Operationalizing Ethics for AI Agents: How Developers Encode Values into Repository Context Files

Christoph Treude
Singapore Management University
Singapore, Singapore
ctreude@smu.edu.sg

Sebastian Baltes
Ruprecht-Karls-Universität Heidelberg
Heidelberg, Germany
sebastian.baltes@uni-heidelberg.de

Marc Cheong
University of Melbourne
Melbourne, Australia
marc.cheong@unimelb.edu.au

Abstract

As AI coding agents become embedded in software development workflows, developers are beginning to operationalize ethical principles by encoding behavioral rules into repository-level context files for AI agents, such as `AGENTS.md` files. Rather than examining the ethics *of* AI agents in the abstract, this vision paper investigates how ethics and values are already being translated *for* AI agents into actionable instructions that shape agent behavior. Through a preliminary investigation, we find that developers are already embedding guidance related to fairness, accessibility, sustainability, tone, and privacy. These artifacts function as a developer-authored governance layer, translating abstract principles into situated, natural-language directives within development workflows. We outline a research agenda for studying this emerging practice, including how encoded values vary across communities, what governance dynamics emerge when multiple contributors negotiate these files, and whether agents reliably adhere to the constraints specified. Understanding how ethics and values are operationalized *for* AI agents is essential to ground AI governance in modern software engineering practice.

CCS Concepts

• **Software and its engineering** → **Collaboration in software development**; • **Computing methodologies** → **Intelligent agents**; • **Social and professional topics** → **Codes of ethics**.

Keywords

AI coding agents, software engineering ethics, AGENTS.md, context engineering, AI governance, responsible AI, agentic software engineering, AI agent configuration

1 Introduction

Ethical principles for AI systems, such as fairness, accountability, transparency, and safety, are widely discussed in research and policy. However, translating these principles into everyday engineering practice remains difficult. Principles might be too vague, unenforceable, or downright clash with each other [5]. Engineers may struggle to translate ethical principles into design requirements.

The increasing integration of AI coding agents into development workflows introduces a new dimension to this long-standing challenge. These agents operate with varying degrees of autonomy and human involvement. They generate code, propose refactorings, review pull requests, and interact directly with repository artifacts. Their behavior is shaped not only by model training and prompts, but also by contextual artifacts embedded in repositories.

In this paper, we argue that a new and largely unexplored phenomenon is emerging: developers are operationalizing ethics *for* AI agents by encoding behavioral constraints directly into repository-level context files. Files such as `AGENTS.md` contain instructions that specify how AI agents should behave within a project. These instructions are written by humans, interpreted by machines, and embedded in development workflows.

Prior work has explored ethics-by-design, embedded governance mechanisms, and Responsible AI engineering processes as ways to integrate ethical principles into development practice [6, 23]. These efforts have primarily targeted human developers through organizational policies and documentation artifacts. Context files introduce a distinct operational layer: machine-interpretable constraints embedded directly in repositories and interpreted by AI agents during development workflows. Ethical frameworks have never been panaceas in human endeavors; practitioners regularly face conundrums that principles alone cannot resolve. AI agents working under natural-language repository directives will face analogous complexity. This translation layer between abstract principles and concrete agent behavior constitutes a new object of study for software engineering research.

2 Background and Related Work

Our work draws on two bodies of research: the integration of ethics and values into software engineering practice, and the study of AI agents in software development. The first informs what ethical principles might be encoded in repository artifacts and how; the second informs where such encoding occurs and what agents do with it.

2.1 Ethics, values, and SE

Ethics and values have long been discussed in software engineering and in digital ethics, *writ large*. Foundational efforts such as the *ACM/IEEE Software Engineering Code of Ethics* articulate professional obligations and societal responsibilities of software engineers [10], while various other frameworks and principles articulate the same on a broader level [5, 11]. However, several authors argue that such codes provide limited actionable guidance for everyday design decisions and must be complemented by explicit ethical deliberation within development teams [9, 18], resonant with critiques of applied AI/digital/technological ethics "…in general… lack[ing] mechanisms to reinforce its own normative claims" [11].

Table 1: Examples of Ethical Operationalization in Context Files: Excerpts from Six Open-Source Repositories.

Link	Verbatim Excerpt
Abhiek187/ez-recipes-web	"When working with the user, ensure you follow all guidelines for ethical AI, such as keeping the human in the loop, taking accountability for changes, and being transparent…"
github/awesome-copilot	"Build systems that are accessible, ethical, and fair. Test for bias…" […] `test_names = ['John Smith', 'José García', 'Lakshmi Patel', 'Ahmed Hassan', '李明']` […] "Different outcomes for same qualifications but different names"
D7460N/DHCP	"Explicitly Avoid …Moral lectures or unsolicited opinions"
haxtheweb/create	"Accessible: HAX maximizes accessibility…" […] "Sustainable: Environmental (less data, lower battery usage)…and community…sustainability."
home-assistant/core	"Inclusivity: Use objective, non-discriminatory language" […] "Clarity: Write for non-native English speakers"
tmobile/magentaA11y	"Respectful, Inclusive Language…" […] "Bias-Aware and Error-Resistant…" […] "Verification-Oriented Responses…"

Beyond professional conduct and on a broad practice- and policy-level, research has explored how ethical values can be integrated into software processes. Value-Sensitive Design [6] and Value-Based Engineering [23] aim to make human values explicit in system design, while the IEEE 7000 standard proposes structured process models to address ethical concerns during system development [22]. Systematic reviews show growing interest in operationalizing human values and acceptable norms [8] in software engineering, particularly in the requirements and design phases, but reveal limited support for later lifecycle stages and implementation practices [2, 21].

Several approaches translate ethical principles into concrete development artifacts. The Responsible AI Pattern Catalogue [16] presents governance, process, and product patterns that translate ethical principles into system-level practices, including standardized reporting templates, ethical requirements, and governance structures intended for human developers and organizations. *Ethical User Stories* and related agile practices embed concerns such as fairness, accessibility, and sustainability into backlog items and sprint routines [12, 27]. Goal-oriented methods derive *Social, Legal, Ethical, Empathetic, and Cultural* (SLEEC) requirements from explicit value models [14], while recent work advocates for lightweight, proactive integration of such considerations into existing engineering workflows [15].

2.2 Agents and Agents4SE

Studies on *in silico* agents, namely the increasing need for ethical conduct in their use, design, deployment, and emergent behaviors, are not new concepts. Consider traditional agent-based social simulations: ethical issues are found to "…arise from both its practice and its organisation" [3]. This 'top-down approach' (from the practitioner's perspective) is also complemented by a similar call from the 'bottom-up approach' (from the agent's perspective), such as the operationalization of 'good' versus 'bad' behavior at agentic level [19]. Most pre-date modern generative AI.

More recent work focuses on AI coding agents specifically. A taxonomy of human–AI collaboration in software engineering characterizes how developers interact with AI tools across different roles and intensities [26]. CRAFT (comprehensive, responsible, adaptive, foundational, translational) has been proposed as a set of values for agentic SE that extends the focus of agent design beyond coding to human–AI collaboration [13]. Building on this framing, recent work identifies trustworthiness dimensions for AI software engineers spanning technical quality, transparency and accountability, epistemic humility, and societal and ethical alignment [1]. This body of work, however, addresses agent capabilities, evaluation, and proper conduct (*vis-'a-vis* their deployers) from the 'bottom-up' rather than how developers encode persistent behavioral constraints within repositories.

Recent research has begun to conceptualize repository-level context files as a distinct class of artifacts in agentic software development. An analysis of 2,303 such files ("Agent READMEs") across 1,925 repositories finds that they resemble configuration code, primarily encoding build instructions, implementation details, and architectural information, while security and performance considerations are rarely specified [4]. Another study examining 466 open-source projects reports substantial variation in how context is expressed (descriptive, prescriptive, prohibitive, explanatory, conditional), without an established content structure [20]. A broader study covering eight configuration mechanisms across 2,853 repositories identifies `AGENTS.md` as an emerging interoperable standard across tools [7]. Empirical evaluations further suggest that `AGENTS.md` can lower agent runtime and token consumption without degrading task completion [17], and the format is used in over 60,000 open-source projects.[1]

Across these bodies of work, prior research has examined professional ethics, value-based development processes, agent trustworthiness, accountability and governance of AI coding tools [25], considerations for ethics in agent-based studies, and the structure and efficiency implications of context files. While existing

[1] https://agents.md/ , accessed 2026-05-06.

Responsible AI frameworks translate ethical principles into governance processes and documentation artifacts for human developers [16], context files target AI agents directly. How ethical principles are translated into such machine-interpretable artifacts remains largely unexplored, and we address this gap by treating `AGENTS.md` as a concrete operational layer for encoding ethics in agentic software development.

3 Preliminary Investigation

To explore how developers operationalize ethics *for* AI agents, we used a combination of GitHub code search and ChatGPT web search to identify repositories containing `AGENTS.md` files with potential ethics-related guidance. We then manually inspected 25 repositories and selected six illustrative examples, presented in Table 1, to motivate the research agenda rather than to offer a comprehensive empirical account. Even in this small exploratory sample, ethical principles are not merely referenced but translated into concrete, machine-interpretable constraints.

For example, one repository instructs the agent to "follow all guidelines for ethical AI", explicitly emphasizing keeping "the human in the loop", "taking accountability for changes", and being "transparent". While this appears to translate abstract notions of oversight into interaction-level constraints, it remains unclear what concrete behavioral changes an AI agent could derive from such high-level and ambiguous directives. The instruction documents ethical intent, but it does not specify executable conditions, triggers, or enforcement mechanisms. In contrast, another repository operationalizes fairness through structured bias testing: beyond requiring systems to be "accessible, ethical, and fair", it provides explicit test data such as `['John Smith', 'José García', 'Lakshmi Patel', …]` and directs the agent to check for "different outcomes for same qualifications but different names", embedding fairness as executable evaluation logic rather than a high-level aspiration. Other repositories encode normative boundaries on agent behavior, for instance by instructing agents to "avoid moral lectures or unsolicited opinions", or by mandating "objective, non-discriminatory language" and communication suitable for non-native English speakers. Across these examples, accountability, fairness, inclusivity, and tone become machine-interpretable constraints intended to shape agent behavior.

These examples demonstrate that developers are actively selecting the ethical concerns that matter in their projects and reformulating them as machine-interpretable constraints. Fairness becomes bias testing logic, accountability becomes interaction constraints, inclusivity becomes linguistic guidance, and sustainability becomes a design requirement. Ethical commitments are therefore not simply declared at the level of principles; they are embedded directly into the instructions that are meant to shape agent behavior within development workflows. This practice is still nascent, and key questions remain: given that humans grapple with the distinction between 'doing the right thing' and 'doing things right', it is unclear how AI agents can be expected to handle such distinctions when guided only by natural-language repository directives.

4 Roadmap

The emergence of repository-level context files that encode behavioral constraints for AI agents has introduced a new dimension to discussions of ethics and AI: rather than debating the ethics *of* AI agents in the abstract, developers are beginning to operationalize ethics *for* AI agents within everyday development workflows. This development opens up a new research frontier for software engineering. A first step is large-scale empirical mapping of context files to identify which ethical categories are encoded most frequently and which remain absent. This work can reveal whether bias, accessibility, sustainability, or privacy dominate developer attention and how these emphases vary across domains and cultures. This will also reveal which value choices developers prioritize when instructing their agents, in turn reflecting the diversity of values humans uphold.

Beyond categorization, it is important to study the translation process itself. How do developers decide which ethical principles to encode? How broad (or specific) are these ethical directives in such files? How are these constraints negotiated in pull requests? How are these principles negotiated between human software engineers and AI counterparts? Mining repository histories combined with qualitative analysis can uncover the socio-technical dynamics underlying ethical operationalization.

Equally important is evaluating whether agents adhere to encoded constraints. Future work should experimentally compare agent behavior with and without context files, measuring compliance with constraints related to bias mitigation, tone, accessibility, and regulatory guidance. Such studies would show whether operationalization shapes agent behavior or merely signals intent.

Longitudinal analysis can further examine how 'encoded ethics' evolves. Ethical commitments and directives can be strengthened after incidents, relaxed under productivity pressure, or adapted to new regulations. Some might be an exercise in 'box-ticking', while others are genuine commitments to ethics, with varying degrees of practice, paralleling the trend of ethical documentation for humans [8]. Repository histories make it possible to study how such shifts unfold over time.

At the same time, it remains unclear whether how ethics are communicated to AI agents should resemble the way ethics are traditionally documented for human contributors, such as in codes of conduct or policy statements [24]. Human-oriented ethical documentation often relies on shared norms, contextual judgment, and implicit understanding. Determining whether similar forms of documentation suffice or whether fundamentally different, machine-oriented representations are required is an important direction for future research.

Future work should also look beyond natural-language context files to the full range of agent configuration artifacts. Agent skills, tool permission settings, hook scripts, and system prompt files each constitute distinct configuration layers that may encode ethical constraints in forms quite different from prose directives [7]. Whether these programmatic artifacts reproduce, complement, or contradict constraints found in natural-language files is unknown; a complete account of operationalized ethics must consider the entire configuration space agents operate within.

Developers are already operationalizing ethics *for* AI agents in current repositories. By studying how these operational constraints are encoded, interpreted, and revised, software engineering research can move beyond abstract principles to an empirical understanding of how AI governance is implemented in practice. Moving forward, these insights can be applied beyond SE to include AI agent behavior in other domains, such as business and social sciences.

Acknowledgments

We thank the organizers and participants of the scoping workshop "AI in a Fragmented World: Navigating Trade-offs Across Disciplines and Cultures", held at the Speinshart Scientific Center for AI and SuperTech (SSC) in February 2026, for discussions that shaped this work. Marc would like to acknowledge the institutional support provided by Google (as part of the *Philosophical, tech, and legal perspectives on human relationships with social robots* project at CAIDE) for funding his travel to SSC.

References

[1] Aldeida Aleti, Baishakhi Ray, Rashina Hoda, and Simin Chen. 2026. Trustworthy AI Software Engineers. *CoRR* abs/2602.06310 (2026). arXiv:2602.06310 doi:10.48550/ARXIV.2602.06310

[2] Razieh Alidoosti, Patricia Lago, Maryam Razavian, and Antony Tang. 2022. *Ethics in Software Engineering: A Systematic Literature Review.* Technical Report. Vrije Universiteit Amsterdam. Intermediate version of later scientific publication.

[3] David Anzola, Pete Barbrook-Johnson, and Nigel Gilbert. 2022. The Ethics of Agent-Based Social Simulation. *J. Artif. Soc. Soc. Simul.* 25, 4 (2022). doi:10.18564/JASSS.4907

[4] Worawalan Chatlatanagulchai, Hao Li, Yutaro Kashiwa, Brittany Reid, Kundjanasith Thonglek, Pattara Leelaprute, Arnon Rungsawang, Bundit Manaskasemsak, Bram Adams, Ahmed E. Hassan, and Hajimu Iida. 2025. Agent READMEs: An Empirical Study of Context Files for Agentic Coding. *CoRR* abs/2511.12884 (2025). arXiv:2511.12884 doi:10.48550/ARXIV.2511.12884

[5] Marc Cheong and Simon Coghlan. 2025. *Transition to digital ethics.* Chapman & Hall/CRC, Philadelphia, PA.

[6] Batya Friedman and David G Hendry. 2019. *Value Sensitive Design: Shaping Technology with Moral Imagination.* MIT Press.

[7] Matthias Galster, Seyedmoein Mohsenimofidi, Jai Lal Lulla, Muhammad Auwal Abubakar, Christoph Treude, and Sebastian Baltes. 2026. Configuring Agentic AI Coding Tools: An Exploratory Study. In *2026 3rd IEEE/ACM International Conference on AI-powered Software (AIware).* IEEE.

[8] Haoyu Gao, Mansooreh Zahedi, Christoph Treude, Sarita Rosenstock, and Marc Cheong. 2024. Documenting Ethical Considerations in Open Source AI Models. In *Proceedings of the 18th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement, ESEM 2024, Barcelona, Spain, October 24-25, 2024*, Xavier Franch, Maya Daneva, Silverio Martínez-Fernández, and Luigi Quaranta (Eds.). ACM, 177–188. doi:10.1145/3674805.3686679

[9] Jan Gogoll, Niina Zuber, Severin Kacianka, Timo Greger, Alexander Pretschner, and Julian Nida-Rümelin. 2021. Ethics in the Software Development Process: from Codes of Conduct to Ethical Deliberation. *Philosophy & Technology* 34, 4 (2021), 1085–1108. doi:10.1007/s13347-021-00451-w

[10] Donald Gotterbarn, Keith W. Miller, and Simon Rogerson. 1997. Software Engineering Code of Ethics. *Commun. ACM* 40, 11 (1997), 110–118. doi:10.1145/265684.265699

[11] Thilo Hagendorff. 2020. The Ethics of AI Ethics: An Evaluation of Guidelines. *Minds Mach.* 30, 1 (2020), 99–120. doi:10.1007/S11023-020-09517-8

[12] Erika Halme, Mamia Agbese, Jani Antikainen, Hanna-Kaisa Alanen, Marianna Jantunen, Arif Ali Khan, Kai-Kristian Kemell, Ville Vakkuri, and Pekka Abrahamsson. 2022. Ethical User Stories: Industrial Study. In *Joint Proceedings of REFSQ-2022 Workshops, Doctoral Symposium, and Posters & Tools Track colocated with the 28th International Conference on Requirements Engineering: Foundation for Software Quality (REFSQ 2022), Aston, Birmingham, UK, March 21, 2022 (CEUR Workshop Proceedings).* CEUR-WS.org. https://ceur-ws.org/Vol-3122/RE4AI-paper-1.pdf

[13] Rashina Hoda. 2025. Toward Agentic Software Engineering Beyond Code: Framing Vision, Values, and Vocabulary. *CoRR* abs/2510.19692 (2025). arXiv:2510.19692 doi:10.48550/ARXIV.2510.19692

[14] Everaldo Silva Júnior, Lina Marsso, Ricardo Caldas, Marsha Chechik, and Genaína Nunes Rodrigues. 2026. Operationalizing Human Values in the Requirements Engineering Process of Ethics-Aware Autonomous Systems. *CoRR* abs/2602.09921 (2026). arXiv:2602.09921 doi:10.48550/ARXIV.2602.09921

[15] Stefan Kapferer, Mirko Stocker, and Olaf Zimmermann. 2024. Towards responsible software engineering: combining Value-based processes, Agile practices, and green metering. In *IEEE International Symposium on Technology and Society, ISTAS 2024, Puebla, Mexico, September 18-20, 2024.* IEEE, 1–4. doi:10.1109/ISTAS61960.2024.10732097

[16] Qinghua Lu, Liming Zhu, Xiwei Xu, Jon Whittle, Didar Zowghi, and Aurelie Jacquet. 2024. Responsible AI Pattern Catalogue: A Collection of Best Practices for AI Governance and Engineering. *ACM Comput. Surv.* 56, 7 (2024), 173:1–173:35. doi:10.1145/3626234

[17] Jai Lal Lulla, Seyedmoein Mohsenimofidi, Matthias Galster, Jie M. Zhang, Sebastian Baltes, and Christoph Treude. 2026. On the Impact of AGENTS.md Files on the Efficiency of AI Coding Agents. *CoRR* abs/2601.20404 (2026). arXiv:2601.20404 doi:10.48550/ARXIV.2601.20404

[18] Yotam Lurie and Shlomo Mark. 2016. Professional Ethics of Software Engineers: An Ethical Framework. *Sci. Eng. Ethics* 22, 2 (2016), 417–434. doi:10.1007/S11948-015-9665-X

[19] Steven Mascaro, Kevin B Korb, Ann E Nicholson, and Owen Woodberry. 2010. *Evolving ethics.* Imprint Academic, Exeter, England.

[20] Seyedmoein Mohsenimofidi, Matthias Galster, Christoph Treude, and Sebastian Baltes. 2026. Context Engineering for AI Agents in Open-Source Software. In *Proceedings of the 23rd IEEE/ACM International Conference on Mining Software Repositories (MSR 2026).*

[21] Mojtaba Shahin, Waqar Hussain, Arif Nurwidyantoro, Harsha Perera, Rifat Ara Shams, John C. Grundy, and Jon Whittle. 2022. Operationalizing Human Values in Software Engineering: A Survey. *IEEE Access* 10 (2022), 75269–75295. doi:10.1109/ACCESS.2022.3190975

[22] Sarah Spiekermann. 2021. What to Expect From IEEE 7000: The First Standard for Building Ethical Systems. *IEEE Technol. Soc. Mag.* 40, 3 (2021), 99–100. doi:10.1109/MTS.2021.3104386

[23] Sarah Spiekermann. 2023. *Value-Based Engineering: A Guide to Building Ethical Technology for Humanity.* De Gruyter, Berlin, Boston. doi:10.1515/9783110793383

[24] Parastou Tourani, Bram Adams, and Alexander Serebrenik. 2017. Code of Conduct in Open Source Projects. In *IEEE 24th International Conference on Software Analysis, Evolution and Reengineering, SANER 2017, Klagenfurt, Austria, February 20-24, 2017*, Martin Pinzger, Gabriele Bavota, and Andrian Marcus (Eds.). IEEE Computer Society, 24–33. doi:10.1109/SANER.2017.7884606

[25] Christoph Treude. 2026. Accountable Agents in Software Engineering: An Analysis of Terms of Service and a Research Roadmap. In *2026 3rd IEEE/ACM International Conference on AI-powered Software (AIware).* IEEE.

[26] Christoph Treude and Marco Aurélio Gerosa. 2025. How Developers Interact with AI: A Taxonomy of Human-AI Collaboration in Software Engineering. In *IEEE/ACM Second International Conference on AI Foundation Models and Software Engineering, Forge@ICSE 2025, Ottawa, ON, Canada, April 27-28, 2025.* IEEE, 236–240. doi:10.1109/FORGE66646.2025.00033

[27] Olaf Zimmermann, Mirko Stocker, and Stefan Kapferer. 2024. Bringing ethical values into Agile software engineering. In *The Leading Role of Smart Ethics in the Digital World.* Universidad de La Rioja, 87–98.