

Out of Place in My Own Work: How Generative AI Displaces Software Engineers

Theocharis Tavantzis ✉ 🏠 

Aalborg University, Denmark

Stefano Lambiase ✉ 🏠 

Aalborg University, Denmark

Daniel Russo ✉ 🏠 

Aalborg University, Denmark

Abstract

Background. Generative AI (GenAI) makes software engineers faster, yet a growing number of them report feeling less like engineers. They describe feeling disconnected from their own code, uncertain about their skills, and uneasy about what their profession is becoming. Prior studies capture fragments of this experience in isolation, yet no shared concept or framework describes it as a whole.

Aim. Drawing on postphenomenology, Floridi's re-ontologization of the infosphere, and Feenberg's critical theory of technology, we introduce *AI-induced displacement*: the multi-dimensional way in which GenAI tools unsettle engineers' sense of craft, identity, and professional belonging.

Method. We interviewed 21 software engineers across different roles and organizations, and analyzed their accounts through thematic analysis.

Results. We identify six dimensions of displacement, concerning (i) how engineers justify and adopt AI-generated solutions, (ii) shifts in professional identity, (iii) tensions with normative and ethical standards, (iv) disruptions to individual focus and cognitive flow, (v) changing dynamics of recognition and collaboration within teams, and (vi) the erosion of intrinsic meaning and long-term professional sustainability.

Conclusions. This typology offers researchers a theoretically grounded lens for investigating the human side of GenAI adoption in software engineering, and helps practitioners design integration strategies that foreground augmentation rather than replacement.

2012 ACM Subject Classification Software and its engineering; Software and its engineering → Software creation and management; Human-centered computing

Keywords and phrases AI-induced displacement, Generative AI, Human-centered AI, Software Engineering

Digital Object Identifier 10.4230/LIPIcs.ESEM.2026.9

Category Technical Track Paper

Related Version *Full Version:* https://stefanolambiase.github.io/assets/papers/ESEM_2026_AI_Induced_Displacement.pdf

Supplementary Material *Dataset (Replication package):* <https://doi.org/10.6084/m9.figshare.31803529>

Funding This work is partially supported by the Innovation Fund Denmark for the project AI4SE1DK (4354-00006B)

Acknowledgements We would like to express our gratitude to all the participants for dedicating their time and to those who facilitated contact with them, whose support has been instrumental to this work. We acknowledge the use of Generative AI to ensure linguistic accuracy and enhance the readability of this article.



© Theocharis Tavantzis, Stefano Lambiase, and Daniel Russo;
licensed under Creative Commons License CC-BY 4.0

20th International Symposium on Empirical Software Engineering and Measurement (ESEM 2026).

Editors: Robert Feldt, Maria Paasivaara, Daniel Mendez, Stefan Wagner, and Marvin Muñoz Barón;

Article No. 9; pp. 9:1–9:20



Leibniz International Proceedings in Informatics

LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

“I am shipping faster than ever, but I no longer feel like I am really engineering. I paste prompts, get code, fix what breaks, and move on. The code works, yet it does not feel like mine. It feels like my role is shifting away from demonstrating coding skill toward something else that I am still trying to define.”

Participant P10, Senior Test Engineer

Such accounts are becoming increasingly common as Generative AI (GenAI) expands the frontiers of many domains, including Software Engineering (SE). Nearly every field has begun to experiment with GenAI tools [7], and AI-powered services are now widely adopted [39]. In SE, Large Language Models (LLMs) reshape core activities [1], including the elicitation of software requirements [4] and code development [48]. Developers report productivity gains in information search, task completion, and problem solving [36], which together suggest a profound reconfiguration of SE practice rather than a mere incremental tool upgrade.

From a philosophy of technology perspective, GenAI does not simply introduce new tools into an otherwise stable practice. Postphenomenological accounts emphasize that technologies *mediate* human–world relations: they shape how the world shows up to practitioners, and how practitioners perceive, act, and exercise judgment in their work [51, 52]. Floridi uses the term *infosphere* to describe the broader informational environment in which humans, technologies, data, and informational processes are increasingly interconnected. From this perspective, digital technologies contribute to a process of *re-ontologization*: they do not merely change the ways in which people interact with an existing informational environment, but transform the environment itself and, consequently, the nature of the entities, relations, and forms of agency that constitute it [22, 23]. For Feenberg, such technologies embody specific “technical codes” that privilege certain values (speed and automation, for example), while marginalizing others, such as craft, deliberation, and democratic control over technology’s social role [16, 17].

Seen through these lenses, GenAI adoption in SE is not a neutral insertion of assistance into existing workflows. It is a transformation of how SE work is experienced and what it means to be a software engineer. Earlier research highlights different facets of this transformation; we frame these facets as interconnected forms of displacement that evolve over time. In the present, practitioners experience behavioral and cognitive challenges, as efficiency gains coincide with over-reliance, de-skilling, and job insecurity [49]. We observe, in parallel, a form of **epistemic displacement** that emerges when, as Booyse and Scheepers [5] note, black-box algorithms limit transparency and explainability, fostering uncertainty, distrust, and potential ethical risks. **Phenomenological displacement**, in turn, takes shape through the disruption of individuals’ workflow and cognitive flow [47]. Over time, individuals may feel socially pressured to adopt GenAI in light of widely perceived productivity gains [26], may experience an erosion of their perceived skills and mastery [14], or may adopt outputs that follow poor development practices [11, 24]. We label these forms of displacement **social**, **identity**, and **normative**, respectively.

Taken together, these forms of displacement reveal a deepening misalignment between AI-mediated work and established ways of knowing, acting, and judging, which leaves engineers feeling increasingly “out of place” within their own practice. Prior research discusses these concerns and issues in isolation, yet little is known about how SE practitioners experience and interpret this transformation in their day-to-day workflow. We lack, in particular, empirically grounded studies of how AI-embedded processes reshape software engineers’ sense of agency, expertise, and belonging in their everyday practice, and of how these experiential shifts relate to the broader reconfiguration of professional identity and standards in the age of GenAI.

The objective of this study is to examine the different aspects and dimensions of this emerging misfit empirically, and introduce the notion of **AI-induced displacement** to describe a set of systematically observed experiences reported by practitioners. In the SE context, we define AI-induced displacement as *a multi-dimensional experience in which GenAI tools, as designed and deployed, make engineers feel out of place in their own work by disrupting their sense of craft, epistemic agency, professional identity, work meaning, normative standards, and social recognition*. To investigate this phenomenon, we adopt a qualitative, exploratory methodology. We conducted semi-structured interviews with SE practitioners across multiple organizations and application domains, and performed thematic analysis on the collected data.

Given the increasing calls for human-centered AI integration in SE [42], a unifying construct that captures how GenAI tools affect individuals and how these experiences in turn shape patterns of adoption and resistance is both timely and consequential. This study contributes (i) *an empirically grounded conceptualization of AI-induced displacement*, and (ii) *implications for human-centered GenAI integration*, offering a vocabulary and a lens for understanding when AI-embedded processes support meaningful SE work and when they undermine it.

The remainder of the paper is organized as follows. Section 2 reviews the literature related to AI-induced displacement. Section 3 presents the research methodology. Section 4 reports the results and answers the research question. Section 5 discusses the findings and proposes directions for future work. Section 6 examines threats to validity and Section 7 concludes our study.

2 Background and Related Work

This section presents the philosophical foundations that motivate our construct, and surveys prior work on the impact of GenAI in software engineering.

2.1 Philosophical Foundations of AI-Induced Displacement

To conceptualize AI-induced displacement, this study builds on three strands in the philosophy of technology: postphenomenology, Floridi’s digital re-ontologization of the infosphere, and Feenberg’s critical theory of technology.

Drawing on Ihde and Verbeek, postphenomenological approaches treat technologies not as neutral instruments but as *mediators* of human–world relations [51,52], co-shaping perception, action, and experience by configuring what is perceptible, actionable, and taken-for-granted in situated practice. In other words, they participate in the constitution of everyday life-worlds [52]. Applied to SE, GenAI tools do not simply assist workflows but mediate problem framing, solution discovery, and the meaningfulness of engineering work.

Floridi’s philosophy of information captures a broader ontological shift. He argues that contemporary ICTs contribute to a *re-ontologization* of the “infosphere”: they change not only what we do, but also how agency and identity are constituted in informational environments, as artificial agents participate in distributed systems of action and responsibility [22,23]. In SE, GenAI redistributes authorship, expertise, and justification across human–AI assemblages, challenging the engineer as the primary locus of decisions, code authorship, and accountability.

Feenberg’s critical theory of technology adds a socio-political lens by arguing that technological systems embody specific design logics (what Feenberg terms *technical codes*): value-laden choices that privilege certain forms of rationality over others, typically at the expense of participation, craft, and reflective judgment [16,17]. Current GenAI tools tend

to crystallize a rationality organized around speed, automation, and output volume. As this logic dominates, engineers grow estranged from both the product of their work and its professional meaning: professional judgment erodes as engineers defer to outputs they do not fully understand; moral discomfort arises when AI use conflicts with professional standards; and productivity baselines leave little room for reflective engineering work. Feenberg’s framework thus predicts that AI-mediated work changes not only how engineers work, but what that work means to them; a prediction our empirical study directly explores.

Taken together, these three pillars support an understanding of AI-induced displacement as more than a set of isolated usability or psychological issues: they motivate a unified, experiential construct capturing how GenAI can render software engineers “out of place” by reshaping how they perceive, act, justify, and are recognized as professionals. We use them not as propositions to be empirically tested, but as sensitizing concepts that define the scope of our study, directing attention to how GenAI adoption might reconfigure engineers’ identity, agency, and meaning-making.

2.2 AI in Software Engineering: Five Fragmented Dimensions

Building on these philosophical perspectives, prior work in SE and related fields can be seen as documenting fragments of a broader experiential shift. Studies typically examine specific consequences of AI adoption rather than the overall transformation of practice, often touching on displacement through: (i) individuals’ skills and identity, (ii) limited transparency and understanding, (iii) workflow disruption, (iv) conflicts with professional standards and norms, and (v) social pressures and negative feelings due to peers’ attitudes and impressions.

Impact on Transparency and Justification. A body of work examines the lack of transparency and justification in AI-generated outputs, revealing epistemic tensions between AI recommendations and developers’ established reasoning practices. The inherent opacity of many AI tools can lead developers to adopt suggestions without access to clear reasoning [45]. This in turn reduces trust in AI-embedded tools, since developers cannot predict or explain AI behavior [28]. Fleisher [21] argues that poor explanations hinder adoption, and stresses the need for explainable AI (XAI) that offers practical insight into AI decisions. Rudin [40] underscores that interpretable models can reduce negative effects of AI decisions across domains, including SE. Booyse and Scheepers [5] report mistrust and frustration caused by black-box decision-making, while Treude and Kula [50] argue that transparency alone may overwhelm developers without proper interfaces. These studies jointly highlight *epistemic* aspects of displacement, where engineers struggle to reconcile AI outputs with their own standards of justification and understanding.

Impact on Individuals’ Skills and Identity. Shukla et al. [46] examine the de-skilling of practitioners that follows AI adoption, and show how reliance on AI can shift work from active generation and sense-making toward passive validation. Through qualitative analysis, UX practitioners reported that, while AI-assisted design can improve efficiency, heavy reliance on it raises concerns about de-skilling and about the erosion of human judgment as a core professional resource [46]. Recent work examines how GenAI adoption affects professional identity when core sources of competence are challenged. Eshraghian et al. [12] show that impact varies with expertise, with stronger professional identities linked to more positive integration, while skepticism is common among less engaged users. Hermann et al. [25] find that while GenAI may boost productivity, it can threaten needs for competence, autonomy, and relatedness, potentially destabilizing professional identity. De La Cruz et al. [14] warn that over-reliance on AI may weaken developers’ sense of mastery. Together, these studies indicate *identity* displacement, where engineers struggle to maintain a coherent sense of themselves as skilled professionals.

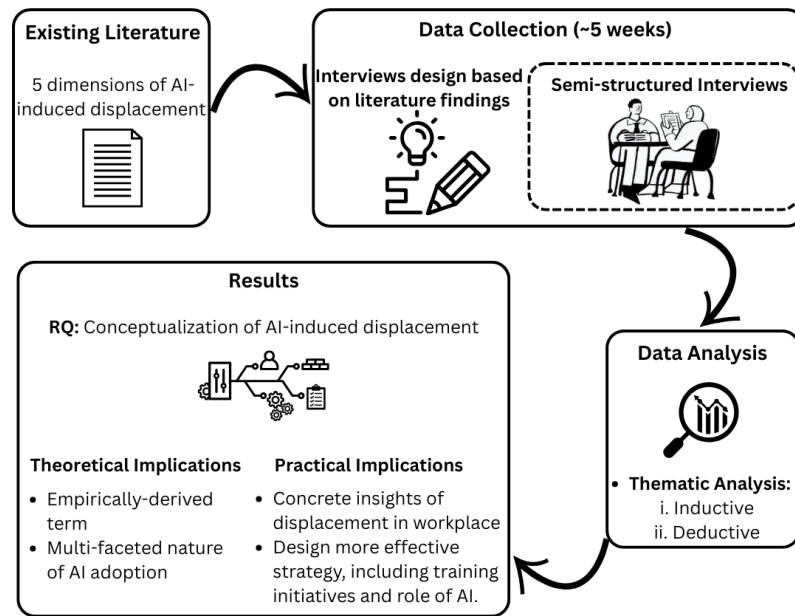
Impact on Professional Standards and Norms. Another strand of work focuses on whether GenAI outputs align with normative expectations of “good work” in SE. Early evaluations examined the capability of AI systems to produce correct and effective code to meet functional requirements [37]. Recent studies increasingly question alignment with security, style, and design standards [24, 38, 43]. Fu et al. [24] demonstrate that AI coding assistants can introduce code that conflicts with established secure development practices unless carefully reviewed. Sabra et al. [43] and Pan et al. [38] report that compliance with style guides and design patterns depends on prompting, as LLMs do not adhere to them by default. These findings point to *normative* displacement, where AI-generated code clashes with engineers’ standards of good engineering, safety, and responsibility.

Impact on Individual Flow and Work Rhythms. A line of research emphasizes workflow compatibility as a key determinant of AI adoption [9, 41], underscoring the importance of maintaining continuity and fluency in everyday work practices. Rather than adoption per se, critical decisions concern which tasks are suitable for AI support. Dell’Acqua et al. [9], for example, show that while GenAI can increase efficiency, it may also mislead skilled professionals and degrade output quality when applied to incompatible tasks. At the same time, GenAI reshapes personal workflows by shifting users from content generation to output evaluation, a transition that can interrupt established rhythms of work and undermine experienced flow [47]. Shah et al. [44] qualitatively explain that extensive manual validation and reduced trust in AI outputs further increase cognitive burden and interrupt standard work practices. This indicates that inadequate task-level integration drives workflow disruption, highlighting *phenomenological* forms of displacement where engineers experience breaks in flow and skilled coping [44, 47].

Impact on Human Relations and Social Recognition. Finally, research has begun to document social and relational consequences of GenAI adoption. Houck et al. [26] report that developers experience AI tools as augmenting their work, boosting creativity and self-perceived productivity. At the same time, not using AI may signal lower productivity in the eyes of peers, creating social pressure that shapes adoption independently of task fit or personal preference. On the other hand, Melegati et al. [35] examine negative feelings toward AI, including fear of dependency and guilt for not doing tasks oneself. Lou et al. [32] show that AI agents are shifting from tools to collaborators, reshaping roles, coordination, and accountability. Thus, they call for research on socio-behavioral aspects of teaming. Together, these works illuminate *social* forms of displacement, where recognition, expectations, and accountability are reconfigured within teams.

☰ Related Work: Summary and Research Gap.

Prior work documents five dimensions of what we term AI-induced displacement: (i) **epistemic** (what counts as valid reasons and evidence), (ii) **phenomenological** (intrusions into workflow and flow), (iii) **identity-related** (threats to competence, mastery, and professional self-conception), (iv) **normative** (tensions with ethics and standards of good work), and (v) **social** (reconfigurations of recognition, expectations, and relations). Yet, these dimensions have been treated in isolation, lacking a unified account grounded in coherent philosophical theories. This study synthesizes them into a single notion of **AI-induced displacement** and examines how software engineers experience this transformation of their work.



■ Figure 1 Research overview.

3 Research Method

The objective of this study is to examine empirically how the integration of GenAI into SE work shapes professionals' perceptions of their role, skills, and professional identity, and how these perceptions can be understood as forms of AI-induced displacement. Given the exploratory and concept-developing orientation of this work [8], we formulated a single overarching research question:

❓ **RQ:** *How do software engineers experience and interpret the integration of GenAI tools in their workflow, and how can these experiences be understood as dimensions of AI-induced displacement?*

To answer the research question, we adopted a **qualitative methodology based on semi-structured interviews**, suitable for capturing practitioners' experiences and meaning-making [8]. As shown in Figure 1, the research design comprised three phases. First, we reviewed peer-reviewed literature to identify dimensions related to displacement in AI-mediated work, which informed our interview protocol. Second, we conducted semi-structured interviews with software engineers across multiple organizations and domains. Third, we applied thematic analysis with both inductive and deductive elements to the interview transcripts, using literature-informed dimensions as a sensitizing framework while remaining open to new patterns. This process enabled us to refine the notion of **AI-induced displacement** as an empirically grounded construct.

3.1 Data Collection

The study used semi-structured interviews to collect the qualitative data. This choice was motivated by the freedom that semi-structured interviews afford the interviewer, allowing follow-up questions and the in-depth probing of emerging topics based on the interviewee's input. We designed our protocol following the guidelines of Hove and Ada [27].

We conducted 21 interviews with participants across the globe. The first session was a pilot interview to evaluate the initial protocol and ensure that the questions were clear with no ambiguities. However, the final analysis encompassed the insights gathered from all the interviews, as even the pilot session was of significant quality, offering valuable insights. The interviews were conducted online, with a median duration of 50 minutes. We observed theoretical data saturation after the 12th interview, with 4 additional codes emerging until interview 18. The remaining sessions confirmed the codes and themes we identified.

As shown in Table 1, participants were recruited globally, with Europe as the most represented region. Also, the participants hold different positions, enriching the findings based on the unique nature of each role. Even though the majority of the practitioners are mid-experienced, we included individuals with high seniority, with 7 participants having more than 15 years of working experience.

Table 2 reports the demographic information of the interviewees' organizations. Sixteen different companies are represented, which constitutes a diverse sample in terms of domain and size. The firms operate in areas ranging from automotive to banking services, and span from small companies with fewer than 10 employees to large technology firms with more than 100,000 employees. Across the included organizations, adoption relied on organizationally provisioned tools, such as company-issued Copilot licenses, while in some cases individuals described self-directed use of GenAI tools.

This research activity was approved by the Ethical Committee of Aalborg University with case number 2026-505-00766. To ensure compliance, we asked all participants for their consent to participate in our study and allow us to record and transcribe the interview session, aligned with the prescriptions of our Ethical Committee. We informed them about their right to withdraw or skip certain questions, while we provided details on the data analysis procedure.

3.2 Data Analysis

We conducted a hybrid *Thematic Analysis* to analyze the qualitative data, combining inductive and deductive coding, following the mixed approach proposed by Fereday and Muir-Cochrane [19]. We grounded our analysis in the principles outlined by Braun and Clarke [6]. This approach involves a systematic process of identifying, analyzing, and interpreting patterns across the dataset. The analysis began with familiarization through repeated reading of the data, followed by the generation of initial codes related to the scope of the study. These codes were then examined and clustered to construct candidate themes, which were iteratively reviewed and refined. In this regard, we defined and named the themes to clearly articulate their scope and focus. The final phase involved reporting the final themes in a clear and cohesive way. For the transcript analysis, we used NVivo software, a tool that enables collaborative qualitative data analysis.

Our coding consisted of two linked stages. In the first, deductive stage, we derived an initial set of sensitizing codes from the dimensions identified in prior literature on GenAI's impact on engineers, such as on their identity, and used these to guide our first pass through the transcripts. In the second, inductive stage, we allowed additional codes and sub-themes to emerge directly from participants' own accounts wherever they weren't captured by this initial set, iteratively refining and merging codes into candidate themes. The six reported dimensions result from reconciling these two threads, combining literature-derived dimensions with emergent ones, rather than restating the starting framework unchanged.

Two authors performed the analysis collaboratively to limit interpretation bias. The process was iterative, and regular analytic discussions served to examine and refine code and theme development. Differences in interpretation were used to deepen engagement with the data and to sharpen theme definitions, rather than to achieve consensus.

■ **Table 1** Participants Demographic Information.

ID	Role	Experience	Location	GenAI Tools ¹⁾
P01	DevOps Engineer	1 year	Italy	A, B, C
P02	Software Developer	4 years	Italy	A
P03	Data Scientist	3 years	Greece	A
P04	AI Engineer	1 year	Italy	A, B, C, D, E
P05	Software Engineer	3 years	Italy	A, B, C
P06	Software Developer	7 years	Sweden	F, G
P07	Software Engineer	22 years	Denmark	A, B, H
P08	Software Engineer	2 years	China	B, D, H
P09	DevOps Engineer	7 years	Sweden	F, G
P10	Test Engineer	24 years	Sweden	F, G
P11	VP of Engineering	15 years	Sweden	A, C, D
P12	Software Engineer	16 years	Denmark	A, B, C, L
P13	Data Scientist	3 years	Sweden	A, D
P14	Software Developer	5 years	Denmark	A, D, J
P15	Data Scientist	5 years	Sweden	A, B, I
P16	AI Strategist and Trainer	30 years	Denmark	B, D, K, P
P17	IT Development Manager	25 years	Denmark	K
P18	Distinguished Engineer	22 years	New Zealand	A, D, M, N, O
P19	Technical Team Lead	6 years	Netherlands	A, D, K
P20	Technical Team Lead	10 years	Denmark	D, J
P21	Senior UX Researcher	7 years	USA	B

1) A = ChatGPT; B = Gemini; C = Microsoft Copilot; D = Claude; E = Cline; F = Amazon Q Developer CLI; G = Kiro; H = Antigravity; I = Amazon Q; J = Cursor; K = GitHub Copilot; L = Llama; M = NotebookLM; N = Windsurf; O = Wispr Flow; P = OpenClaw.

4 Findings: AI-Induced Displacement

The thematic analysis of the interview data yielded six discrete themes. Five align with the dimensions identified in the Related Work Summary (Section 2), while *Meaning displacement* emerged inductively from the data. Table 3 presents the identified codes organized by displacement dimension, along with their frequency of occurrence in the interview transcripts. For reasons of space, we report only codes with frequency equal to or greater than seven, with the exception of Epistemic displacement, where we retain all three codes that emerged.

4.1 Epistemic Displacement

Epistemic displacement refers to the shift on the ways that software engineers adopt solutions, accept and justify them as valid; 3 codes emerged in this theme.

The most cited code was the shift from internal reasoning to AI-mediated heuristic verification. Practitioners today tend to rely on GenAI tools to make sense of solutions, rather than constructing the solution themselves or consulting traditional knowledge sources such as Stack Overflow. P04, for example, mentioned 🌀 “...the first thing I would do is to ask Gemini to explain that code. And then I would go on my own to read code.” According to P05, this pattern is typical when developers face programming languages with which they are unfamiliar or in which they lack prior experience.

■ **Table 2** Organizations Demographic Information.

Org. ID	Part. ID(s)	Domain	#employees
Org01	P01	Automotive	~ 30,000
Org02	P02	IT Consulting	11 – 50
Org03	P03	IT Consulting	11 – 50
Org04	P04	Soft. Development	~ 100,000
Org05	P05	IT Consulting	~ 45,000
Org06	P06, 09, 10, 13, 15	Telecommunication	~ 90,000
Org07	P07, 12	Research	501 – 1,000
Org08	P08	Soft. Development	2,001 – 5,000
Org09	P11	IT Services	11 – 50
Org10	P14	Startup Studio	< 10
Org11	P16	AI Consulting	< 10
Org12	P17	IT Services	501 – 1,000
Org13	P18	Banking Services	~ 5,000
Org14	P19	Soft. Development	101 – 200
Org15	P20	Soft. Development	51 – 200
Org16	P21	IT Services	~ 200,000

Frequently reported code was adopting AI-generated outputs without fully understanding them (when it comes to low-risk tasks). P09 conveyed *“if it is a small feature or something like one line of code or two, you know, small fixing, then it doesn’t matter, it just fixes it and then I just I leave it.”*, while P02 shared that for *“simple, UI tasks, it’s more common to adopt something, “blindly,” since it cannot cause harm”*; P13 added that tight deadlines can lead to adopting such solutions.

The least appeared code was the adoption of GenAI outputs that had non-functional weaknesses. For instance, P04 described a situation where they used GenAI to write a parser, which created latency issues. So a senior colleague consulted them to revisit the code and use a different library.

4.2 Identity Displacement

Identity displacement is defined as *the change in what it means to be a software engineer, which skills they need to possess, and how they perceive themselves in the GenAI era*; in this theme, 14 distinct codes emerged.

A considerable share of the codes concentrate on the shift in the engineer’s role within the software life cycle, and on the skills that are useful for software engineers. Almost all interviewees agreed that the role is shifting from implementation and code development toward code reviewing. P12 noted that *“I code less than before. I guess that’s a little bit of change, because GenAI writes the code and of course I’m reviewing more because of the output I need to review.”* The shift also extends to architecture and conceptualization of solutions; P08 noted the ability to *“pivot from tedious debugging tasks to high-level architectural design, and focus on more interesting stuff”*. Less commonly, interviewees described the need to move toward orchestration, owing to agentic AI systems and the potential they offer (P20: *“...maybe I don’t feel like I can do more work, but I can delegate more work. I can run three agents at the same time”*).

■ **Table 3** AI-induced displacement: Dimensions and most frequent codes. Codes are ordered by decreasing frequency; f denotes the number of interviews in which a code occurred.

Dimension	Code
Epistemic: <i>How engineers adopt solutions and justify them as valid.</i>	Shift from internal reasoning to AI-mediated heuristic verification ($f=20$)
	Using GenAI outputs without full comprehension in low-risk tasks ($f=16$)
	Non-Functional shortcomings of AI-generated adopted code ($f=6$)
Identity: <i>The change in what it means to be a software engineer, which skills they need to possess, and how they perceive themselves in the GenAI era.</i>	Shift to code reviewing ($f=18$)
	Declined pattern recognition and creativity ($f=16$)
	Declined centrality of code skills ($f=15$)
	Shift to architecture and conceptualization ($f=15$)
	Developing capabilities in GenAI technologies ($f=13$)
	Shift to orchestration ($f=12$)
	Reasoning as a central quality in the GenAI era ($f=11$)
	Increased usefulness of emotional intelligence ($f=10$)
	Openness to change as central quality ($f=10$)
	GenAI reliance hinders learning process ($f=9$)
	Impostor phenomenon due to GenAI reliance ($f=9$)
Normative: <i>Tensions with individuals' ethics and with team and organizational standards.</i>	Mismatch between team norms and legal use ($f=12$)
	Documentation more central to team standards ($f=10$)
	GenAI ethical misuse ($f=10$)
	Normative conflict between organizational standards and real use cases ($f=7$)
Phenomenological: <i>The interruptions to individuals' cognitive flow and work routines introduced by GenAI.</i>	Give up GenAI use in the mid-workflow ($f=18$)
	Flow disruption due to trial-and-error approach ($f=17$)
	Time loss due to limited GenAI's understanding ($f=11$)
	LLMs hallucination during collaboration ($f=10$)
	Cognitive disorientation due to GenAI workflow incompatibility ($f=9$)
	Disrupt the craft flow ($f=9$)
	Workflow incompatibility of approved GenAI tools ($f=7$)
Social: <i>Change of recognition, expectations, and peer relations.</i>	Decreased mentoring, knowledge and experience-sharing ($f=14$)
	Peers' skepticism and bias towards heavily AI-generated work ($f=14$)
	Top-down pressure to adopt GenAI ($f=11$)
Meaning: <i>Changes in the experiential relationship with work: personal contribution, intrinsic motivation, moral comfort, and long-term sustainability.</i>	Increased expectations or assigned workload ($f=15$)
	Job instability ($f=14$)
	Decreased intrinsic motivation ($f=12$)
	Fear of being left behind ($f=11$)
	Perceived pressure to improve ($f=10$)
	Risk of burnout in high-productivity contexts ($f=10$)
	Decreased epistemic confidence ($f=9$)
	Increased stress due to rapid progress ($f=9$)
	Moral discomfort ($f=7$)

A second cluster of codes concerns the impact of GenAI on engineers' skills and on the skill requirements of the role. The majority of interviewees agreed that reliance on GenAI has resulted in declining pattern recognition and creativity. P19 explained that ☞ *"then you simply say 'what the AI is doing makes sense and I agree with that, so let's go with that'. But because you already have one possible direction fully thought out, it lowers the reason to be creative about potential other paths."* Coding skills, similarly, were perceived as less central to the definition of a software engineer than they had been before GenAI integration. P09 captured this decline in importance by remarking that ☞ *"I would say that coding is and will decline. I would say that at least more than 60% are not doing hard coding now like before."*

Practitioners also emphasized the skills and qualities that they believe have become more central. More than 65% of the participants defined a good software engineer as one who possesses the required capabilities in GenAI technologies. These skills encompass both hard and soft skills, including context engineering and an awareness of tools' capabilities and limitations, as P16 and P01 shared, respectively. Finally, almost half of the professionals emphasized the centrality of emotional intelligence in the GenAI era (P03: ☞ *"It became really important how you will talk with your managers, colleagues, and customers. This is what differentiates us from an AI tool now"*).

Some participants reflected on the shift in their perceptions of their own skills and professional identity. According to their narratives, many appear to experience heightened impostor feelings as a result of the increased reliance on, and capabilities of, these tools (P02: ☞ *"As a developer, I did my own work, my diplomas and certificates, but maybe sometimes you feel like I am not just good enough."*). In specific cases, practitioners such as P01 even felt inferior to GenAI ☞ *"due to the speed of completing tasks"*.

4.3 Normative Displacement

Normative displacement concerns the tensions that GenAI brings to individuals' ethics and the standards of work within the team and the organizations; 9 codes emerged.

A recurrent code concerned the mismatch between use cases that organizations allow and what is acceptable by team norms. P13 described cases in which people write emails with GenAI as a typical example of an unacceptable use case within teams, since such use affects writing style and tone and makes communication feel impersonal. A similar tension was observed between organizational policies and real-world use cases. P16, in particular, stressed that ☞ *"Certainly, the tension exists. That's my main competency as a consultant, to help them move away from this, you know, signed operating procedure that you revise once per year versus having to update it like every week or every second week, or it might get updated automatically based on new versions and models to meet the state-of-the-art"*.

A frequently cited code was related to the ethical misuse of GenAI tools. We noted three main reasons that contribute to this misuse. P01 attributed their misuse to the incompatibility of the internal GenAI tools with their tasks. On the other hand, P10 explained that the lack of clarity on what is allowed and forbidden can lead to GenAI abuse. Increased demands and expectations, together with tight deadlines, constituted the third reason that motivated ethical abuse of GenAI tools, as communicated by P03.

Lastly, professionals shared how GenAI affects conventional work practices from different angles. Documentation became more central in team standards due to the capabilities of GenAI tools in explaining code snippets accurately and creating high-quality reports. P08 noted that ☞ *"it is considered as a standard practice for every piece of code to be fully documented"*. 4 interviewees conveyed their concerns about the existing hiring and evaluation

processes. Since GenAI tools are widely adopted, even in university studies, firms need to identify which skills are essential and adjust their hiring processes to evaluate candidates based on these skills.

4.4 Phenomenological Displacement

Phenomenological displacement captures *the interruptions to individuals' cognitive flow and work routines introduced by GenAI*; we recognized 9 codes in this thematic area.

Interviewees discussed several aspects associated with the intrusions they experience as GenAI is incorporated into their work. One of the most frequent codes was flow disruption arising from a trial-and-error approach, with P06 explaining that, in ☞ *"vibe coding sessions, you usually have to give it multiple answers before it actually gets on the right track"*. A code that appeared frequently alongside this one concerned the time loss caused by the limited understanding that GenAI tools display (P11: ☞ *"I try as a next step to add more functionality, and then I notice that it deleted some of the old one. When I ask it to re-add the deleted functionality, it didn't understand and added something completely different."*).

A further source of workflow interruption is LLM hallucination. P13 described a case in which the team had to work with a library that is uncommon in public repositories. The result was code that did not work and that caused confusion and time loss. P19, among others, observed that waiting time imposed by agentic AI work itself produces disruption: ☞ *"also the distraction part is that you are no longer putting 100% of your attention to the actual work that you are doing, and you're trying to offload it pretty much like you would have a junior developer sitting next to you... And this is going well, and they give you the right answers back. But what do you do in the meantime?"*

Workflow incompatibility appeared as a factor that can divert professionals' attention from their actual work. P01, in this regard, observed that ☞ *"the provided license of Copilot is very bad. For something reliable I cannot use it."* This workflow incompatibility, as P02 explained, induces cognitive disorientation. P02 noted in particular that ☞ *"It really breaks the chain of thoughts you have. Maybe you have the idea in mind, and you're working on it. At the same time, you copy-paste the code to the LLM, and that doesn't respond, and then you feel like you lose the concentration."*

A common result is giving up GenAI use in the middle of the workflow and starting to work manually from scratch. P21 shared their experience that sometimes when using GenAI to generate a report, you "pay" a higher cost since the interaction is not useful and you end up doing it manually. P09 communicated a more emotionally negative experience by saying ☞ *"...but it is sometimes that it just doesn't understand what I say. Then I get angry and OK, I say I need to do it myself!"*.

4.5 Social Displacement

Social displacement is defined as *the reconfigurations caused by GenAI of recognition, expectations, and relations between colleagues*; we identified 11 codes for this theme.

The interviewees broached various forms of social displacement introduced by GenAI. Regarding the impact on professional relations, the majority of practitioners agreed that GenAI incorporation reduced the mentoring between junior and senior people, and the knowledge-sharing overall. P07 justified this fact with the increasing capabilities of these tools (☞ *"I think there were much more questions and such interactions back in the day. Now it's like they can just find out themselves."*). Few professionals highlighted a negative effect in specific activities, such as pair-programming, meetings, and code reviews. The former, particularly, was the most frequent of the 3, as P01, P04, and P12 stressed is a thing of the past.

Reconfigurations were observed within the perceptions of individuals too. The most frequent code in this type of social displacement refers to peer skepticism and bias towards work that was produced heavily relying on GenAI. Particularly, 14 interviewees agreed that people are not willing to trust GenAI; hence, they would be stricter in reviewing work that was produced largely with the help of GenAI. Fewer, but still present, were the cases that individuals perceived others as less competent or professional due to heavy reliance on GenAI. For instance, P15 conveyed that *“it’s very annoying when they are generating large language outputs that don’t add any real value. That is just unprofessional”*.

Interestingly, participants raised two kinds of GenAI adoption pressure within their narratives. 11 practitioners talked about pressure from the high-level management to integrate GenAI into their daily workflow. P08 provided a case where *“employees” integration is regulated by monthly KPIs with specific token consumption targets, and the department even maintains an employee usage leaderboard to rank our activity levels!*”. This indicates the monitoring and pressure that individuals feel to adopt technologies that might not be applicable in certain situations. The other type of pressure relates to the one occurred by peers. So, P10 elaborated on how individuals can feel the pressure from their peers, since *“you see that other colleagues are actually going a step further in the usage of AI and then you realize that maybe you are missing something and need to catch up”*.

4.6 Meaning Displacement

Meaning displacement refers to the changes in an engineer’s experiential relationship with the work: the sense of personal contribution the work affords, the intrinsic motivation it sustains, the moral comfort with which it can be performed, and the conditions under which professional practice remains personally sustainable over time. Feenberg’s concept of alienation, understood as the estrangement of workers from the meaning and product of their labor, is the precise theoretical antecedent of this dimension; we adopt the term *meaning displacement* rather than alienation to foreground the experiential content reported by practitioners and to maintain terminological distance from the professional register already occupied by identity displacement. Seventeen codes emerged in this theme.

The most recurrent codes concerned escalating expectations and the fear of professional obsolescence. Practitioners reported that GenAI does not simply reduce workload; it raises the productivity baseline against which they are evaluated. P10 noted plainly that *“if management expects to produce 30% more with AI, then the expectations on you are to produce 30% more”*. P07 warned that this trajectory carries structural consequences: *“as time goes, people are going to expect much more from you to be done at the same time frame. That’s where it can become stressful and potentially create more problems, like burn-outs”*. Fear of professional obsolescence pervaded the accounts, with P07 describing adoption as existentially compulsory: *“people start more and more adopting, and this is inevitable right? So, we have to do this. Otherwise, we are just going to be replaced by somebody else who does”*. P10 extended this into temporal pressure, describing *“this permanent feeling that I need to catch up every day, and this is the first time ever that we have this”*.

The erosion of intrinsic meaning and professional self-worth emerged with comparable frequency. Practitioners reported that delegating core tasks to GenAI hollows out the sense of personal contribution. P11 pointed to the motivational consequences of removing coding from the workflow, while P02 captured the affective dimension: *“saying that you solve the task, relying only on the AI makes you feel a bit less worth than what you actually are”*. Several accounts accompanied this with nostalgia for pre-GenAI craft work and a distorted perception of one’s own output, with P10 questioning *“maybe I didn’t do that much,*

because the AI was doing most of the hard work". Job instability reinforced this sense of diminished standing, with P03 describing engineers as *"expendable"* and P05 noting that *"enterprises are carrying jobs and replacing them with generative AI"*.

Practitioners also introduced a moral dimension to meaning displacement. Many reported discomfort and guilt when using GenAI in ways that conflicted with their professional values. P15 felt *"shamed"* for heavy AI reliance, while P01 and P02 independently used the word *"dirty"* to describe similar experiences. In several cases this escalated into explicit regret: P15 reflected *"maybe I shouldn't have done it"*, and P04 disclosed *"sometimes you have to ship fast, or maybe sometimes you are just too tired. But it is not correct. It is something that I should have never done"*.

5 Discussion

This section discusses the significance of our findings for academia and practice, along with directions for future studies.

5.1 Findings Interpretation and Theoretical Implications

Our findings contribute theoretically to the AI adoption literature by offering interpretations that confirm and extend existing research.

5.1.1 A Unified View of the Impact of Generative AI on Developers Displacement

The primary contribution of our study is to provide a unified view of the concept of AI-induced displacement, which has previously been treated in the literature in fragmented and loosely connected ways (e.g., [24, 40, 46]), and to articulate it through a set of distinct dimensions. As with any theoretical contribution, this conceptualization offers a shared vocabulary and a coherent definition that can guide a more concrete research agenda going forward, and that helps to reduce fragmentation in the emerging body of knowledge on human-centered GenAI adoption in SE.

In addition, the articulation of AI-induced displacement into multiple dimensions enables future research directions, some of which we are already pursuing, aimed at developing measurement instruments for the phenomenon. Such instruments would allow researchers to (1) investigate individual dimensions of displacement through quantitative approaches and (2) assess these issues in practice, thereby supporting the design of informed interventions and integration strategies.

5.1.2 Compatibility as a Central Actor

Prior studies have underscored the importance of GenAI tools' compatibility in adopting these tools [31, 41]. Our findings showed that compatibility is also a main driver of displacement, as it appears in different forms. For example, many interviewees agreed that incompatibility of approved tools resulted in cognitive confusion and concentration loss, as a form of phenomenological displacement. Similarly, this misfit of provided tools leads professionals to ethical misuse of GenAI, using external solutions that are not allowed, triggering moral discomfort.

5.1.3 The path forward: a necessary shift or a misaligned trajectory?

The significant majority of the interviewees acknowledged the potential of GenAI tools and the resulting shift of their role toward higher levels of the development workflow. In particular, the transition toward code reviewing emerged as the most frequent form of identity displacement. Other commonly reported shifts included architectural design and conceptualization, reported by 15 participants, as well as the orchestration of agentic systems ($f = 12$). These insights align with emerging discussions in the research community [13, 29], which suggest that future competencies in SE may increasingly revolve around higher-level tasks, such as architectural thinking, orchestration, and mentoring of AI agents.

However, these results raise an important question regarding the direction that the industry should pursue. While it may appear straightforward for researchers to argue that a transition toward higher-level tasks is both necessary and inevitable, prior research consistently shows that, especially among junior developers, writing code, experimenting with new technologies, and designing algorithmic solutions (in short, “get one’s hands dirty”) are the activities perceived as most engaging and enjoyable [34]. These activities are often preferred over higher-level tasks such as code review or managerial coordination [2, 20].

This observation becomes even more distinct when viewed through a historical lens. Disruptive tools in software development have typically produced their greatest benefits not by automating all tasks or the technically low-level ones, as personal biases might suggest, but by automating tasks that developers perceive as tedious. In doing so, these tools improved not merely productivity but efficiency, enabling developers to focus on tasks they considered meaningful and stimulating. This shift often resulted in higher enjoyment and morale among practitioners [30]. Moreover, large-scale industry evidence points in the same direction. The DORA report on the impact of GenAI in software development [10] describes a “valuable work” paradox, in which AI adoption reduces the time developers spend on the work they find valuable, while the time devoted to toilsome activities, such as meetings and bureaucracy, remains unchanged. The authors attribute such a pattern to AI accelerating precisely the tasks developers enjoy, without yet automating the tedious ones. The same report also associates higher AI adoption with lower delivery throughput and stability, driven by larger and harder-to-review batches of generated code. In line with our findings, this suggests that the current trajectory might displace engineers from the activities that sustain their engagement, while concentrating AI support where the risks remain comparatively higher.

This tension may also connect to what our findings suggest as a productivity paradox of GenAI adoption. While participants widely acknowledged productivity gains, the significant majority reported an increase in assigned tasks and overall workload ($f = 15$), with several participants ($f = 10$) explicitly warning about the risk of increased burnout due to rising expectations and demands. These concerns resonate with the findings of Feng et al. [18], who show that GenAI adoption can simultaneously increase job demands and organizational pressure, thereby elevating burnout risk. Mazeika et al. [15] further document the growing demands from senior engineers to oversee and correct AI-generated outputs. This AI remediation burden, over time, may erode engineers’ professional efficacy [33]. This suggests that burnout might be slow and cumulative, rarely visible on productivity dashboards, until it surfaces as attrition.

Closely related, a relatively frequent aspect of the *Meaning displacement* dimension concerned the decline of intrinsic motivation among practitioners ($f = 12$). Participants associated this decline with the increasing capabilities of GenAI tools and with the tasks they take over, in particular coding activities. This insight aligns with the key findings of Wu et al. [53], who show that immediate productivity gains may come at the cost of long-term psychological and motivational outcomes for practitioners.

Taken together, these findings raise a broader question: whether the ongoing shift toward higher levels of abstraction is truly necessary or desirable. The question becomes particularly relevant once we consider that, at the current state of the technology, GenAI tools applied to lower-level activities such as code generation still frequently introduce vulnerabilities, bugs, and other technical issues [24, 38, 43] that can lead to long-term problems such as technical debt [3]. Paradoxically, the industry narrative often encourages engineers to step away from the activities in which they find the most engagement, while relying on AI for tasks in which the risks remain comparatively higher.

5.2 Implications for Practice

From a practical standpoint, we offer the following suggestions for practitioners:

- ✎ **Practitioners should monitor developer displacement.** The proposed notion of AI-induced displacement can serve as a diagnostic instrument for managers, allowing them to assess how employees experience GenAI integration beyond productivity metrics. In addition, managers should take the different dimensions into account in order to make informed and specialized decisions.
- ✎ **Companies should carefully consider the shift in tasks.** Our findings discuss the skills that practitioners consider central in the GenAI era. In this respect, the results indicate that SE curricula may need to shift toward skills such as critical AI oversight and architecture-level thinking. This shift, however, clashes with the preferences of younger developers; rather than immediately modifying activities, we suggest first discussing the actual preferences within teams. Increased productivity in the short term can mean a drastic reduction in efficiency in the long term.
- ✎ **Compatibility of AI tools is crucial and should always be a priority.** AI-induced displacement is largely influenced by the compatibility of the approved tools. Rather than adopting the best tool on paper, we recommend dialogue and greater customization of tools to make developers feel comfortable and more effective in their work.

5.3 Future Research Directions

Beyond its practical contributions, our study provides a foundation for future research:

- 📖 **Researchers should use and further validate the proposed notion.** Starting from our dimensions, researchers can (1) better contextualize their work, (2) expand in terms of dimensions, and (3) conduct more structured work, building on top of our notion.
- 📖 **Conducting longitudinal investigations of AI-induced displacement.** Since the progress of GenAI technologies is rapid, we acknowledge that displacement is unlikely to be static. Future studies could examine how AI-induced displacement evolves over time, and whether initial disruption leads to adaptation, resistance, identity reconstruction, or skill transformation.
- 📖 **Linking displacement to concrete outcomes.** The present work provides a theoretical foundation for approaching and comprehending displacement in SE caused by AI. Future research could quantitatively examine the downstream consequences of AI-induced displacement on individual well-being, motivation, performance, and career trajectories.

6 Threats to validity

This section outlines the threats to the study's validity and how we mitigated them.

Construct validity. The conceptual framing of the proposed notion of AI-induced displacement poses a potential threat to construct validity, since the term is not yet established in the literature and may appear in forms we overlooked. We conducted broad database searches to capture multiple angles of displacement. In addition, the wording of the interview questions could itself introduce bias into the responses. We followed established protocols, reviewed the questions as a team, and conducted a pilot session to minimize this risk. However, we view open-ended, exploratory instrumentation targeting unexplored areas, such as technical and organizational, as a valuable direction for future work.

External validity. The main threat concerns the transferability of our findings beyond the sample. Acknowledging that risk, we increased the interview sample despite reaching substantial theoretical saturation at twelve interviews, and we sought diversity in roles, experience, and geographic location to limit role-, experience-, or culture-specific biases.

Internal validity. Researchers' interpretations could affect thematic analysis outcomes, since coding may be influenced by prior experiences and beliefs. For this reason, two authors collaboratively conducted the coding process of the data. Although we did not compute inter-coder reliability metrics such as Cohen's Kappa, thorough analytic meetings followed the analysis to resolve differences in interpretation. Finally, the interviewees may have introduced social-desirability bias in their responses. To counter this, we assured participants of their confidentiality and thereby encouraged honest accounts of their thoughts and experiences.

7 Conclusions

This work offers a first step toward defining a construct that captures the experience of displacement among SE professionals as GenAI enters their daily work. We identify a set of dimensions that characterize this phenomenon and describe, for each dimension, the experiences reported by practitioners. These insights help clarify how software engineers undergo such forms of displacement, when and to what extent they emerge, and which consequences and underlying causes are associated with them.

Building on this foundation, we plan to extend our work through a large-scale quantitative evaluation of the presence and prevalence of these dimensions in practitioners' professional lives. Future work includes a mixed-methods study to develop a validated instrument capable of measuring the dimensions we identified. This step will enable concrete measurement of AI-induced displacement, completing the conceptual-to-operational transition of the construct, and opening the way to large-scale investigations in both industry and research settings.

Data Availability Statement

The detailed protocol, interview structure, and examples of the data analysis are available in our Figshare replication package (<https://doi.org/10.6084/m9.figshare.31803529>). We do not release the full coding spreadsheet, as participants' consent covered the use of anonymized illustrative excerpts rather than publication of the complete corpus, which could pose confidentiality risks.

References

- 1 Iftekhar et al. Ahmed. Artificial Intelligence for Software Engineering: The Journey So Far and the Road Ahead. *ACM Trans. Softw. Eng. Methodol.*, 34(5), May 2025. doi:10.1145/3719006.
- 2 Adam Alami, Victor Vadmand Jensen, and Neil A. Ernst. Accountability in Code Review: The Role of Intrinsic Drivers and the Impact of LLMs. *ACM Transactions on Software Engineering and Methodology*, 34:1–44, 2025. doi:10.1145/3721127.
- 3 Ahmed Aljohani and Hyunsook Do. PromptDebt: A Comprehensive Study of Technical Debt Across LLM Projects. In *Proceedings of the 29th International Conference on Evaluation and Assessment in Software Engineering*, pages 371–382, 2025. doi:10.1145/3756681.3756976.
- 4 Sathurshan Arulmohan, Marie-Jean Meurs, and Sébastien Mosser. Extracting Domain Models from Textual Requirements in the Era of Large Language Models. In *2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*, pages 580–587. IEEE Press, 2023. doi:10.1109/MODELS-C59198.2023.00096.
- 5 D. Booyse and C. B. Scheepers. Barriers to adopting automated organisational decision-making through the use of Artificial Intelligence. *Management Research Review*, 47(1):64–85, 2024. doi:10.1108/MRR-09-2021-0701.
- 6 Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2):77–101, 2006. doi:10.1191/1478088706qp063oa.
- 7 J. Constantz. Nearly a third of white-collar workers have tried ChatGPT or other AI programs, according to a new survey. *Time Magazine*, 2023. Retrieved on December 17th, from: <https://time.com/6248707/survey-chatgpt-ai-use-at-work/>.
- 8 J. W. Creswell and J. D. Creswell. *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches*. Sage Publications, 2017.
- 9 Fabrizio et al. Dell’Acqua. Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality. Working Paper 24-013, Harvard Business School Technology & Operations Management Unit / The Wharton School, September 2023. doi:10.2139/ssrn.4573321.
- 10 DORA Team. Impact of Generative AI in Software Development. Technical report, Google Cloud, 2025. Accessed: August 2026. URL: <https://dora.dev/ai/gen-ai-report/report/>.
- 11 Sarah D’Angelo, Ambar Murillo, Satish Chandra, and Andrew Macvean. What Do Developers Want From AI? *IEEE Software*, 41(3):11–15, 2024. doi:10.1109/MS.2024.3363538.
- 12 Farjam Eshraghian, Najmeh Hafezieh, Farveh Farivar, and Sergio de Cesare. AI in software programming: understanding emotional responses to GitHub Copilot. *Inf. Technol. People*, 38:1659–1685, 2024. doi:10.1108/ITP-01-2023-0084.
- 13 Ahmed E. Hassan et al. Agentic Software Engineering: Foundational Pillars and a Research Roadmap, 2025. doi:10.48550/arXiv.2509.06216.
- 14 Elyson De La Cruz et al. Redefining the Programmer: Human-AI Collaboration, LLMs, and Security in Modern Software Engineering. *Computers, Materials and Continua*, 85(2):3569–3582, 2025. doi:10.32604/cmc.2025.068137.
- 15 Mantas Mazeika et al. Remote labor index: Measuring ai automation of remote work, 2025. arXiv:2510.26787.
- 16 Andrew Feenberg. *Questioning Technology*. Routledge, 1999.
- 17 Andrew Feenberg. *Transforming Technology: A Critical Theory Revisited*. Oxford University Press, 2002.
- 18 Zixuan Feng, Sadia Afroz, and Anita Sarma. From Gains to Strains: Modeling Developer Burnout with GenAI Adoption, 2026. arXiv:2510.07435.
- 19 Jennifer Fereday and Eimear Muir-Cochrane. Demonstrating Rigor Using Thematic Analysis: A Hybrid Approach of Inductive and Deductive Coding and Theme Development. *International Journal of Qualitative Methods*, 5(1):80–92, 2006. doi:10.1177/160940690600500107.
- 20 Isabella Ferreira, Jinghui Cheng, and Bram Adams. The "Shut the f**k up" Phenomenon: Characterizing Incivility in Open Source Code Review Discussions. *Proceedings of the ACM on Human-Computer Interaction*, 5:1–35, 2021. doi:10.1145/3479497.

- 21 Will Fleisher. Understanding, Idealization, and Explainable AI. *Episteme*, 19(4):534–560, 2022. doi:10.1017/epi.2022.39.
- 22 Luciano Floridi. *The Onlife Manifesto: Being Human in a Hyperconnected Era*. Springer, 2014.
- 23 Luciano Floridi and J. W. Sanders. On the morality of artificial agents. *Minds and Machines*, 14(3):349–379, 2004. doi:10.1023/B:MIND.0000035461.63578.9d.
- 24 Yujia Fu, Peng Liang, Amjed Tahir, Zengyang Li, Mojtaba Shahin, Jiaxin Yu, and Jinfu Chen. Security Weaknesses of Copilot-Generated Code in GitHub Projects: An Empirical Study. *ACM Trans. Softw. Eng. Methodol.*, 34(8), October 2025. doi:10.1145/3716848.
- 25 Erik Hermann, Stefano Puntoni, and Carey Morewedge. GenAI and the psychology of work. *Trends in Cognitive Sciences*, 29, May 2025. doi:10.1016/j.tics.2025.04.009.
- 26 Brian Houck, Travis Lowdermilk, Cody Beyer, Steven Clarke, and Ben Hanrahan. The SPACE of AI: Real-World Lessons on AI’s Impact on Developers, 2025. arXiv:2508.00178.
- 27 Siw Elisabeth Hove and Bente Anda. Experiences from Conducting Semi-structured Interviews in Empirical Software Engineering Research. In *Proceedings of the 11th IEEE International Software Metrics Symposium*, METRICS ’05, page 23, USA, 2005. IEEE Computer Society. doi:10.1109/METRICS.2005.24.
- 28 Mansi Khemka and Brian Houck. Toward Effective AI Support for Developers: A survey of desires and concerns. *Commun. ACM*, 67(11):42–49, October 2024. doi:10.1145/3690928.
- 29 Karina Kohl and Luigi Carro. When Code Becomes Abundant: Redefining Software Engineering Around Orchestration and Verification, 2026. arXiv:2602.04830v1.
- 30 Stefano Lambiase, Gemma Catolino, Fabio Palomba, and Filomena Ferrucci. Motivations, challenges, best practices, and benefits for bots and conversational agents in software engineering: A multivocal literature review. *ACM Computing Surveys*, 57(4):1–37, 2024. doi:10.1145/3704806.
- 31 Stefano Lambiase, Gemma Catolino, Fabio Palomba, Filomena Ferrucci, and Daniel Russo. Investigating the Role of Cultural Values in Adopting Large Language Models for Software Engineering. *ACM Trans. Softw. Eng. Method.*, 35(1), December 2025. doi:10.1145/3725529.
- 32 Bowen Lou, Tian Lu, T. S. Raghu, and Yingjie Zhang. Unraveling Human-AI Teaming: A Review and Outlook, 2025. doi:10.48550/arXiv.2504.05755.
- 33 Fred Luthans and Carolyn M. Youssef. Human, social, and now positive psychological capital management. *Organizational Dynamics*, 33(2):143–160, 2004.
- 34 Zainab Masood, Rashina Hoda, Kelly Blincoe, and Daniela Damian. Like, dislike, or just do it? How developers approach software development tasks. *Inf. Softw. Technol.*, 150(C), 2022. doi:10.1016/j.infsof.2022.106963.
- 35 Jorge Melegati, Nicolas Nascimento, Rafael Chanin, Afonso Sales, and Igor Wiese. Exploring potential implications of intelligent tools for human aspects of software engineering. In *Proceedings of the 2024 IEEE/ACM CHASE*, pages 121–132, New York, NY, USA, 2024. Association for Computing Machinery. doi:10.1145/3641822.3641877.
- 36 Amr Mohamed, Maram Assi, and Mariam Guizani. The Impact of LLM-Assistants on Software Developer Productivity: A Systematic Literature Review, 2025. doi:10.48550/arXiv.2507.03156.
- 37 Nhan Nguyen and Sarah Nadi. An empirical evaluation of GitHub copilot’s code suggestions. In *Proceedings of the 19th International Conference on Mining Software Repositories*, MSR ’22, pages 1–5, New York, NY, USA, 2022. Association for Computing Machinery. doi:10.1145/3524842.3528470.
- 38 Zhenyu Pan, Xuefeng Song, Yunkun Wang, Rongyu Cao, Binhua Li, Yongbin Li, and Han Liu. Do Code LLMs Understand Design Patterns? In *2025 IEEE/ACM International Workshop on Large Language Models for Code (LLM4Code)*, pages 209–212, 2025. doi:10.1109/LLM4Code66737.2025.00031.
- 39 PEGA. What consumers really think about AI: A global study. <https://www.pega.com/ai-survey>, 2023. Retrieved December 20, 2025.

- 40 Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, 2019. doi:10.1038/s42256-019-0048-x.
- 41 Daniel Russo. Navigating the Complexity of Generative AI Adoption in Software Engineering. *ACM Trans. Softw. Eng. Methodol.*, 33(5), June 2024. doi:10.1145/3652154.
- 42 Daniel Russo et al. Generative AI in Software Engineering Must Be Human-Centered: The Copenhagen Manifesto. *J. Syst. Softw.*, 216(C), October 2024. doi:10.1016/j.jss.2024.112115.
- 43 Abbas Sabra, Olivier Schmitt, and Joseph Tyler. Assessing the Quality and Security of AI-Generated Code: A Quantitative Analysis, 2025. doi:10.48550/arXiv.2508.14727.
- 44 Syed Tauhid Ullah Shah, Mohammad Hussein, Ann Barcomb, and Mohammad Moshirpour. Explainability as a Compliance Requirement: What Regulated Industries Need from AI Tools for Design Artifact Generation, 2025. doi:10.48550/arXiv.2507.09220.
- 45 Md Istiak Hossain et al. Shihab. The Effects of GitHub Copilot on Computing Students' Programming Effectiveness, Efficiency, and Processes in Brownfield Coding Tasks. In *Proceedings of the 2025 ACM Conference on International Computing Education Research V.1*, ICER '25, pages 407–420, New York, NY, USA, 2025. Association for Computing Machinery. doi:10.1145/3702652.3744219.
- 46 Prakash Shukla, Phuong Bui, Sean S Levy, Max Kowalski, Ali Baigelenov, and Paul Parsons. De-skilling, Cognitive Offloading, and Misplaced Responsibilities: Potential Ironies of AI-Assisted Design. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '25, New York, NY, USA, 2025. Association for Computing Machinery. doi:10.1145/3706599.3719931.
- 47 Auste Simkute, Lev Tankelevitch, Viktor Kewenig, Ava Elizabeth Scott, Abigail Sellen, and Sean Rintel. Ironies of Generative AI: Understanding and Mitigating Productivity Loss in Human-AI Interaction. *International Journal of Human-Computer Interaction*, 41(5):2898–2919, 2025. doi:10.1080/10447318.2024.2405782.
- 48 Ze Tang, Jidong Ge, Shangqing Liu, Tingwei Zhu, Tongtong Xu, Liguang Huang, and Bin Luo. Domain Adaptive Code Completion via Language Models and Decoupled Domain Databases. In *Proceedings of the 38th IEEE/ACM International Conference on Automated Software Engineering*, ASE '23, pages 421–433. IEEE Press, 2024. doi:10.1109/ASE56229.2023.00076.
- 49 Theocharis Tavantzis and Robert Feldt. Unpacking Organizational Change in AI Transformations of Software Engineering. In *2025 IEEE/ACM 18th International Conference on Cooperative and Human Aspects of Software Engineering (CHASE)*, pages 149–160, 2025. doi:10.1109/CHASE66643.2025.00026.
- 50 Christoph Treude and Raula Gaikovina Kula. Interacting with AI Reasoning Models: Harnessing "Thoughts" for AI-Driven Software Engineering, 2025. doi:10.48550/arXiv.2503.00483.
- 51 Peter-Paul Verbeek. *Moralizing Technology: Understanding and Designing the Morality of Things*. University of Chicago Press, 2011.
- 52 Peter-Paul Verbeek. Beyond interaction: A short introduction to mediation theory. In J. van den Hoven, P. Vermaas, and I. van de Poel, editors, *Designers and Users: Perspectives on User Involvement in Design*, pages 41–48. Springer, 2015.
- 53 S. Wu, Y. Liu, M. Ruan, et al. Human-generative AI collaboration enhances task performance but undermines human's intrinsic motivation. *Scientific Reports*, 15:15105, 2025. doi:10.1038/s41598-025-98385-2.