May 4, 2026

Building Responsible AI Agents: Design and Development Choices for Navigating Third-Party Platform Risks

By: Molly Melcher , Kimberly Culp , Zach Harned

What You Need To Know

- Platforms want their human users to be bound by their terms of service. Developers of AI agents may mitigate the risk of platforms blocking their agents by working with platforms to develop technology that surfaces terms of service to human users for acceptance and logs accepted terms.

Related Professionals

Molly Melcher
Partner · Litigation

Kimberly Culp

- Developers should consider embedding compliance guardrails, including audit logs, disclosure protocols, and escalation pathways, directly into agent design to further facilitate platforms' acceptance of their agents.
- Developers whose agents access third-party platform content should consider designing systems that respect platform terms, robots.txt directives, and licensing frameworks, as platforms are actively suing AI companies over unauthorized access.

Why Design Choices Are Legal Choices for AI Agents

Agentic AI systems act autonomously by making decisions, navigating digital environments, and transacting on third-party platforms on behalf of users or organizations, often bypassing human signoff entirely. The legal frameworks governing online contracting, payments, and consumer protection were built around humans making deliberate choices. When the actor is an AI agent built and deployed by a developer, each of those assumptions is called into question. The platforms that offer consumer-facing products and services may not want to contend with these

Counsel · Litigation

Zach Harned
Associate · Intellectual Property

Related Practices

→ Litigation

Related Industries

— AI & Machine Learning

uncertainties by allowing agentic AI onto the platform. But some of those risks may be managed through decisions made early in the product lifecycle. There are three areas where those decisions matter most: (1) contract formation workflows, (2) compliance logging and consumer-facing guardrails, and (3) platform access controls and licensing.

I. Design Contract Formation Workflows to Surface and Log Consent

Almost every third-party platform has terms of service. If the agent cannot form a binding contract on the user's behalf, the agentic AI developer faces two problematic prongs: (1) The user may not be bound by the platform's terms and so the platform may actively try to remove the agent, and (2) the platform may hold the developer directly liable for unauthorized access.

Certain design choices may help overcome these issues. Developers should consider building agents that pause and surface terms to human users for affirmative review before acceptance or, at minimum, that log and present accepted terms after the fact to the user. If agents identified themselves as automated systems to platform operators, the platform would be able to pass along its terms to the

human user through the agent or via another mechanism.

Depending on the platform and its strategy regarding agentic AI, agents who demonstrate valid consent may be more likely to be accepted by the platforms.

II. Embed Compliance Logging and Consumer Guardrails at the Design Stage

Without proper guardrails, agentic AI may present unintended risks to agentic developers.

Liability allocation among developers, platforms, and users remains unresolved, and developers should not assume that building the technology insulates them from liability. For example, liability may fall on consumers under §5 of the Federal Trade Commission Act (and other similar laws) in cases where agentic AI dynamically adapts platform messaging back to the consumer in a way that misrepresents claims on the platform; disseminates unsubstantiated product claims (e.g., inferring product characteristics from other consumer reviews when those reviews misrepresent the product); generates fake reviews; and replicates "dark patterns" such as manipulative urgency offers, hidden fees, and confusing cancellation flows. Similarly,

agentic AI may create risk to developers regarding certain antidiscrimination statutes and regulations if the agent determines that certain jobs are not appropriate for their human user because of that user's protected characteristics. Developers may want to try to pass any liability over to the platform. How these future disputes get resolved may hinge, at least in part, on the developer's documented steps to comply with applicable laws.

Moreover, legislatures are passing new AI laws that the courts have yet to interpret. For example, California's new companion chatbot law is vague in many ways, including as to the core scope of the law: What is a "companion chatbot"?

Developers should consider designing agents that act at the consumer's behest and do not extrapolate a platform's claims beyond those being made by the platform. Before deployment, developers may wish to red team their agentic AI systems, including conducting simulations and pressure testing, and they should also consider documenting their design choices to comply with uncertain legal requirements.

Because of the uncertainty regarding liability allocation, platforms may also hesitate to allow agentic AI

applications on their platform unless the developer can demonstrate sufficient guardrails have been implemented. All the guardrails above assume the agent is on the platform lawfully. The final design layer, described below, considers that issue.

III. Enforce Platform Access Controls Before the Agent's First Request

Platforms are actively suing AI companies that access content without authorization. These cases underscore that even where an agent's purpose is benign, unauthorized access by the agent may expose the developer to breach of contract, trespass, copyright, unfair competition, and Computer Fraud and Abuse Act (and related state statute) claims.

Developers should consider designing agents that parse and respect robots.txt directives, honor rate limits, identify themselves transparently, and avoid accessing password-protected accounts without their human's authorization. Indeed, even with human authorization, some platforms may still challenge the agent's legal right to access the platform. Where platform content is needed for commercial purposes, developers should consider pursuing formal licensing agreements rather than scraping. Developers may also

consider partnering with the particular platform to ensure the agentic activity enabled by the developer is acceptable to the platform.

A Design-First Compliance Checklist

Agentic AI developers who consider the platform's interest and find ways to collaborate on compliance may be the winners in the latest AI race. Developers designing for platform acceptance should consider designing their agentic AI to (1) obtain affirmative user consent to terms; (2) reflect consumer protection laws in their design, output, and actions; and (3) respect platform directions on agentic use of the platform.

Related Insights

News

Fenwick Bolsters Powerhouse Patent Litigation Practice with Addition of Two Partners in New York and Washington, DC

April 29, 2026

Publications

Federal Court Rules Communi with an AI Model About Legal S Are Not Protected by Privilege

February 24, 2026

View more related insights