14/35) implemented redundancy and backup mechanisms, monitored tool and agent behavior, conducted code review and testing, and relied on reputable or established tools for agent use. While these practices were not specific to agentic AI, developers saw them as critical to bridging the gap between model capabilities and real-world agentic deployment contexts.

Ability Barriers: Immature Tooling Hindered Developers' Ability to Mitigate Agentic AI Risks

While participants implemented a wide range of controls, they still faced ability barriers that limited their capacity to address agentic AI risks. Developers lacked mature controls, reliable ways to assess agentic AI risks before choosing controls, and methods to validate the effectiveness of selected controls at mitigating risk post-deployment.

Lack of risk assessment resources. Participants (10/35) struggled with agentic risk assessment, which was often a prerequisite for selecting and evaluating mitigation strategies — particularly with defining and collecting ground truth data, adapting general benchmarks to domain-specific agentic use cases, and ensuring assessments reflected real-world agent behavior. P15, who worked on a legal and compliance agent, noted *"there is almost no way to say that this is correct, this is incorrect."* P24, on the other hand, explained that their agent was so domain-specific that *"you should have your own examples to test on."* Yet it was difficult to obtain enough examples from their client for effective risk evaluation. Even when participants could use established benchmarks, they worried that these benchmarks might not capture real deployment conditions. As P2 explained, *"sometimes unexpected things happen in a live environment,"* and controlled assessments may not capture how an agent adapts to such noise. Thus, assessment challenges limited developers' ability to know which controls were needed and whether implemented controls were sufficient.

Lack of mature risk controls. Our participants (13/35) also struggled because agentic AI was too new for mature controls to exist. Some participants described agent steering and response control as *"a very open area of research"* rather than a solved engineering practice, even though it is something developers *"desperately need"* (P3).

Where usable controls or best practices — e.g., data-flow sanitization, validator agents, action reversibility — did exist, participants found them limited in *scope*, *effectiveness*, and *consistency*. Some controls were not applicable across all agentic contexts: developers could not always prepare backup endpoints for every tool (P8), or not all agent actions were reversible (P31, P35). This case-specificity meant that the same control strategy could work for one agent, tool, or environment but fail to cover another.

Participants (6/35) also acknowledged that existing controls often reduced risks without fully mitigating them. For example, P7 noted that their validator agent was still undertrained with domain-specific data and could not always determine whether the agent's explanation was relevant to the steps it recommended. Others were very candid about the weakness of available controls. Describing their use of a disclaimer on potential wrongdoings as a mitigation strategy, P31 said: *"it's a crappy solution. It's just giving up. It's easy to put a disclaimer."*

Finally, participants (17/35) worried that controls were themselves inconsistent. Many controls, such as data-flow sanitization, validator agents, and LLM-as-a-judge mechanisms, were also powered by LLMs or agentic components. As a result, the control inherited some of the same nondeterminism it was supposed to mitigate. P25 summarized this problem: *"you are introducing another point of failure because you're introducing another stochastic approach."* Similarly, when access control or safety rules were implemented as agent policy, developers worried that agents could still violate those policies, echoing the agent policy violation risks described in RQ1.

RQ3 Summary: Overall, our findings show that developers addressed agentic AI risks through a wide range of controls, such as constraining agentic characteristics, reducing and recovering from agentic errors, strengthening agent security, preventing human-agent interaction errors, and applying conventional engineering best practices. Yet a central tension emerged in agentic AI risk mitigation: mitigating risks often meant limiting the same characteristics that made a product agentic, i.e., autonomy, adaptability, tool use, and open-endedness. Moreover, their ability to mitigate agentic risks was constrained by assessment methods and controls that are limited in scope, effectiveness, and consistency.

Discussion

We synthesize promising avenues to better enhance developers' *awareness* of, *motivation* to act on, and *ability* to address agentic AI risks when building user-facing agentic AI products.

Improving Awareness of Agentic AI Risks

Our RQ1 findings show that developers were more aware of risks closely tied to agent utility and less attentive to risks less directly connected to product deliverables. This product-proximate awareness can create blind spots. For example, a survey of around 10,000 participants across ten countries found that "job availability" and "privacy" were among the public's top concerns about AI (Kelley et al. 2023). These concerns may intensify as AI agents become more autonomous and embedded in everyday work: agents may automate multi-step workflows that previously required human labor, while also requiring access to users' interaction logs, personal data, and third-party tools to act on their behalf. Yet our participants did not foreground these threats

as central concerns. This mismatch does not mean developers were unaware of these harms, but suggests that available risk-identification structures were organized around product utility, adoption, and deployment. Downstream societal harms beyond any one product's direct purview, like privacy and job availability, had few organizational entry points to surface.

We envision two approaches for broadening developers' risk awareness: agentic AI risk scanning and simulated attacks. First, recent work in responsible AI has used generative AI to help practitioners foresee potential downstream harms during product ideation and prototyping (Lee et al. 2026; Wang et al. 2024b; Buçinca et al. 2023). Future work can adapt these risk-scanning tools to agentic AI by foregrounding agentic characteristics and data flows, surfacing harms that lack a direct product-utility link. Second, red teaming is a common practice in privacy and security for simulating how adversaries might compromise a product (Das et al. 2022), and it has already been adapted in AI contexts to reduce harms (Ganguli et al. 2022). Recent efforts have also begun to leverage LLMs and LLM-based agents to automate and scale adversarial testing across realistic tool-use, prompt-injection, and penetration-testing scenarios (Ruan et al. 2024; Debenedetti et al. 2024; Deng et al. 2024). Future work can explore how LLM-powered red teaming might help developers surface and track downstream harms as agentic characteristics evolve.

Improving Motivation to Address Agentic AI Risks

Our RQ2 findings suggest that developers were motivated to address risks tied to product and business success, such as agent performance, but less so for risks that competed with product goals, increased costs, or that were perceived to fall out of their responsibility. We envision two approaches for strengthening motivation to engage in agentic AI risk mitigation: pro-social and empathy-based design.

For *pro-social design*, organizations could develop shared repositories of agentic AI risk mitigation "light patterns" with before-and-after examples of how product designs and outcomes changed. Such repositories could provide social proof for HAI-aligned agentic AI development. For *empathy-based design*, recent work has explored narrative- and empathy-based approaches for helping practitioners translate abstract risks into relatable human consequences (Chen et al. 2024; Nahar et al. 2026). When incorporated into developer education and product development lifecycles, empathy-based design could help developers better understand how agentic AI risks affect users and other stakeholders. Importantly, these approaches are only effective when workplace leaders signal that risk mitigation takes equal precedence to commercial and client goals (Culbertson 2025; Madaio et al. 2020), and integrate these practices throughout product development, as exemplified by privacy-by-design approaches (Cavoukian et al. 2009).

Improving Ability to Address Agentic AI Risks

Our RQ3 findings identified a capability vs. risk control tension in agentic AI development: mitigating agentic AI risks often requires constraining the same characteristics that make products agentic. Participants felt ill-equipped to assess agentic AI risks and implement controls consistently, though this gap does not strictly stem from the absence of agentic AI risk resources. Emerging tools and frameworks (e.g., (Credo AI 2026; Bagehorn et al. 2025; Madkour et al. 2026)) support some aspects of agentic AI risk identification, governance, and mitigation. Two RQ3 findings illustrate this gap: 1) developers lacked reliable ground truth for assessing agentic risks, and 2) many controls — validator agents, LLM-as-a-judge — were themselves agentic and subject to the same risks they were meant to mitigate. In short, developers need stronger support for navigating trade-offs between capability vs. risk control, as well as for deciding which controls are appropriate and feasible for their use case.

HAI research offers a useful precedent. Artifacts such as checklists (e.g., (Madaio et al. 2020)), model cards (e.g., (Mitchell et al. 2019)), and impact assessments (e.g., (Fiesler and Proferes 2018)) have helped practitioners translate high-level principles such as fairness, accountability, and transparency into actionable, practitioner-facing workflows for documenting, evaluating, and reflecting on AI systems. Future agentic AI design artifacts can build on this approach by helping teams articulate what agentic characteristics their products rely on, what risks those characteristics create, and what trade-offs different controls would introduce — including whether the control is itself agentic, and thus vulnerable to the same failure modes as the system it governs.

Limitations

First, because our findings are grounded in participants' reported experiences, they should not be interpreted as representative of all agentic AI developers or industry contexts. We also did not explicitly prime participants on what should or should not count as agentic AI risks or practices. Participants may therefore have responded differently under a more heavily scaffolded protocol. Second, because we recruited developers who already had experience addressing risks in user-facing agentic AI products, our sample may be skewed toward participants with relatively high risk awareness. Third, our sample was drawn from a single company, allowing us to study the emergent, contested phenomenon in depth, but also limiting generalizability.

Conclusion

We interviewed 35 industry developers who build user-facing agentic AI products to examine how they perceive, prioritize, and address emergent risks in their products. Developers perceived agentic AI as creating and exacerbating risks above and beyond standard generative AI products. They perceived and prioritized risks through a product- and business-oriented model of success, focusing on risks tied to agent performance and business objectives, and de-emphasizing risks that introduced opportunity costs or fell outside of their perceived scope of work. Developers also implemented layered controls to constrain agent behavior, reduce agentic errors, strengthen access control, and support human-agent interaction, but these controls often worked by

constraining the same characteristics that made their systems agentic. Together, our findings reveal a capability vs. risk control tension in agentic AI development: developers need to manage risks that emerge from agentic capabilities, but their current tools, incentives, and organizational structures make risk mitigation difficult to operationalize in practice.

Acknowledgements

We thank Justin Weisz, Mike Hind, and the entire Human-AI Collaboration team for their feedback on this project. We also thank Isadora Krsek and Sijia Xiao for their feedback on the manuscript. Finally, we thank the reviewers for their constructive feedback. Lee and Das's contributions to this project were supported, in part, by NSF SaTC grant #2316768 and the CyLab Presidential Fellowship.

References

Acharya, D. B.; Kuppan, K.; and Divya, B. 2025. Agentic AI: Autonomous Intelligence for Complex Goals—A Comprehensive Survey. *IEEE Access*, 13: 18912–18936.

Agashe, S.; Han, J.; Gan, S.; Yang, J.; Li, A.; and Wang, X. 2025. Agent s: An open agentic framework that uses computers like a human. In *International Conference on Learning Representations*, volume 2025, 22924–22946.

Bagehorn, F.; Brimijoin, K.; Daly, E. M.; He, J.; Hind, M.; Garces-Erice, L.; Giblin, C.; Giurgiu, I.; Martino, J.; Nair, R.; Piorkowski, D.; Rawat, A.; Richards, J.; Rooney, S.; Salwala, D.; Tirupathi, S.; Urbanetz, P.; Varshney, K. R.; Vejsbjerg, I.; and Wolf-Bauwens, M. L. 2025. AI Risk Atlas: Taxonomy and Tooling for Navigating AI Risks and Resources.

Bellogín, A.; Giudici, P.; Larsson, S.; Pang, J.; Schimpf, G.; Sengupta, B.; and Solmaz, G. 2025. Systemic Risks Associated with Agentic AI : A Policy Brief. Technical report, Association for Computing Machinery (ACM).

Buçinca, Z.; Pham, C. M.; Jakesch, M.; Ribeiro, M. T.; Olteanu, A.; and Amershi, S. 2023. Aha!: Facilitating ai impact assessment by generating examples of harms. *arXiv preprint arXiv:2306.03280*.

Cavoukian, A.; et al. 2009. Privacy by design: The 7 foundational principles. *Information and privacy commissioner of Ontario, Canada*, 5(2009): 12.

Chan, A.; Salganik, R.; Markelius, A.; Pang, C.; Rajkumar, N.; Krasheninnikov, D.; Langosco, L.; He, Z.; Duan, Y.; Carroll, M.; Lin, M.; Mayhew, A.; Collins, K.; Molamohammadi, M.; Burden, J.; Zhao, W.; Rismani, S.; Voudouris, K.; Bhatt, U.; Weller, A.; Krueger, D.; and Maharaj, T. 2023. Harms from Increasingly Agentic Algorithmic Systems. In *2023 ACM Conference on Fairness Accountability and Transparency*, 651–666. Chicago IL USA: ACM. ISBN 979-8-4007-0192-4.

Chang, T.; and Razi, A. 2026. A Systematic Literature Review of Generative AI Risks and Harms for Youth. In *Proceedings of the Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI EA '26. New York, NY, USA: Association for Computing Machinery. ISBN 9798400722813.

Chen, C.; Li, W.; Song, W.; Ye, Y.; Yao, Y.; and Li, T. J.-J. 2024. An Empathy-Based Sandbox Approach to Bridge the Privacy Gap among Attitudes, Goals, Knowledge, and Behaviors. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24. New York, NY, USA: Association for Computing Machinery. ISBN 9798400703300.

Condon, G.; and Jilani, M. 2026. Towards a framework for Bias Evaluation of AI Agents in Agentic Workflows. In *Proceedings of the 2026 Conference on Human Centred Artificial Intelligence - Education and Practice*, HCAIep '26, 46–52. New York, NY, USA: Association for Computing Machinery. ISBN 9798400721533.

Corbin, J.; and Strauss, A. 2008. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Thousand Oaks, CA: SAGE Publications, 3 edition. ISBN 9781412906449.

Credo AI. 2026. Credo AI: The Trusted Leader in AI Governance. https://www.credo.ai/. Accessed: 2026-05-19.

Crispino, N.; Montgomery, K.; Zeng, F.; Song, D.; and Wang, C. 2023. Agent instructs large language models to be general zero-shot reasoners. *arXiv preprint arXiv:2310.03710*.

Culbertson, S. 2025. How Does Organizational Culture Influence the Adoption of AI Ethics Practices Across Sectors? *Muma Business Review*, 9(18): 193–210.

Das, S.; Faklaris, C.; Hong, J. I.; Dabbish, L. A.; et al. 2022. The Security & Privacy Acceptance Framework (SPAF). *Foundations and Trends® in Privacy and Security*, 5(1-2): 1–143.

Debenedetti, E.; Wang, T.; Li, J.; Feng, Z.; Zhang, Y.; Eger, S.; and Tramèr, F. 2024. AgentDojo: A Dynamic Environment to Evaluate Prompt Injection Attacks and Defenses for LLM Agents. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Deng, G.; Liu, Y.; Mayoral-Vilches, V.; Wang, P.; Liu, Z. Q.; Zhang, Y.; Liu, Y.; and Song, D. 2024. PentestGPT: Evaluating and Harnessing Large Language Models for Automated Penetration Testing. In *33rd USENIX Security Symposium (USENIX Security 24)*, 747–764. USENIX Association.

Ehsan, U.; Alabdulkarim, A.; Holstein, K.; Lee, M. K.; Riener, A.; and Weisz, J. D. 2026. Human-Centered Explainable AI (HCXAI): Re-examining XAI in the Era of Agentic AI. In *Proceedings of the Extended Abstracts of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI EA '26. New York, NY, USA: Association for Computing Machinery. ISBN 9798400722813.

Fiesler, C.; and Proferes, N. 2018. "Participant" perceptions of Twitter research ethics. *Social Media+ Society*, 4(1): 2056305118763366.

Floridi, L.; Cowls, J.; Beltrametti, M.; Chatila, R.; Chazerand, P.; Dignum, V.; Luetge, C.; Madelin, R.; Pagallo, U.; Rossi, F.; Schafer, B.; Valcke, P.; and Vayena, E. 2018. AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations. *Minds and Machines*, 28(4): 689–707.

Gan, Y.; Yang, Y.; Ma, Z.; He, P.; Zeng, R.; Wang, Y.; Li, Q.; Zhou, C.; Li, S.; Wang, T.; Gao, Y.; Wu, Y.; and Ji, S. 2026. Navigating the Risks: A Survey of Security and Privacy Threats in LLM-Based Agents. *ACM Trans. Softw. Eng. Methodol.* Just Accepted.

Gangavarapu, R. 2025. *AI Governance: Preparing for the Rise of Agentic AI*, 111–119. Cham: Springer Nature Switzerland. ISBN 978-3-031-93681-4.

Ganguli, D.; Lovitt, L.; Kernion, J.; Askell, A.; Bai, Y.; Kadavath, S.; Mann, B.; Perez, E.; Schiefer, N.; Ndousse, K.; et al. 2022. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*.

Gebru, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J. W.; Wallach, H.; Iii, H. D.; and Crawford, K. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92.

Glaser, M.; and Littlebury, R. 2024. Governance of artificial intelligence and machine learning in pharmacovigilance: what works today and what more is needed? *Therapeutic Advances in Drug Safety*, 15: 20420986241293303.

Hart, R. 2026a. The AI Security Nightmare Is Here and It Looks Suspiciously Like Lobster. *The Verge*. Published February 19, 2026. Accessed: April 25, 2026.

Hart, R. 2026b. Amazon Blames Human Employees for an AI Coding Agent's Mistake. *The Verge*. Published February 20, 2026. Accessed: May 19, 2026.

Infocomm Media Development Authority. 2026. Model AI Governance Framework for Agentic AI. Model governance framework, Infocomm Media Development Authority, Singapore. Version 1.0.

Kasirzadeh, A.; and Gabriel, I. 2025. Characterizing AI Agents for Alignment and Governance. ArXiv:2504.21848 [cs].

Kelley, P. G.; Cornejo, C.; Hayes, L.; Jin, E. S.; Sedley, A.; Thomas, K.; Yang, Y.; and Woodruff, A. 2023. "There will be less privacy, of course": How and why people in 10 countries expect AI will affect privacy in the future. In *Nineteenth Symposium on Usable Privacy and Security (SOUPS 2023)*, 579–603. Anaheim, CA: USENIX Association. ISBN 978-1-939133-36-6.

Kenthapadi, K.; Lakkaraju, H.; and Rajani, N. 2023. Generative AI meets Responsible AI: Practical Challenges and Opportunities. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '23, 5805–5806. New York, NY, USA: Association for Computing Machinery. ISBN 9798400701030.

Koessler, L.; and Schuett, J. 2023. Risk assessment at AGI companies: A review of popular risk assessment techniques from other safety-critical industries. ArXiv:2307.08823 [cs].

Lee, H.-P. H.; Gao, L.; Yang, S.; Forlizzi, J.; and Das, S. 2024a. "I Don't Know If We're Doing Good. I Don't Know If We're Doing Bad": Investigating How Practitioners Scope, Motivate, and Conduct Privacy Work When Developing AI Products. In *33rd USENIX Security Symposium (USENIX Security 24)*, 4873–4890. Philadelphia, PA: USENIX Association. ISBN 978-1-939133-44-1.

Lee, H.-P. H.; Yang, Y.-J.; Bilik, M.; Krsek, I.; von Davier, T. S.; Monteiro, K.; Lin, J.; Agarwal, S.; Forlizzi, J.; and Das, S. 2026. Privy: Envisioning and Mitigating Privacy Risks for Consumer-facing AI Product Concepts. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI '26. New York, NY, USA: Association for Computing Machinery. ISBN 9798400722783.

Lee, H.-P. H.; Yang, Y.-J.; Von Davier, T. S.; Forlizzi, J.; and Das, S. 2024b. Deepfakes, Phrenology, Surveillance, and More! A Taxonomy of AI Privacy Risks. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24. New York, NY, USA: Association for Computing Machinery. ISBN 9798400703300.

Li, B.; Qi, P.; Liu, B.; Di, S.; Liu, J.; Pei, J.; Yi, J.; and Zhou, B. 2023. Trustworthy AI: From Principles to Practices. *ACM Comput. Surv.*, 55(9).

Li, L.; Ma, R.; Xue, Z.; and Xiong, J. 2026. Characterizing User-Reported Risks across LLM Chatbots. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI '26. New York, NY, USA: Association for Computing Machinery. ISBN 9798400722783.

Li, T.; Agarwal, Y.; and Hong, J. I. 2018. Coconut: An IDE Plugin for Developing Privacy-Friendly Apps. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(4).

Li, T.; Reiman, K.; Agarwal, Y.; Cranor, L. F.; and Hong, J. I. 2022. Understanding Challenges for Developers to Create Accurate Privacy Nutrition Labels. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22. New York, NY, USA: Association for Computing Machinery. ISBN 9781450391573.

Madaio, M. A.; Stark, L.; Wortman Vaughan, J.; and Wallach, H. 2020. Co-designing checklists to understand organizational challenges and opportunities around fairness in AI. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 1–14.

Madkour, N.; Newman, J.; Raman, D.; Jackson, K.; Murphy, E. R.; and Yuan, C. 2026. Agentic AI Risk-Management Standards Profile. Technical report, UC Berkeley Center for Long-Term Cybersecurity, Berkeley, CA. Accessed: 2026-05-19.

Miehling, E.; Ramamurthy, K. N.; Varshney, K. R.; Riemer, M.; Bouneffouf, D.; Richards, J. T.; Dhurandhar, A.; Daly, E. M.; Hind, M.; Sattigeri, P.; Wei, D.; Rawat, A.; Gajcin, J.; and Geyer, W. 2025. Agentic AI Needs a Systems Theory. ArXiv:2503.00237 [cs.AI].

Mitchell, M.; Wu, S.; Zaldivar, A.; Barnes, P.; Vasserman, L.; Hutchinson, B.; Spitzer, E.; Raji, I. D.; and Gebru, T. 2019. Model cards for model reporting. In *Proceedings of the conference on fairness, accountability, and transparency*, 220–229.

Nahar, N.; Yang, C.; Chen, Y.; Hanwen Deng, W.; Holstein, K.; Eslami, M.; and Kästner, C. 2026. "I Don't Think RAI Applies to My Model" – Engaging Non-champions with Sticky Stories for Responsible AI Work. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems*, CHI '26. New York, NY, USA: Association for Computing Machinery. ISBN 9798400722783.

Orr, W.; and Davis, J. L. 2020. Attributions of ethical responsibility by Artificial Intelligence practitioners. *Information, Communication & Society*, 23(5): 719–735.

Pan, M. Z.; Arabzadeh, N.; Cogo, R.; Zhu, Y.; Xiong, A.; Agrawal, L. A.; Mao, H.; Shen, E.; Pallerla, S.; Patel, L.; Liu, S.; Shi, T.; Liu, X.; Davis, J. Q.; Lacavalla, E.; Basile, A.; Yang, S.; Castro, P.; Kang, D.; Gonzalez, J. E.; Sen, K.; Song, D.; Stoica, I.; Zaharia, M.; and Ellis, M. 2026. Measuring Agents in Production. ArXiv:2512.04123 [cs].

Pant, A.; Hoda, R.; Spiegler, S. V.; Tantithamthavorn, C.; and Turhan, B. 2024. Ethics in the Age of AI: An Analysis of AI Practitioners' Awareness and Challenges. *ACM Trans. Softw. Eng. Methodol.*, 33(3): 80:1–80:35.

Rakova, B.; Yang, J.; Cramer, H.; and Chowdhury, R. 2021. Where Responsible AI meets Reality: Practitioner Perspectives on Enablers for Shifting Organizational Practices. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW1): 1–23.

Roth, E. 2026. OpenClaw's AI "Skill" Extensions Are a Security Nightmare. https://www.theverge.com/news/874011/openclaw-ai-skill-clawhub-extensions-security-nightmare. Published February 4, 2026. Accessed: 2026-04-25.

Ruan, Y.; Maddison, C. J.; Hashimoto, T. B.; and Liang, P. 2024. Identifying the Risks of LM Agents with an LM-Emulated Sandbox. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*.

Sanderson, C.; Douglas, D.; Lu, Q.; Schleiger, E.; Whittle, J.; Lacey, J.; Newnham, G.; Hajkowicz, S.; Robinson, C.; and Hansen, D. 2023. AI Ethics Principles in Practice: Perspectives of Designers and Developers. *IEEE Transactions on Technology and Society*, 4(2): 171–187. ArXiv:2112.07467 [cs].

Shneiderman, B. 2020. Bridging the gap between ethics and practice: guidelines for reliable, safe, and trustworthy human-centered AI systems. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 10(4): 1–31.

Shome, P.; Krishnan, S.; and Das, S. 2026. Why Johnny Can't Use Agents: Industry Aspirations vs. User Realities with AI Agents. In *ACM Conference on AI and Agentic Systems (CAIS)*.

Steenstra, I.; and Bickmore, T. 2025. A Risk Ontology for Evaluating AI-Powered Psychotherapy Virtual Agents. In *Proceedings of the 25th ACM International Conference on Intelligent Virtual Agents*, IVA '25. New York, NY, USA: Association for Computing Machinery. ISBN 9798400715082.

Tahaei, M.; Frik, A.; and Vaniea, K. 2021. Privacy champions in software teams: understanding their motivations, strategies, and challenges. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 1–15.

Thiebes, S.; Lins, S.; and Sunyaev, A. 2021. Trustworthy Artificial Intelligence. *Electronic Markets*, 31(2): 447–464.

Wang, L.; Ma, C.; Feng, X.; Zhang, Z.; Yang, H.; Zhang, J.; Chen, Z.; Tang, J.; Chen, X.; Lin, Y.; Zhao, W. X.; Wei, Z.; and Wen, J. 2024a. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6): 186345.

Wang, Z. J.; Kulkarni, C.; Wilcox, L.; Terry, M.; and Madaio, M. 2024b. Farsight: Fostering Responsible AI Awareness During AI Application Prototyping. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, 1–40. New York, NY, USA: Association for Computing Machinery. ISBN 979-8-4007-0330-0.

Winfield, A. F.; and Jirotka, M. 2018. Ethical governance is essential to building trust in robotics and artificial intelligence systems. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 376(2133): 20180085.

Winston, C.; and Just, R. 2025. A taxonomy of failures in tool-augmented llms. In *2025 IEEE/ACM International Conference on Automation of Software Test (AST)*, 125–135. IEEE.

World Economic Forum; and Capgemini. 2025. AI Agents in Action: Foundations for Evaluation and Governance. White paper, World Economic Forum, Geneva, Switzerland.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Appendix

Semi-structured Interview Protocol

Agentic AI Product Information

1. Can you briefly describe the AI agent you're working on?
2. What is your role in the team?
3. Who are the users of your AI agent?

Developers' Concerns for Their Agentic AI Products We'd like to learn more about your experience handling risks for your AI agent.

4. (For each of the three HAI principles the participant selected in the pre-study questionnaire)
   (a) Can you recall the last time you and your team discussed a specific risk related to [the HAI principle]?
      i. Why is [the risk mentioned] important to consider for your AI agent?
      ii. Do you think [the risk mentioned] emerges because this product is agentic?

Agentic AI Risk Prioritization Here are the risks that you said are relevant to your AI agent. Now, spend a minute refreshing your memory with these risks, and rank these risks in the order of their importance for this particular AI agent.

(Wait for the participant to finish the risk-ranking activity.)

Now, I'd like you to reflect on how you ranked these risks.

5. Let's start with the risk at the top, [the first risk]. Why is this risk ranked over all the other risks?
   (a) What did you consider when ranking this risk first?
6. Let's take a look at the risk at the bottom, [the last risk]. Why is this risk ranked lower than all the other risks here?
   (a) What did you consider when ranking this risk last?
7. Beyond [a summary of the factors the participant mentioned], is there anything else you considered when ranking these risks?

Bow-tie Analysis

Introduction to the bow-tie analysis diagram Look at this bow-tie shape workspace: in the center, we will put the risk that we will be analyzing. On the left side, we look at what could cause the risk; these are called "causes." On the right side, we think about what could happen after the risk happens; these are "consequences." Then, we identify controls, things we can do to either prevent the risk from happening, or reduce and recover its impact if it does. This method helps us think about the underlying factors of risks and what we can do about them.

(The participant picks one of the ranked agentic AI risks that they have attempted to mitigate, and puts the risk at the center of the bow-tie diagram.)

Risk causes Now, let's first identify the root causes of the risk. These causes can be technical or non-technical.

8. (Wait for the participant to put causes on the diagram, and for each cause, ask:)
   (a) How does this cause link to your AI agent?

Preventive controls

9. (For each cause identified, ask:)
   (a) What preventive controls have you applied, either technical or non-technical, that can prevent [the cause] from causing [the risk]?
      i. What challenges, if any, did you and your team face when applying this preventive control?
      ii. What tradeoffs, if any, did you and your team have to make when applying this preventive control?

Risk consequences Let's move on and think about potential consequences of this risk that you'd want to mitigate or reduce for your AI agent.

10. (Wait for the participant to put consequences on the diagram, and for each consequence, ask:)
    (a) Who would be impacted?
    (b) Why is the consequence crucial to consider?

Protective controls

11. (For each consequence identified, ask:)
    (a) What protective controls have you applied, either technical or non-technical, that can reduce or prevent [the consequence] from happening?
       i. What challenges, if any, did you and your team face when applying this protective control?
       ii. What tradeoffs, if any, did you and your team have to make when applying this protective control?

Closing

12. Anything else you'd like to share about addressing risks for your AI agent before we wrap this up?

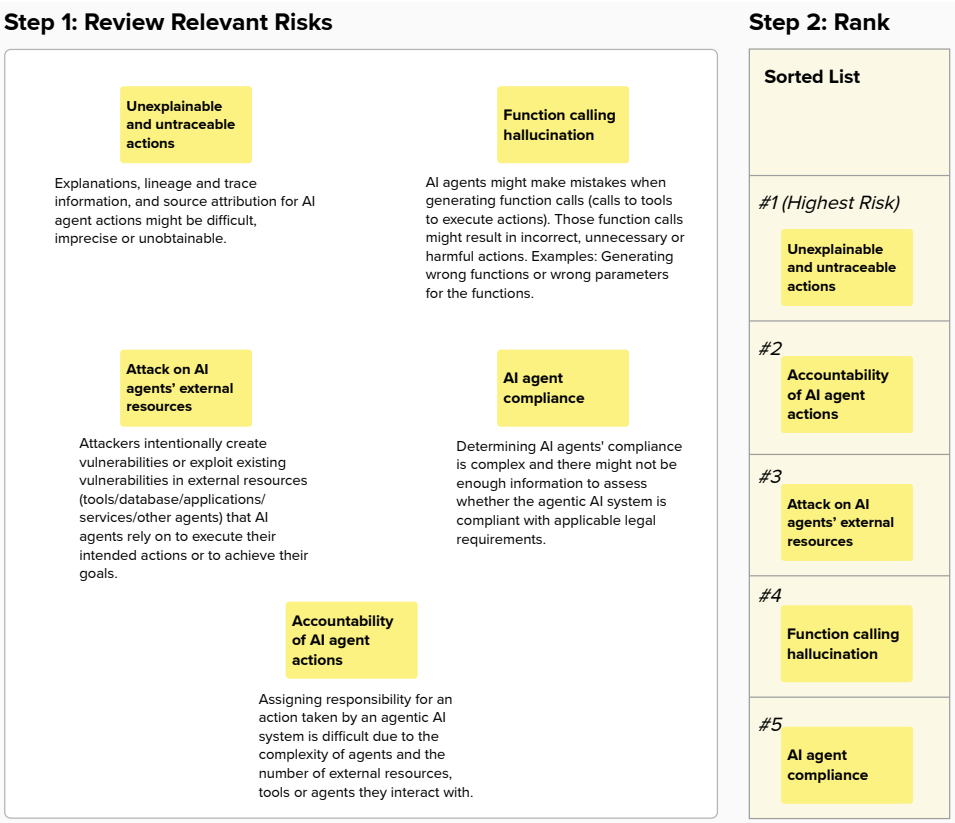

Figure 3: Example of a completed risk-ranking activity used in the interview. Participants reviewed five agentic AI risks selected from their questionnaire responses, ranked them by how important they were to address for their agentic AI product, and explained the factors that shaped their ranking.

Table 1: Agentic AI risks, grouped by human-centered AI (HAI) principle.

HAI Principle	Agentic AI Risk (Bagehorn et al. 2025)
Privacy Protection and Security	**Sharing IP/PI/confidential information with user:** AI agents with unrestricted access to resources or databases or tools could potentially store and share PI/IP/confidential information with system users when performing their actions.
	Sharing IP/PI/confidential information with tools: AI agents with unrestricted access to resources or databases or tools could potentially store and share PI/IP/confidential information with other tools or agents when performing their actions.
Reliability and Safety	**Attack on AI agents' external resources:** Attackers intentionally create vulnerabilities or exploit existing vulnerabilities in external resources (tools/database/applications/services/other agents) that AI agents rely on to execute their intended actions or to achieve their goals.
	Unauthorized use: If attackers can gain access to the AI agent and its components, they can perform actions that can have different levels of harm depending on the agent's capabilities and information it has access to. Examples: - Using stored personal information to mimic identity or impersonate with an intent to deceive. - Manipulating AI agent's behavior via feedback to the AI agent or corrupting its memory to change its behavior. - Manipulating the problem description or the goal to get the AI agent to behave badly or run harmful commands.
	Exploit trust mismatch: Attackers might initiate injection attacks to bypass the trust boundary, which is a distinct point or conceptual line where the level of trust in a system, application or network changes. Background execution in multi-agent environments increases the risk of covert channels if input/output validation is weak.
	Function calling hallucination: AI agents might make mistakes when generating function calls (calls to tools to execute actions). Those function calls might result in incorrect, unnecessary or harmful actions. Examples: Generating wrong functions or wrong parameters for the functions.
Transparency and Explainability	**Lack of AI agent transparency:** Lack of AI agent transparency is due to insufficient documentation of the AI agent design, development, evaluation process, absence of insights into the inner workings of the AI agent, and interaction with other agents/tools/resources.
	Unexplainable and untraceable actions: Explanations, lineage and trace information, and source attribution for AI agent actions might be difficult, imprecise or unobtainable.
Fairness	**Introduce data bias:** Specific actions taken by the AI agent, such as modifying a dataset or a database, can introduce bias in the resource that gets used by others or by itself to take actions.
	Discriminatory actions: AI agents can take actions where one group of humans is unfairly advantaged over another due to the decisions of the model. This may be caused by AI agents' biased actions that impact the world, in the resources consulted, and in the resource selection process. For example, an AI agent can generate code that can be biased.
Performance and Testing	**Incomplete AI agent evaluation:** Evaluating the performance or accuracy or an agent is difficult because of system complexity and open-endedness.
	Redundant actions: AI agents can execute actions that are not needed for achieving the goal. In an extreme case, AI agents might enter a cycle of executing the same actions repeatedly without any progress. This could happen because of unexpected conditions in the environment, the AI agent's failure to reflect on its action, AI agent reasoning and planning errors or the AI agent's lack of knowledge about the problem.
	Reproducibility: Replicating agent behavior or output can be impacted by changes or updates made to external services and tools. This impact is increased if the agent is built with generative AI.
	Mitigation and maintenance: The large number of components and dependencies that agent systems have complicates keeping them up to date and correcting problems.
Accountability	**Accountability of AI agent actions:** Assigning responsibility for an action taken by an agentic AI system is difficult due to the complexity of agents and the number of external resources, tools or agents they interact with.
	AI agent compliance: Determining AI agents' compliance is complex and there might not be enough information to assess whether the agentic AI system is compliant with applicable legal requirements.
Human-centered Values	**Over- or under-reliance on AI agents:** Reliance, that is the willingness to accept an AI agent behavior, depends on how much a user trusts that agent and what they are using it for. Over-reliance occurs when a user puts too much trust in an AI agent, accepting an AI agent's behavior even when it is likely undesired. Under-reliance is the opposite, where the user doesn't trust the AI agent but should. Increasing autonomy (to take action, select and consult resources/tools) of AI agents and the possibility of opaqueness and open-endedness increase the variability and visibility of agent behavior leading to difficulty in calibrating trust and possibly contributing to both over- and under-reliance.
	Misaligned actions: AI agents can take actions that are not aligned with relevant human values, ethical considerations, guidelines and policies. Misaligned actions can occur in different ways such as: - Applying learned goals inappropriately to new or unforeseen situations. - Using AI agents for a purpose/goals that are beyond their intended use. - Selecting resources or tools in a biased way - Using deceptive tactics to achieve the goal by developing the capacity for scheming based on the instructions given within a specific context. - Compromising on AI agent values to work with another AI agent or tool to accomplish the task.
Human, Social and Environmental Wellbeing	**AI agents' impact on human dignity:** If human workers perceive AI agents as being better at doing the job of the human, the human can experience a decline in their self-worth and wellbeing.
	AI agents' impact on human agency: The autonomous nature of AI agents in performing tasks or taking actions might affect the individuals' ability to engage in critical thinking, make choices and act independently.
	AI agents' impact on jobs: Widespread adoption of AI agents to perform complex tasks might lead to widespread automation of roles and might lead to job displacement.
	AI agents' impact on environment: Complexity of the tasks and possibility of AI agents performing redundant actions could lead to computational inefficiencies and add to the environmental impact.

Table 2: Participant Information

PID	Job title	Application domain of the agentic AI products	Selected human-centered AI principles	Ranked agentic AI risks
P1	Technical Specialist	Sales & Marketing	Reliability and Safety Privacy Protection and Security Performance and Testing	Redundant actions Function calling hallucination Incomplete AI agent evaluation Attack on AI agents' external resources Unauthorized use
P2	Research Scientist	Information Technology	Performance and Testing Reliability and Safety Transparency and Explainability	Incomplete AI agent evaluation Function calling hallucination Redundant actions Mitigation and maintenance Exploit trust mismatch
P3	Software Developer	Research & Development	Privacy Protection and Security Reliability and Safety Performance and Testing	Sharing IP/PI/confidential information with tools Exploit trust mismatch Attack on AI agents' external resources Function calling hallucination Unauthorized use
P4	Research Scientist	Information Technology	Transparency and Explainability Performance and Testing Accountability	Incomplete AI agent evaluation Accountability of AI agent actions Unexplainable and untraceable actions Lack of AI agent transparency Redundant actions
P5	AI Engineer	Corporate Services	Reliability and Safety Transparency and Explainability Accountability	Attack on AI agents' external resources Unauthorized use Accountability of AI agent actions Lack of AI agent transparency AI agent compliance
P6	Software Developer	Software Development	Privacy Protection and Security Accountability Human-centred Values	Sharing IP/PI/confidential information with user Misaligned actions AI agent compliance Sharing IP/PI/confidential information with tools Over- or under-reliance on AI agents
P7	Software Developer	Information Technology	Performance and Testing Reliability and Safety Transparency and Explainability	Attack on AI agents' external resources Unexplainable and untraceable actions Unauthorized use Redundant actions Incomplete AI agent evaluation
P8	Software Developer	Customer Support	Performance and Testing Transparency and Explainability Accountability	Unexplainable and untraceable actions AI agent compliance Accountability of AI agent actions Incomplete AI agent evaluation Mitigation and maintenance
P9	Software Developer	Information Technology	Performance and Testing Privacy Protection and Security Reliability and Safety	Incomplete AI agent evaluation Reproducibility Sharing IP/PI/confidential information with tools Function calling hallucination Redundant actions
P10	AI Engineer	Finance & Banking	Reliability and Safety Transparency and Explainability Performance and Testing	Reproducibility Lack of AI agent transparency Unexplainable and untraceable actions Function calling hallucination Attack on AI agents' external resources
P11	Software Developer	Corporate Services	Privacy Protection and Security Fairness Transparency and Explainability	Sharing IP/PI/confidential information with tools Unexplainable and untraceable actions Lack of AI agent transparency Introduce data bias Discriminatory actions
P12	Platform Engineer	Customer Support	Transparency and Explainability Performance and Testing Reliability and Safety	Incomplete AI agent evaluation Reproducibility Unexplainable and untraceable actions Lack of AI agent transparency Function calling hallucination
P13	Software Developer	Corporate Services	Accountability Privacy Protection and Security Transparency and Explainability	Sharing IP/PI/confidential information with user Accountability of AI agent actions Sharing IP/PI/confidential information with tools Lack of AI agent transparency AI agent compliance
P14	AI Engineer	Customer Support	Human, Social and Environmental Wellbeing Performance and Testing Reliability and Safety	Unauthorized use Redundant actions Exploit trust mismatch AI agents' impact on jobs AI agents' impact on environment
P15	Technical Specialist	Legal & Compliance	Transparency and Explainability Accountability Reliability and Safety	Unexplainable and untraceable actions Accountability of AI agent actions Attack on AI agents' external resources Function calling hallucination AI agent compliance
P16	Technical Consultant	Corporate Services	Privacy Protection and Security Accountability Transparency and Explainability	Sharing IP/PI/confidential information with tools Sharing IP/PI/confidential information with user Accountability of AI agent actions Unexplainable and untraceable actions Lack of AI agent transparency
P17	Technical Solution Architect	Healthcare Services	Human, Social and Environmental Wellbeing Reliability and Safety Accountability	AI agents' impact on human agency Accountability of AI agent actions Exploit trust mismatch Unauthorized use AI agents' impact on environment
P18	Data Scientist	Legal & Compliance	Performance and Testing Transparency and Explainability Reliability and Safety	Function calling hallucination Attack on AI agents' external resources Reproducibility Unexplainable and untraceable actions Lack of AI agent transparency
P19	AI Engineer	Finance & Banking	Performance and Testing Privacy Protection and Security Transparency and Explainability	Sharing IP/PI/confidential information with user Sharing IP/PI/confidential information with tools Lack of AI agent transparency Mitigation and maintenance Incomplete AI agent evaluation
P20	Research Scientist	Sales & Marketing	Reliability and Safety Performance and Testing Privacy Protection and Security	Sharing IP/PI/confidential information with tools Reproducibility Function calling hallucination Sharing IP/PI/confidential information with user Unauthorized use

Table 3: Participant Information (continued)

PID	Job title	Application domain of the agentic AI products	Selected AI ethic principles	Ranked agentic AI risks
P21	Technical Lead	Customer Support	Performance and Testing Accountability Privacy Protection and Security	Sharing IP/PI/confidential information with tools Incomplete AI agent evaluation Accountability of AI agent actions Sharing IP/PI/confidential information with user Mitigation and maintenance
P22	AI Engineer	Customer Support	Performance and Testing Transparency and Explainability Reliability and Safety	Unexplainable and untraceable actions Unauthorized use Reproducibility Lack of AI agent transparency Mitigation and maintenance
P23	Technical Specialist	Finance & Banking	Transparency and Explainability Reliability and Safety Accountability	AI agent compliance Unexplainable and untraceable actions Accountability of AI agent actions Exploit trust mismatch Function calling hallucination
P24	AI Engineer	Customer Support	Transparency and Explainability Privacy Protection and Security Fairness	Lack of AI agent transparency Introduce data bias Sharing IP/PI/confidential information with user Sharing IP/PI/confidential information with tools Discriminatory actions
P25	AI Engineer	Information Technology	Performance and Testing Privacy Protection and Security Transparency and Explainability	Mitigation and maintenance Incomplete AI agent evaluation Lack of AI agent transparency Unexplainable and untraceable actions Sharing IP/PI/confidential information with tools
P26	Technical Solution Architect	Government & Public Sector	Human, Social and Environmental Wellbeing Reliability and Safety Accountability	Function calling hallucination Attack on AI agents' external resources AI agents' impact on human agency AI agents' impact on environment Accountability of AI agent actions
P27	Software Developer	Customer Support	Accountability Performance and Testing Privacy Protection and Security	Sharing IP/PI/confidential information with user Incomplete AI agent evaluation Accountability of AI agent actions Sharing IP/PI/confidential information with tools Redundant actions
P28	Technical Solution Architect	Corporate Services	Accountability Performance and Testing Transparency and Explainability	Lack of AI agent transparency Accountability of AI agent actions Unexplainable and untraceable actions Reproducibility Redundant actions
P29	Technical Specialist	Finance & Banking	Transparency and Explainability Performance and Testing Privacy Protection and Security	Sharing IP/PI/confidential information with user Unexplainable and untraceable actions Lack of AI agent transparency Incomplete AI agent evaluation Mitigation and maintenance
P30	Demand Strategist	Sales & Marketing	Performance and Testing Reliability and Safety Privacy Protection and Security	Redundant actions Function calling hallucination Reproducibility Attack on AI agents' external resources Mitigation and maintenance
P31	Research Scientist	Corporate Services	Reliability and Safety Performance and Testing Accountability	AI agent compliance Incomplete AI agent evaluation Accountability of AI agent actions Mitigation and maintenance Redundant actions
P32	Research Scientist	Sales & Marketing	Accountability Performance and Testing Privacy Protection and Security	Sharing IP/PI/confidential information with user Sharing IP/PI/confidential information with tools AI agent compliance Redundant actions Reproducibility
P33	AI Engineer	Government & Public Sector	Transparency and Explainability Performance and Testing Human-centred Values	Redundant actions Reproducibility Incomplete AI agent evaluation Unexplainable and untraceable actions Over- or under-reliance on AI agents
P34	Research Scientist	Information Technology	Privacy Protection and Security Performance and Testing Accountability	Sharing IP/PI/confidential information with tools Sharing IP/PI/confidential information with user Accountability of AI agent actions AI agent compliance Mitigation and maintenance
P35	Research Scientist	Information Technology	Privacy Protection and Security Reliability and Safety Accountability	Sharing IP/PI/confidential information with user AI agent compliance Sharing IP/PI/confidential information with tools Exploit trust mismatch Attack on AI agents' external resources