International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(18s) 2026 pp. 1174-1192
https://doi.org/10.70917/ijcisim-2026-4969
Article

AI-Agent Use and Job Satisfaction among Software Developers: Evidence on Coordination and Implementation Friction

Zile Fan[1]* , Chandra Mohan Vasudeva Panicker[1]

[1] School of Business & Management, Lincoln University College, Malaysia
* Corresponding author: full121314@163.com

Abstract: AI agents can plan and execute multi-step work, yet little is known about how their use is associated with employees' job satisfaction. Drawing on research on agentic information systems and job demands–resources theory, this study distinguishes adoption frequency from the resources and demands reported by current users. Data come from the publicly released 2025 Stack Overflow Developer Survey. The adoption analysis includes 22,289 employed professional developers across 165 countries, and the within-user analysis includes 5,988 current agent users across 141 countries. We estimate ordinary least-squares models with standard errors clustered by country and assess robustness using a cross-fitted doubly robust adjusted contrast, inverse-probability weighting for complete-case selection, a binary satisfaction outcome, and false-discovery-rate correction. Compared with respondents who neither used nor planned to use agents, weekly and daily use are associated with job-satisfaction scores that are 0.175 and 0.239 points higher, respectively, on the 0–10 scale; lighter use and copilot-only use are not statistically distinguishable from the reference group. The doubly robust adjusted contrast is 0.145 points (95% CI [0.071, 0.219]). Among current users, perceived improvement in team coordination is the most consistent positive correlate of satisfaction. Implementation friction is negatively associated with satisfaction, whereas the negative association for organizational restrictions is less stable across robustness analyses. Efficiency is no longer independently associated with satisfaction when demands are modeled jointly; capability perceptions and use intensity are also null. Taken together, the results point to implementation quality rather than use intensity alone. Because the survey is cross-sectional and agent use is self-selected, all estimates are associational.
Keywords: AI agents; job satisfaction; software developers; job demands–resources theory; human–AI collaboration; implementation friction; open data

1. Introduction

Generative artificial intelligence (AI) is beginning to move from conversational assistance to agentic systems that can decompose goals, call tools, modify artifacts, and continue with limited prompting. For organizations, this is more than an increase in output speed. Delegation changes who initiates actions, how errors travel through workflows, where accountability sits, and what coordination remains for employees. Agentic information-systems theory accordingly treats use as reciprocal delegation between a person and an artifact, rather than one-way consumption of a passive tool (Baird & Maruping, 2021). The managerial question is whether these systems create usable resources without shifting disproportionate verification, integration, and governance work to employees (Berente et al., 2021; Raisch & Krakowski, 2021).

Experimental evidence documents clear task-level gains in several settings. Generative AI can shorten professional writing tasks and raise evaluated quality (Noy & Zhang, 2023), while a large workplace study reports meaningful productivity gains for customer-support agents, especially less experienced workers (Brynjolfsson et al., 2025). In software development, three recent field experiments involving 4,867 developers find a 26.08% increase in completed tasks from AI coding assistance (Cui et al., 2026). Together, these studies show that AI can serve as a task resource in the settings examined, but they do not establish whether more frequent use is associated with employees'

broader experience of work. Output gains may coexist with monitoring, loss of control, unreliable suggestions, new learning obligations, security risks, or workflow disruption.

Research on human–AI collaboration further shows why productivity and work experience should be analyzed separately. A meta-analysis of 106 experiments finds that human–AI combinations do not consistently outperform the stronger party working alone, and that complementarity varies with task and collaboration design (Vaccaro et al., 2024). Field experiments similarly document a 'jagged' technological frontier within which AI can improve performance but outside which it can increase error (Dell'Acqua et al., 2026b). In programming, code generation alternates between acceleration and exploration (Barke et al., 2023), accepted completions need not indicate durable code quality (Ziegler et al., 2022), and users incur costs in reading, validating, modifying, and waiting for suggestions (Mozannar et al., 2024). AI-assisted participants can even produce less secure code while remaining more confident in its security (Perry et al., 2023). Job satisfaction should therefore be examined as an outcome distinct from task performance rather than assumed to follow from productivity gains.

Employee-level research points to a similarly double-edged relationship. Generative-AI collaboration may provide useful work resources while also intensifying alienation, expedient behavior, technostress, or uncertainty (Hai et al., 2025; Högemann et al., 2025). Recent JD–R research links AI affordances and organizational support to innovative performance while recognizing workload and literacy demands (Li et al., 2026). Even so, reviews of AI in working life find fragmented measures, rapidly changing systems, and little comparable evidence across occupations and countries (Savolainen et al., 2026). The evidence base is thinner still for AI agents. Most studies concern general generative AI or code completion, not systems presented to users as autonomous entities that operate with minimal intervention; outcomes also tend to focus on task output, adoption intention, or trust rather than satisfaction with the professional role as a whole.

Against this background, we combine agentic information-systems theory with job demands–resources (JD–R) theory and separate two questions that are often conflated. The first concerns adoption: how is the frequency of agent use associated with job satisfaction relative to nonuse, planned adoption, or copilot-only use? The second concerns implementation: among current users, which reported resources and demands retain an association with satisfaction when considered together? The resource domains are perceived efficiency, capability development, and team coordination; the demand domains are verification, implementation, and organizational restriction. This two-layer design treats adoption as the start of implementation rather than its full description and does not label contemporaneous perceptions as mediators.

The analysis uses the public-use file from the 2025 Stack Overflow Developer Survey. After the stated eligibility filters, the adoption sample contains 22,289 employed professional developers in 165 countries, and the within-user sample contains 5,988 complete responses in 141 countries. The primary specifications use ordinary least squares with country and industry fixed effects and standard errors clustered by country. A cross-fitted doubly robust adjusted contrast, inverse-probability-of-completion weighting, a binary high-satisfaction model, and false-discovery-rate corrections assess robustness. The dataset offers unusual geographic breadth in a knowledge-intensive occupation, but its voluntary, cross-sectional design cannot identify causal effects. We therefore describe estimates as associations, correlates, or adjusted contrasts.

By separating adoption frequency from within-user experiences, the study treats agent use as a work-design configuration rather than a simple exposure variable. It distinguishes team coordination from individual efficiency and separates verification concerns from implementation and governance frictions. These distinctions connect reciprocal delegation with JD–R theory without treating cross-sectional perceptions as mediators.

Figure 1 presents the conceptual framework, Figure 2 documents sample construction, Figure 3 displays the main adjusted estimates, and Table 1 reports the operational definitions.

2. Theoretical background and hypotheses

2.1. AI agents as delegated and embedded information systems

Conventional adoption models often represent an information system as a passive object that a user chooses to employ. AI agents complicate that representation because they can pursue delegated goals, initiate intermediate steps, interact with other systems, and return outputs that require human acceptance or correction. Baird and Maruping's (2021) framework locates agentic use in reciprocal delegation: people allocate work to the artifact, while the artifact can redirect subtasks, exceptions, or requests back to the person. Management of AI therefore involves balancing autonomy with control, learning with standardization, and local experimentation with organization-wide accountability (Berente et al., 2021).

Delegation is productive only when task allocation and interface design fit human capabilities. Experiments show cognitive costs when people must decide whether to rely on an AI recommendation or reclaim the task (Fügener et al., 2022). Algorithm appreciation and aversion are not fixed user traits; they depend on the decision context, perceived agency, error visibility, and alternatives (Jussupow et al., 2024). For programming, interfaces can materially alter agent performance (Yang et al., 2024), while live execution feedback can help developers validate generated code and calibrate reliance (Ferdowsi et al., 2024). Thus, 'using an agent' encompasses a sociotechnical arrangement of delegation, inspection, workflow integration, and collective norms.

The automation–augmentation paradox supplies a complementary organizational lens. Automation can substitute for tasks while augmentation increases the value of human judgment; both processes can occur in the same implementation and reinforce one another (Raisch & Krakowski, 2021). Work-design research similarly argues that digital systems reshape autonomy, skill variety, feedback, cognitive load, and social contact rather than exerting a uniform technological effect (Parker & Grote, 2022). Reviews of algorithmic management show that the same system may enable autonomy for one group and intensify control or opacity for another (Noponen et al., 2024; Parent-Rocheleau & Parker, 2022). These perspectives imply that use frequency is an incomplete predictor of employee outcomes unless the resources and demands generated by use are also observed.

2.2. A job demands–resources account of agent use

JD–R theory defines job resources as physical, psychological, social, or organizational aspects of work that help achieve goals, reduce demands, or stimulate learning. Job demands require sustained effort and therefore carry psychological or physiological costs. Crucially, a technology is not inherently a resource or a demand: its role depends on how it changes the work system (Bakker et al., 2023). An AI agent can reduce repetitive effort, provide rapid feedback, and expand problem-solving capacity, yet it can also generate ambiguous outputs, require constant verification, force tool reconfiguration, or conflict with security rules. The theory is therefore suited to the coexistence of gains and frictions documented in AI-supported work (Kumar et al., 2024; Verma et al., 2024).

We treat overall job satisfaction as a summary evaluation of the professional role. It should respond not only to task speed but also to whether employees can exercise competence, coordinate effectively, and complete work without avoidable obstacles. Developer-specific evidence shows that autonomy, mutual support, and task characteristics are associated with both satisfaction and performance (Russo et al., 2023), while process debt in agile development places satisfaction at risk through documentation, synchronization, and infrastructure problems (Gustavsson et al., 2025). Agent-related resources and demands map naturally onto these established features of developer work.

Figure 1 previews the two-layer framework. Panel A evaluates adoption-frequency contrasts across respondents. Panel B evaluates resources and demands only among current users, because the survey asked detailed agent-impact questions conditionally. The arrows represent hypothesized associations rather than a mediation sequence.

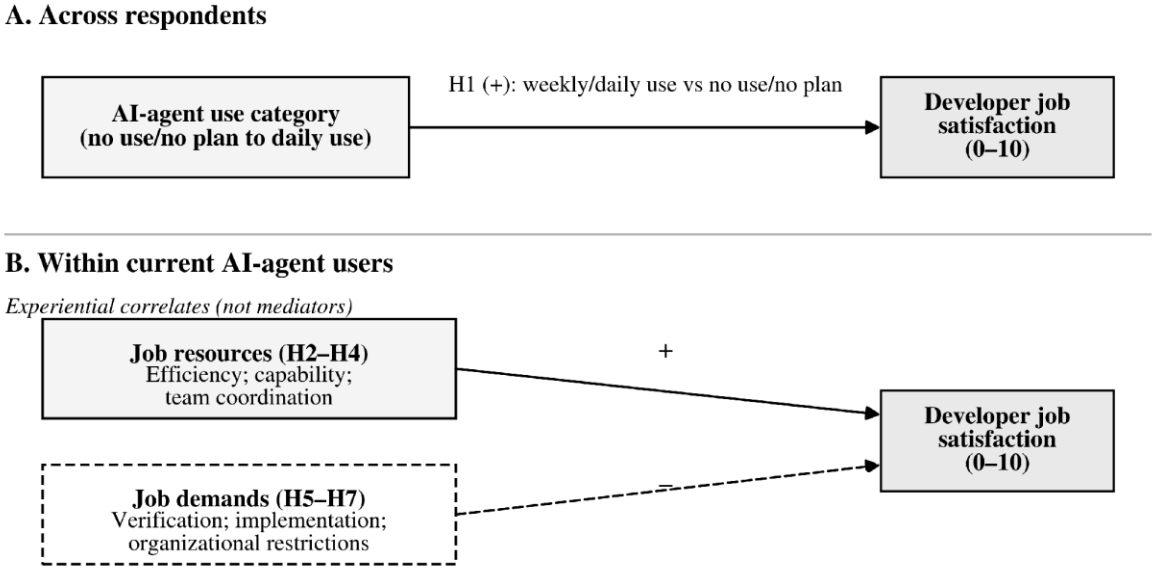

Figure 1. Two-layer conceptual framework.

Note: Panel A tests adjusted adoption-frequency contrasts in the full analytic sample; Panel B tests resource and demand correlates among current AI-agent users. Solid and dashed paths denote hypothesized positive and negative associations, respectively. All models include the covariates listed in Section 3.3; Panel B does not represent mediation. Source: Authors' conceptualization.

2.3. Adoption frequency and job satisfaction

Weekly or daily use may indicate that an agent has become integrated into consequential work, although the cross-sectional survey does not observe duration or routinization directly. Repeated delegation can reduce switching costs, support faster feedback, and help users learn when the system is reliable. Productivity experiments and workplace field studies provide a plausible resource pathway from effective use to a better work experience (Noy & Zhang, 2023; Brynjolfsson et al., 2025; Cui et al., 2026). A linear dose response is nevertheless unlikely. Occasional users may still be learning the interface, may deploy agents only for exceptional tasks, or may face constraints on more frequent use. Copilot-only users receive suggestions but do not report delegating multi-step work, while planned adopters have intention without experience. We therefore expect the clearest positive difference in the weekly and daily categories, while treating lighter categories as separate contrasts rather than intermediate doses.

H1. *Compared with respondents who neither use nor plan to use AI agents, weekly or daily AI-agent use is positively associated with developer job satisfaction.*

2.4. Experiential resources among current users

Efficiency resources capture increased productivity, reduced time on specific tasks, and automation of repetitive work. These benefits conserve effort and permit attention to shift toward more meaningful or complex activities. Large language models have produced significant gains in controlled programming tasks (Weber et al., 2024), although their advantages are weaker for integrated development than for bounded coding (W. Wang et al., 2024). Because job satisfaction reflects the whole role, the expected association should be positive but need not be large or independent of other resources.

H2. *Among current AI-agent users, perceived efficiency resources are positively associated with developer job satisfaction.*

Capability resources capture accelerated learning, more effective solution of complex problems, and improved code quality. These experiences can strengthen mastery and professional growth—important features of satisfying knowledge work. AI programming assistants are valued for syntax recall and speed but are also criticized for mismatched requirements and limited controllability (Liang et al., 2024). Consequently, capability benefits should matter when they reflect genuine learning and problem solving rather than superficial completion acceptance.

H3. *Among current AI-agent users, perceived capability-development resources are positively associated with developer job satisfaction.*

Coordination resources capture whether agents improve collaboration within the team. This is theoretically distinct from individual efficiency. Agent outputs can serve as boundary objects, accelerate explanation, and make specialized knowledge more accessible, but they can also fragment work when team members use inconsistent tools or cannot inspect one another's agent-generated changes. A recent field experiment shows that generative AI can partly reproduce the performance and social-affective contribution of teamwork in a defined setting (Dell'Acqua et al., 2026a). In developer work, where integration and review cross individual task boundaries, coordination should be especially consequential for satisfaction.

H4. *Among current AI-agent users, perceived team-coordination resources are positively associated with developer job satisfaction.*

2.5. Experiential demands among current users

Verification burden combines concern about information accuracy with concern about security and data privacy. These concerns require vigilance even when generated output appears plausible. Trust in code generation is context dependent, and developers seek quality signals, configurable controls, and validation support (R. Wang et al., 2024). Empirical work documents non-reproducible or erroneous code solutions (Moradi Dakhel et al., 2023) and security overconfidence among assisted users (Perry et al., 2023). Sustained checking may therefore consume attention and reduce satisfaction.

H5. *Among current AI-agent users, perceived verification burden is negatively associated with developer job satisfaction.*

Implementation burden captures difficulty integrating agents with existing tools and workflows, time and effort required to learn effective use, and the cost of agent platforms. These frictions are proximal obstacles to goal achievement. They resemble process debt: the employee must repair or navigate the production system before substantive work can proceed (Gustavsson et al., 2025). Studies of AI-assisted programming identify substantial costs in reading, validating, editing, and waiting (Mozannar et al., 2024), while technostress research highlights complexity, uncertainty, and support deficits (Högemann et al., 2025; Kumar et al., 2024).

H6. *Among current AI-agent users, perceived implementation burden is negatively associated with developer job satisfaction.*

Organizational restrictions capture strict IT or information-security rules that prevent use of agent tools or platforms. Such rules may be justified by legal, privacy, intellectual-property, or security responsibilities. Nevertheless, when access is blocked without usable alternatives, training, or explanation, workers can experience role conflict and reduced control. Responsible AI practices can function as supportive resources when they clarify expectations, but as hindrance demands when rules are opaque or incompatible with task requirements (Verma et al., 2024). The prediction concerns the experienced obstruction, not the legitimacy of governance itself.

H7. *Among current AI-agent users, perceived organizational restrictions are negatively associated with developer job satisfaction.*

Taken together, H2–H4 represent distinct resource domains and H5–H7 represent distinct demand domains; the hypotheses do not require every domain to be significant. Two exploratory interactions—efficiency resource ×

verification burden and coordination resource × managerial role—are examined after the main effects, but no contribution is premised on their statistical significance.

3. Method

3.1. Data source and sample construction

The study uses the 2025 Stack Overflow Developer Survey, an openly released cross-sectional survey of developers and technologists (Stack Overflow, 2025). Stack Overflow fielded the survey from 29 May to 23 June 2025 and recruited mainly through its own website, email, social channels, and in-product messages; fewer than 2% of respondents were recruited through Reddit. The official methodology explicitly notes that highly engaged Stack Overflow users were more likely to encounter the invitation. The data are therefore a voluntary convenience sample, not a probability sample of the global developer population. No survey weights are supplied. Descriptive percentages and model estimates in this article refer to respondents in the stated analytic samples.

The downloaded public-use comma-separated file contained 49,191 rows and 49,191 unique ResponseId values. The public dashboard reports 49,009 responses across 177 countries (Stack Overflow, 2025), and the accompanying documentation does not reconcile the difference. We therefore report the released-file count and base every subsequent filter on that replicable file. Eligibility required respondents to identify as developers by profession, to report employment or independent contractor/freelancer/self-employed status, and to provide a valid current-role job-satisfaction score. These criteria produced a sample of 24,949 respondents.

The adoption model additionally required a valid response to the AI-agent use question, leaving 22,289 respondents across 165 countries. The clean doubly robust contrast excluded planned adopters and copilot-only respondents so that 7,113 current agent users could be compared with 8,104 respondents who neither used nor planned to use agents (N = 15,217). Detailed impact and challenge items were asked of current users. Complete values on the six focal resource/demand measures and job satisfaction yielded 5,988 users across 141 countries. Figure 2 reports the complete flow and makes the different estimands explicit.

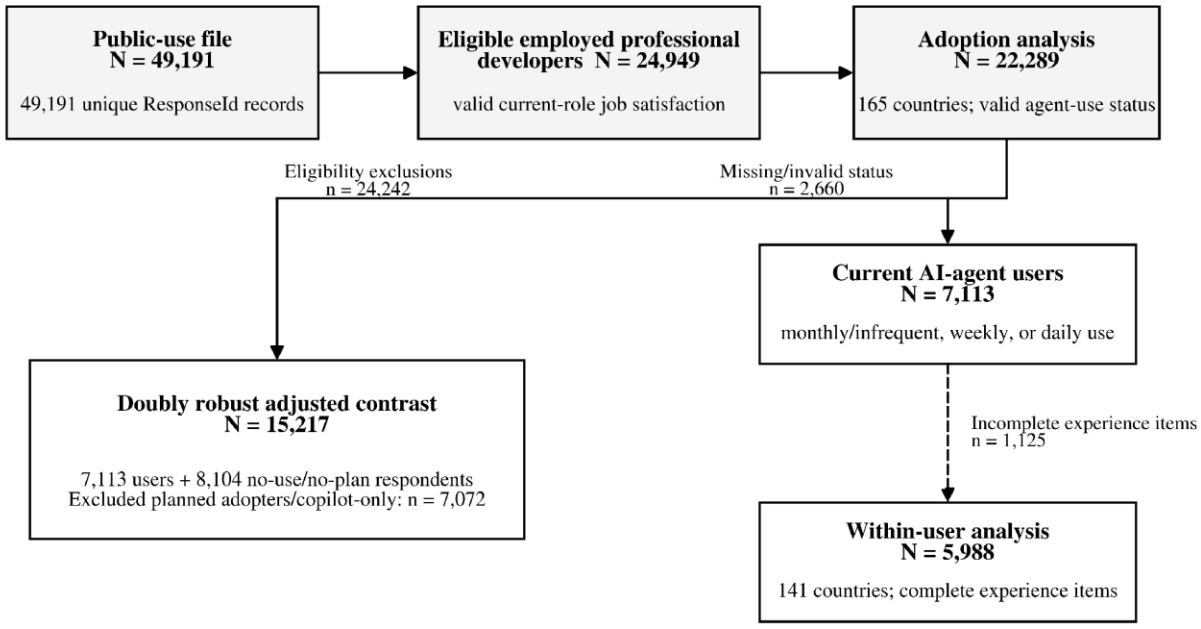

Figure 2. Analytic-sample construction from the downloadable 2025 Stack Overflow Developer Survey file.

Note: Counts are unique ResponseId records. The adoption, doubly robust, and within-user branches address distinct estimands; exclusions are shown beside each transition. Source: Authors' analysis of the public-use file.

3.2. Measures

Table 1 reports the exact operationalization. Overall job satisfaction is the response to 'How satisfied are you in your current professional developer role?' on a 0–10 scale. AI-agent status derives from the survey definition of agents as autonomous software entities that operate with minimal to no direct human intervention. We retain all six mutually exclusive responses: no use/no plan, no current use/plans adoption, copilot or autocomplete only, agent use monthly or infrequently, weekly, and daily. This coding prevents assistant-only use from being misclassified as agent adoption.

Table 1. Constructs, operationalization, and analytic role

Construct	Operationalization	Scale	Reliability/source	Role
Job satisfaction	Overall satisfaction with current professional developer role	0–10	Single survey item	Outcome
Agent-use category	No use/no plan; plans adoption; copilot only; monthly/infrequent; weekly; daily	Categorical	Survey item	H1 focal predictor
Efficiency resource	Productivity increase; reduced task time; automation of repetitive tasks (mean)	1–5	α = .817	H2 resource
Capability resource	Accelerated learning; solving complex problems; improved code quality (mean)	1–5	α = .789	H3 resource
Coordination resource	Improved collaboration within the team	1–5	Single item	H4 resource
Verification burden	Concern about accuracy; concern about security/privacy (mean)	1–5	Analytic composite	H5 demand
Implementation burden	Integration difficulty; learning effort; platform cost (mean)	1–5	Analytic composite	H6 demand
Organizational restriction	Strict company IT/InfoSec rules prevent use	1–5	Single item	H7 demand

Covariates	Experience; manager role; log annual pay; age; education; employment; organization size; remote-work mode; industry; country	Mixed	Public survey fields	Observed adjustment set

Note. Agreement items are coded 1 = strongly disagree to 5 = strongly agree. The demand measures are theory-defined analytic composites rather than reflective latent scales; their components are also modeled separately. α denotes Cronbach's alpha in the within-user sample.

Efficiency is the mean of three agreement items covering productivity, time reduction, and repetitive-task automation. Capability is the mean of accelerated learning, complex problem solving, and improved code quality. Their alpha reliabilities are .817 and .789, respectively. Coordination is a single item asking whether agents improved collaboration within the team. The seven benefit items jointly have alpha = .879, but we retain the three theoretically distinct resource domains because individual efficiency and social coordination need not move together.

The survey presents six challenges as distinct statements. We define verification burden as the mean of accuracy and security/privacy concern, implementation burden as the mean of integration difficulty, learning effort, and platform cost, and organizational restriction as the strict-rule item. The five non-rule demand items have alpha = .617. That value and the conceptual heterogeneity of the items argue against treating them as a single reflective scale. We therefore call the two multi-item measures analytic composites, interpret them as domains rather than latent factors, and repeat the preferred model using all six challenge items separately.

The adjustment set includes work experience in years, individual-contributor versus people-manager role, the natural logarithm of annual compensation converted to U.S. dollars, age, education, employment type, organization size, remote-work mode, industry, and country. Continuous missing values are median-imputed with missingness indicators; categorical missingness is retained as an explicit level. We exclude ToolCountWork from preferred models because contemporaneous tool use could itself respond to agent adoption. It is added in a sensitivity model. We also do not enter a derived remote-flexibility indicator together with RemoteWork categories, thereby avoiding deterministic multicollinearity.

3.3. Analytical strategy

The first model estimates adjusted differences in the 0–10 satisfaction score across the five agent-status categories relative to no use/no plan. The model includes industry and country fixed effects, and standard errors are clustered by country. The estimating equation is:

$$JS_i = \alpha + \Sigma\beta_g AgentGroup_{ig} + \gamma^T X_i + \delta_{c(i)} + \lambda_{s(i)} + \varepsilon_i. \quad (1)$$

where JS_i is job satisfaction for respondent i, $AgentGroup_{ig}$ denotes the five non-reference agent categories, and X_i is the observed covariate vector. $\delta_{c(i)}$ and $\lambda_{s(i)}$ denote country and industry fixed effects.

As a complementary specification, we estimate a cross-fitted augmented inverse-probability-weighted (AIPW) contrast between current agent users and no-use/no-plan respondents. Five stratified folds are used. A logistic model estimates the propensity for current use, while separate ridge outcome models estimate expected satisfaction for users and nonusers; numeric predictors are standardized after median imputation and categorical predictors are one-hot encoded. Propensities are clipped to [.05, .95]. Cross-fitting limits own-observation overfitting, and the augmentation

provides double robustness with respect to the two nuisance-model classes (Chernozhukov et al., 2018). The estimator is:

$$\theta = n^{-1}\Sigma_i[\hat{m}_{1i} - \hat{m}_{0i} + A_i(Y_i - \hat{m}_{1i})/\hat{e}_i - (1 - A_i)(Y_i - \hat{m}_{0i})/(1 - \hat{e}_i)] \quad (2)$$

Here, A_i identifies current agent use, Y_i is satisfaction, $\hat{e}_i = \hat{e}(X_i)$ is the estimated propensity, and $\hat{m}_{ai} = \hat{m}_a(X_i)$ is the treatment-specific outcome regression. The reported standard error clusters the cross-fitted influence score by country. Because conditional exchangeability and positivity with respect to unobserved variables cannot be established in this survey, θ is described as a doubly robust covariate-adjusted contrast—not a causal average treatment effect.

The within-user analysis standardizes the six focal experience measures and estimates nested OLS models with standard errors clustered by country: controls only (M1), resources plus controls (M2), resources and demands plus controls (M3), and an exploratory interaction model (M4). Agent-use intensity is coded 1 = monthly/infrequent, 2 = weekly, and 3 = daily. The preferred main-effects specification is:

$$JS_i = \alpha + R_i^T\beta + D_i^T\eta + \rho Intensity_i + \gamma^T X_i + \delta_{c(i)} + \lambda_{s(i)} + \varepsilon_i. \quad (3)$$

In Equation (3), R_i is the vector of standardized efficiency, capability, and coordination resources; D_i is the vector of standardized verification, implementation, and restriction demands; and $Intensity_i$ is current-use frequency. The coefficient scale is job-satisfaction points per one-standard-deviation difference in an experience measure.

3.4. Robustness, multiplicity, and missing data

Four checks address model dependence. First, ToolCountWork is added to the adoption and within-user models. Second, job satisfaction is recoded as high satisfaction (8–10 versus 0–7) and estimated with an unpenalized logistic model; countries with fewer than 30 complete users are pooled for the fixed-effect design, while inference remains clustered by original country. Third, each challenge item replaces the demand composites. Fourth, the preferred model is estimated separately for individual contributors and people managers. Benjamini–Hochberg adjustments control the false discovery rate across the nine focal item-level tests and across the 12 role-specific resource/demand tests (Benjamini & Hochberg, 1995). These analyses are labeled robustness or exploratory rather than new hypothesis tests.

Complete-case selection is addressed transparently. Current users with complete experience measures (N = 5,988) are compared with those excluded (N = 1,125) on observed characteristics. A logistic selection model then predicts completion using experience, manager role, pay, tool count, agent intensity, age, education, employment, organization size, remote-work mode, industry, and country. Inverse completion-probability weights are truncated at their 1st and 99th percentiles, and Equation (3) is re-estimated by weighted least squares with standard errors clustered by country. This check cannot recover selection on unobserved variables but shows whether the principal coefficients are robust to observed completion differences.

All calculations were performed in Python 3.12 using pandas, NumPy, SciPy, and scikit-learn. Figure 3 is generated directly from model output through a reproducible Matplotlib workflow; no generative image model was used. Because the focal variables come from the same respondent and survey wave, statistical adjustment cannot remove common-method bias or reverse causality (Podsakoff et al., 2024). We address these risks through transparent construct definitions, item-level checks, and conservative interpretation rather than treating any diagnostic as a definitive correction.

4. Results

4.1. Sample profile and bivariate patterns

Table 2 shows the six observed agent-status groups in the adoption sample. Respondents who neither used nor planned to use agents formed the largest group (36.4%), followed by planned adopters (17.3%), daily users (15.0%), copilot-only users (14.5%), weekly users (9.2%), and monthly or infrequent users (7.7%). Raw satisfaction varied over a narrow range: 7.213 among planned adopters to 7.456 among daily users. The monotonic rise appears only across the three current-agent frequency groups; the raw means do not support treating intention or copilot-only use as equivalent to agent use.

Table 2. Agent-use groups and unadjusted job satisfaction

Agent-use group	N	%	Mean	SD
No use / no plan	8,104	36.4	7.222	1.957
No use / plans adoption	3,847	17.3	7.213	1.913
Copilot only	3,225	14.5	7.259	1.818
monthly/infrequent	1,719	7.7	7.276	1.952
weekly	2,060	9.2	7.400	1.785
daily	3,334	15.0	7.456	1.898

Note. N = 22,289 respondents with valid agent-use status. Percentages are within this analytic sample and are not population-weighted. Job satisfaction ranges from 0 to 10.

Table 3 reports within-user descriptives and correlations. Efficiency is rated most favorably (M = 3.925), whereas coordination benefit is closer to disagreement/neutrality (M = 2.581). Verification concern is high (M = 4.173), but its bivariate correlation with satisfaction is small (r = −.043). The three resource domains correlate moderately to strongly with one another, especially efficiency and capability (r = .680), supporting their joint inclusion to estimate incremental associations. No construct correlation approaches unity.

Table 3. Descriptive statistics and correlations in the within-user sample

Variable	M	SD	1	2	3	4	5	6	7
1. Job satisfaction	7.414	1.870	1.00	—	—	—	—	—	—
2. Efficiency resource	3.925	0.871	0.08	1.00	—	—	—	—	—
3. Capability resource	3.406	0.978	0.07	0.68	1.00	—	—	—	—
4. Coordination resource	2.581	1.136	0.09	0.47	0.57	1.00	—	—	—
5. Verification burden	4.173	0.788	-0.04	-0.18	-0.21	-0.17	1.00	—	—
6. Implementation burden	3.318	0.859	-0.06	-0.12	-0.06	0.02	0.34	1.00	—

7. Organizational restriction	2.359	1.339	-0.05	-0.06	0.03	0.08	0.15	0.24	1.00

Note. N = 5,988. Pearson correlations are shown below the diagonal. Resource and demand measures range from 1 to 5; job satisfaction ranges from 0 to 10.

4.2. Agent-use frequency and job satisfaction

Table 4 presents the adoption models. Relative to the no-use/no-plan reference group, planned adoption (b = −0.009, p = .820), copilot-only use (b = 0.045, p = .218), and monthly or infrequent agent use (b = 0.031, p = .521) are not distinguishable from zero. Weekly use is associated with 0.175 additional satisfaction points (95% CI [0.085, 0.264]), and daily use with 0.239 points (95% CI [0.163, 0.314]). On a 0–10 outcome, these are small differences. H1 is supported for the prespecified weekly and daily categories, not for lighter use.

Table 4. Adjusted agent-use contrasts in job satisfaction

Contrast (reference: no use/no plan)	N	Estimate	SE	95% CI	p
Plans adoption	22,289	-0.009	0.040	[-0.088, 0.070]	0.820
Copilot only	22,289	0.045	0.037	[-0.027, 0.117]	0.218
Monthly/infrequent agent use	22,289	0.031	0.048	[-0.064, 0.127]	0.521
Weekly agent use	22,289	0.175***	0.045	[0.085, 0.264]	<.001
Daily agent use	22,289	0.239***	0.038	[0.163, 0.314]	<.001
Current user vs no-use/no-plan (AIPW)	15,217	0.145***	0.038	[0.071, 0.219]	<.001

*Note. OLS rows include the full observed adjustment set, country and industry fixed effects, and country-clustered standard errors (165 clusters); R^2 = .050, adjusted R^2 = .040. The AIPW row compares 7,113 current users with 8,104 no-use/no-plan respondents using five-fold cross-fitting; its influence-function SE is clustered across 156 observed countries. Propensity estimates ranged from .097 to .919 (1st–99th percentiles: .207–.836). * p < .05; ** p < .01; *** p < .001.*

The AIPW analysis yields a closely aligned covariate-adjusted difference of 0.145 satisfaction points (SE = 0.038, 95% CI [0.071, 0.219], p < .001). Propensity overlap is acceptable in the observed sample, with only the extreme tails approaching the clipping bounds. The estimate should nevertheless be read as robustness to alternative observed-variable adjustment, not as protection against omitted variables such as organizational climate, self-selection into experimentation, or prior satisfaction. A more satisfied developer may be more willing to adopt agents, and an AI-supportive employer may affect both adoption and satisfaction.

4.3. Resources, demands, and satisfaction among users

Table 5 reports the nested within-user models. With only resources and controls (M2), efficiency is positive but small (b = 0.085, p = .042), coordination is positive (b = 0.134, p = .002), and capability is null. Once demands are entered (M3), coordination remains the largest positive correlate (b = 0.148, 95% CI [0.061, 0.234], p < .001), whereas efficiency falls below conventional significance (b = 0.068, p = .103) and capability remains null (b = 0.024, p = .605).

H4 is supported; H2 is supported only in the resource-only model and not in the preferred joint model; H3 is not supported.

Table 5. Within-user OLS models of job satisfaction

Predictor	M1 Controls	M2 Resources	M3 + Demands	M4 + Interactions
Efficiency resource	—	0.085* (0.041)	0.068 (0.041)	0.065 (0.041)
Capability resource	—	0.019 (0.050)	0.024 (0.047)	0.025 (0.047)
Coordination resource	—	0.134** (0.042)	0.148*** (0.044)	0.146** (0.049)
Verification burden	—	—	0.006 (0.039)	0.004 (0.038)
Implementation burden	—	—	-0.087** (0.030)	-0.088** (0.030)
Organizational restriction	—	—	-0.053* (0.026)	-0.054* (0.026)
Agent-use intensity	0.102*** (0.029)	0.013 (0.027)	-0.004 (0.028)	-0.004 (0.027)
Efficiency × verification	—	—	—	0.020 (0.028)
Coordination × manager	—	—	—	0.007 (0.065)
Adjusted R^2	0.051	0.060	0.062	0.062
N	5,988	5,988	5,988	5,988

*Note. Cells report unstandardized job-satisfaction-point coefficients with cluster-robust SEs in parentheses. Resource and demand predictors are standardized; agent intensity ranges 1–3. All models include experience, manager role, log pay, age, education, employment, organization size, remote-work mode, industry and country fixed effects. N = 5,988; 141 country clusters. * p < .05; ** p < .01; *** p < .001.*

Implementation burden is negatively associated with satisfaction (b = −0.087, 95% CI [−0.146, −0.028], p = .004), supporting H6. Organizational restriction is also negative (b = −0.053, 95% CI [−0.105, −0.001], p = .044), supporting H7 at the conventional level. Verification burden is near zero (b = 0.006, p = .877), so H5 is not supported. Importantly, agent-use intensity is positive in the controls-only model (b = 0.102, p < .001) but becomes null once resources are entered and remains null in M3. Within users, frequency by itself contains little incremental information after the experience of use is represented.

The exploratory interactions in M4 are both null: efficiency × verification burden b = 0.020 (p = .469), and coordination × manager role b = 0.007 (p = .911). They neither improve adjusted R^2 nor qualify the main interpretation. Figure 3 displays the adoption and preferred within-user coefficients on a common outcome scale. The figure makes two patterns visible: positive adoption coefficients emerge only at weekly/daily use, and the social coordination coefficient is larger than the other resource coefficients while implementation and restriction coefficients point in the opposite direction.

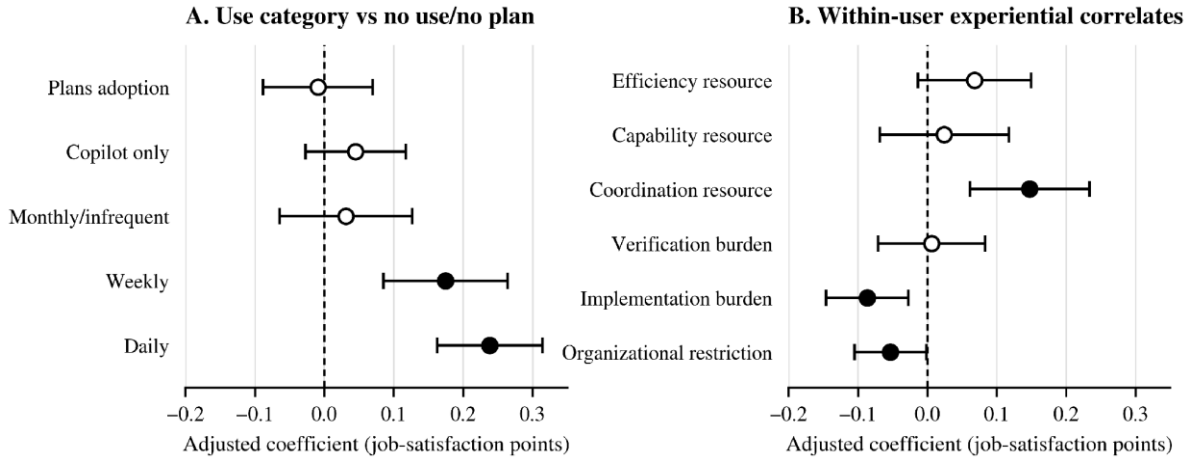

Figure 3. Adjusted associations with job satisfaction on a 0–10 scale.

Note: Panel A reports unstandardized contrasts relative to no use/no plan (N = 22,289); Panel B reports job-satisfaction-point differences for a one-standard-deviation increase in each experiential predictor (N = 5,988). Bars are 95% confidence intervals based on standard errors clustered by country; filled markers indicate intervals that exclude zero. Source: Authors' analysis.

4.4. Robustness and exploratory analyses

Table 6 consolidates the robustness evidence. Adding contemporaneous tool count leaves the adoption coefficients essentially unchanged (weekly b = 0.175; daily b = 0.238) and reproduces the preferred within−user pattern. Complete cases differ most from excluded users in experience (standardized difference = 0.216) and log pay (0.284); differences in satisfaction, manager role, tool count, and agent intensity are below 0.06 in absolute value. Completion weights are modest (mean = 1.187; 99th percentile = 1.552; effective N = 5,941), and the weighted coefficients remain similar: coordination b = 0.157, implementation burden b = −0.085, and restriction b = −0.052 (p = .050).

Table 6. Robustness and exploratory results

Analysis	Focal term	Estimate	95% CI	p or q	Inference
Tool-count sensitivity	Weekly use	b = 0.175	[0.085, 0.265]	<.001	Stable
Tool-count sensitivity	Daily use	b = 0.238	[0.161, 0.316]	<.001	Stable
Completion-weighted	Coordination	b = 0.157	[0.061, 0.253]	p = .001	Stable
Completion-weighted	Implementation	b = -0.085	[-0.143, -0.027]	p = .005	Stable
Completion-weighted	Restriction	b = -0.052	[-0.104, 0.000]	p = .050	Borderline
High satisfaction logit	Coordination	OR = 1.161	[1.076, 1.253]	p < .001	Same sign

High satisfaction logit	Implementation	OR = 0.954	[0.896, 1.016]	p = .143	Same sign
High satisfaction logit	Restriction	OR = 0.930	[0.874, 0.988]	p = .020	Same sign
Item-level demands	Coordination	b = 0.143	[0.058, 0.228]	q = .005	FDR stable
Item-level demands	Integration difficulty	b = -0.065	[-0.107, -0.022]	q = .009	FDR stable
Item-level demands	Platform cost	b = -0.114	[-0.178, -0.050]	q = .005	FDR stable
Item-level demands	Learning effort	b = 0.046	[0.005, 0.088]	q = .066	Not FDR significant
Exploratory interaction	Efficiency × verification	b = 0.020	[−0.035, 0.076]	p = .469	Null
Exploratory interaction	Coordination × manager	b = 0.007	[−0.121, 0.135]	p = .911	Null
Role split (FDR)	IC coordination	b = 0.155	[0.058, 0.253]	q = .024	FDR stable

Note. The binary model defines high satisfaction as 8–10 and reports odds ratios. q is the Benjamini–Hochberg false-discovery-rate adjusted p-value within the specified exploratory family. IC = individual contributor. Tool-count and completion-weighted analyses retain the preferred adjustment set.

The binary model confirms the positive coordination association (OR = 1.161, 95% CI [1.076, 1.253], p < .001) and negative restriction association (OR = 0.930, 95% CI [0.874, 0.988], p = .020); implementation burden has the same negative sign but a confidence interval crossing one. In the disaggregated demand model, coordination, integration difficulty, and platform cost survive FDR correction. Learning effort is unexpectedly positive before adjustment (b = 0.046, p = .029) but not after it (q = .066), consistent with either active learning investment or residual selection rather than a reliable benefit. Accuracy, security, and IT-rule items do not survive item-family correction.

Role-specific models show broadly similar coefficient directions but lower precision among the 907 people managers. After adjustment across 12 role-specific focal tests, only the coordination coefficient among individual contributors remains FDR significant (b = 0.155, q = .024). The role splits therefore do not justify a claim of differential effects. Taken together, the robustness analyses favor a parsimonious conclusion: weekly and daily use have small positive adjusted associations across respondents, while within users the most reproducible experiential signals are team coordination on the resource side and workflow integration/platform cost on the demand side.

5. Discussion

5.1. Interpretation of the findings

Weekly and daily agent use are associated with slightly higher job satisfaction than no use/no plan, whereas planned adoption, copilot-only use, and monthly or less frequent use are not statistically distinguishable from that reference category. The weekly and daily coefficients are small—approximately 0.18–0.24 points on the 0–10 scale—and the cross-fitted estimator yields a similarly small adjusted contrast. These results do not indicate a large shift in job satisfaction and should not be interpreted to mean that weekly or daily agent use causes higher satisfaction. They instead show why evidence of task-level productivity gains does not by itself establish how AI relates to the broader experience of work.

Among current users, use intensity is no longer statistically distinguishable from zero once the reported resource measures are included. Perceived efficiency is positive in the resource-only model but is imprecisely estimated after demands are added, while capability development is null throughout. Perceived coordination remains positively associated with satisfaction across the OLS, completion-weighted, item-level, and binary-outcome specifications. Because the variables were measured contemporaneously and the experience questions were conditional on current use, this pattern is an adjusted association rather than evidence that coordination mediates the adoption–satisfaction relationship.

The demand estimates also vary by dimension. Verification burden is not independently associated with satisfaction after adjustment, although it may relate to outcomes not observed here, such as code quality, security, or cognitive fatigue. Integration difficulty and platform cost provide the clearest negative item-level signals after FDR correction. Organizational restrictions are negative in the main OLS and binary specifications, but the evidence is less stable in the completion-weighted and item-level analyses. Overall, workflow disruption and platform cost are more closely associated with satisfaction than general risk concerns in this sample; the data do not establish why these associations differ.

5.2. Theoretical contributions

5.2.1. Delegation is not equivalent to assistant use

Research on agentic information systems emphasizes reciprocal delegation between users and technological artifacts (Baird & Maruping, 2021). The survey categories allow self-reported copilot-only use to be separated from multi-step agent use. Copilot-only use is not distinguishable from the no-use/no-plan reference category, whereas weekly and daily agent use are. Because the categories are self-selected and no direct contrast between copilot and agent users is reported, this pattern should not be read as evidence that agent use produces higher satisfaction than copilot use. Among current agent users, frequency has no detectable adjusted association with satisfaction once reported resources and demands are included.

5.2.2. Agent experiences as distinct job resources and demands

The JD–R contribution lies in specifying agent-related experiences at a level between technology presence and a single global benefit or technostress score. Recent JD–R research connects workload, organizational support, affordances, and prompt literacy to generative-AI use and innovative performance (Li et al., 2026). We extend that reasoning to a globally distributed developer sample, distinguish agents from conventional copilot use, examine overall job satisfaction, and separate verification, implementation, and governance demands. These distinctions matter because work design—not technological presence by itself—determines whether a system functions as a resource, a demand, or both (Bakker et al., 2023; Parker & Grote, 2022).

The differentiated demand estimates also qualify applications of the automation–augmentation paradox. Agent use may automate some steps while adding work in framing, checking, integration, and exception handling. The null verification estimate indicates only that verification burden is not independently associated with job satisfaction in these data; it does not imply that checking is unnecessary or irrelevant to security or fatigue (Perry et al., 2023). Integration burden shows clearer negative associations with satisfaction, while the evidence for organizational restrictions is less stable. Links between particular demands and other employee or quality outcomes require direct tests.

5.2.3. Team coordination as a social resource

The coordination result suggests that agent-enabled work should be studied beyond the user–tool dyad. Developer output must be reviewed, merged, maintained, and understood by colleagues, and reported coordination remains associated with satisfaction across the main robustness specifications. Human–AI combinations do not automatically create synergy (Vaccaro et al., 2024), while field evidence shows that AI can perform some knowledge-integration and social-affective functions of a teammate in bounded settings (Dell'Acqua et al., 2026a). Our single-

item measure cannot distinguish among shared conventions, knowledge transfer, inspectability, or other mechanisms. Team-level studies should test these mechanisms directly and treat agentic systems as elements of coordination structures rather than only as individual productivity tools.

5.3. Managerial implications

Managers should supplement adoption counts with measures of implementation quality. Activated licenses, prompt volume, and daily-active users capture exposure but not whether agent-supported work functions well. Pilot dashboards could also track integration and rework time, cost allocation, policy exceptions, team knowledge sharing, and satisfaction. Given the small adjusted adoption coefficients, scaling decisions should specify practically important thresholds for both output and employee outcomes.

Because perceived coordination is the most consistent positive correlate, pilots should examine how agent-generated work moves through teams. Shared use conventions, links between generated changes and validation evidence, approved-tool documentation, and ordinary code-review and incident-learning routines can make outputs more inspectable. Live execution support and contextual trust controls may help developers calibrate reliance (Ferdowsi et al., 2024; R. Wang et al., 2024). These practices should be evaluated rather than assumed to improve satisfaction.

Organizations can reduce several forms of implementation friction by budgeting platform costs centrally, maintaining environment-specific integrations and documentation, and allocating time for learning. The exploratory coefficient for learning effort did not survive FDR correction and therefore does not show that learning time improves satisfaction in this sample. Evidence from vocational education—a different population and outcome—indicates that intrinsic motivation and programme relevance shape learning outcomes (Zhang, 2026). Used only as an analogy, this finding suggests evaluating whether agent training is engaging and role-aligned rather than treating attendance as evidence of capability.

Governance should pair restrictions with approved tools, escalation channels, data-classification guidance, and clear rationales. These provisions may reduce ambiguity while preserving security and privacy requirements (Verma et al., 2024). The negative restriction coefficient is not equally stable across all specifications, so the present data support examining governance quality rather than asserting that restrictions themselves reduce satisfaction.

5.4. Limitations and a forward research agenda

The principal limitation is temporal and causal. All focal variables were measured once from the same respondent. Higher satisfaction may encourage experimentation, favorable workplace climate may influence both agent access and satisfaction, and retrospective evaluations may be colored by general affect. Fixed effects, AIPW, and sensitivity analyses balance observed variables; none supplies the untestable conditional exchangeability needed for causal interpretation. Longitudinal panel designs should measure satisfaction before adoption, during implementation, and after routinization. Staggered organizational rollouts could support difference-in-differences designs if assignment timing is plausibly exogenous, while randomized interface or training interventions could test specific workflow mechanisms.

External validity is bounded by the recruitment design. Stack Overflow recruitment favors engaged platform users, and the data provide no population weights. The public file/dashboard count discrepancy further shows why researchers should version and report the exact analytic file. Results should not be generalized to all developers, other occupations, or national workforces. Replication is needed in employer samples, less platform-engaged developer communities, and occupations where agent outputs are less directly inspectable. Cross-cultural measurement invariance and multilevel organization effects also warrant study; country fixed effects absorb mean differences but do not test whether resource–demand relationships vary institutionally.

Measurement is constrained by an omnibus public survey. Job satisfaction and coordination are single items, the demand domains are analytic composites rather than validated reflective scales, and detailed experience items

were shown only to current users. Objective usage logs, code-review data, security incidents, cycle time, and network measures of help and coordination would reduce common-source dependence. The distinction between copilot and agents rests on the survey definition and self-classification, while rapidly evolving systems blur that boundary. Future studies should record autonomy, tool invocation, memory, duration, reversibility, and human approval points rather than rely only on product labels. Generative-AI capabilities themselves continue to evolve quickly (Feuerriegel et al., 2024).

Missing data and dependence introduce further uncertainty. The within-user model omits 1,125 current users with incomplete experience items; completion weighting addresses observed differences but cannot correct nonignorable missingness. Compensation is also missing for some respondents and is adjusted with an indicator. Country clustering uses 141–165 groups, some of them small, whereas organization identifiers would permit modeling the more proximal dependence structure. Low explained variance and small coefficients are consistent with the many determinants of job satisfaction that the survey does not measure. The estimates are therefore more useful for prioritizing implementation questions than for predicting individual satisfaction.

Several designs could test the proposed mechanisms more directly. Team-level field experiments can compare individual-only access with shared-agent workflows and distinguish explanation, standardization, and workload redistribution as sources of coordination. Process tracing can separate productive verification from avoidable rework and connect each to quality, fatigue, and satisfaction. Governance experiments can compare blanket restrictions with approved sandboxes, review thresholds, and accountable exception processes. Longitudinal skill studies can examine whether agents accelerate learning initially but weaken unaided capability when delegation becomes habitual. Related firm-level research positions responsible innovation as a pathway linking collaborative and responsible entrepreneurship to venture performance (Zhang & Murad, 2026). Although that evidence concerns SMEs rather than employees using AI, it motivates testing whether responsible implementation similarly conditions the relationship between agent-enabled coordination and employee outcomes. Across these designs, the relevant unit is the evolving human–agent–team system rather than a static adopter.

6. Conclusion

The results do not indicate large differences in developer job satisfaction by AI-agent use status. In this public cross-sectional survey, weekly and daily use are associated with slightly higher adjusted satisfaction than no use/no plan, whereas lighter use and copilot-only use are not statistically distinguishable from that reference category. Among current users, perceived improvement in team coordination is the most consistent positive correlate. Integration difficulty and platform cost provide the clearest negative item-level signals; organizational restrictions are negative in the main specifications but less stable across robustness analyses. Efficiency is no longer independently associated with satisfaction after demands are included, capability development is null, and use frequency adds no explanatory value within users.

These findings favor a work-design interpretation of agent use. Task-level productivity gains should not be assumed to improve the overall job experience. Organizations can test whether shared workflows, inspectable outputs, supported integration, transparent cost allocation, and usable governance improve both performance and employee outcomes. Longitudinal and team-level designs are required before any causal claim can be made.

7. Declarations

Funding: No external funding was reported for this study.

Conflict of interest: The authors declare no conflict of interest.

Ethics statement: This study is a secondary analysis of an openly released, de-identified public-use dataset. No participant contact, intervention, or access to direct identifiers occurred.

Data availability: The survey data and schema are available from the Stack Overflow Survey repository under the Open Database License: https://github.com/StackExchange/Survey/tree/main/packages/archive/2025.

REFERENCES

1. Baird, A., & Maruping, L. M. (2021). The next generation of research on IS use: A theoretical framework of delegation to and from agentic IS artifacts. MIS Quarterly, 45(1), 315–341. https://doi.org/10.25300/MISQ/2021/15882
2. Bakker, A. B., Demerouti, E., & Sanz-Vergel, A. (2023). Job demands–resources theory: Ten years later. Annual Review of Organizational Psychology and Organizational Behavior, 10, 25–53. https://doi.org/10.1146/annurev-orgpsych-120920-053933
3. Barke, S., James, M. B., & Polikarpova, N. (2023). Grounded Copilot: How programmers interact with code-generating models. Proceedings of the ACM on Programming Languages, 7(OOPSLA1), Article 78, 85–111. https://doi.org/10.1145/3586030
4. Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. Journal of the Royal Statistical Society: Series B (Methodological), 57(1), 289–300. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x
5. Berente, N., Gu, B., Recker, J., & Santhanam, R. (2021). Managing artificial intelligence. MIS Quarterly, 45(3), 1433–1450. https://doi.org/10.25300/MISQ/2021/16274
6. Brynjolfsson, E., Li, D., & Raymond, L. R. (2025). Generative AI at work. The Quarterly Journal of Economics, 140(2), 889–942. https://doi.org/10.1093/qje/qjae044
7. Chernozhukov, V., Chetverikov, D., Demirer, M., Duflo, E., Hansen, C., Newey, W., & Robins, J. (2018). Double/debiased machine learning for treatment and structural parameters. The Econometrics Journal, 21(1), C1–C68. https://doi.org/10.1111/ectj.12097
8. Cui, K. Z., Demirer, M., Jaffe, S., Musolff, L., Peng, S., & Salz, T. (2026). The effects of generative AI on high-skilled work: Evidence from three field experiments with software developers. Management Science, Articles in Advance. https://doi.org/10.1287/mnsc.2025.00535
9. Dell'Acqua, F., Ayoubi, C., Lifshitz, H., Sadun, R., Mollick, E., Mollick, L., Han, Y., Goldman, J., Nair, H., Taub, S., & Lakhani, K. R. (2026a). The cybernetic teammate: A field experiment on generative AI and teamwork. Organization Science, 37(4), 1217–1242. https://doi.org/10.1287/orsc.2025.20702
10. Dell'Acqua, F., McFowland III, E., Mollick, E., Lifshitz, H., Kellogg, K. C., Rajendran, S., Krayer, L., Candelon, F., & Lakhani, K. R. (2026b). Navigating the jagged technological frontier: Field experimental evidence of the effects of artificial intelligence on knowledge worker productivity and quality. Organization Science, 37(2), 403–423. https://doi.org/10.1287/orsc.2025.21838
11. Ferdowsi, K., Huang, R., James, M. B., Polikarpova, N., & Lerner, S. (2024). Validating AI-generated code with live programming. Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems, Article 143, 1–8. https://doi.org/10.1145/3613904.3642495
12. Feuerriegel, S., Hartmann, J., Janiesch, C., & Zschech, P. (2024). Generative AI. Business & Information Systems Engineering, 66, 111–126. https://doi.org/10.1007/s12599-023-00834-7
13. Fügener, A., Grahl, J., Gupta, A., & Ketter, W. (2022). Cognitive challenges in human–artificial intelligence collaboration: Investigating the path toward productive delegation. Information Systems Research, 33(2), 678–696. https://doi.org/10.1287/isre.2021.1079
14. Gustavsson, T., Ahmad, M. O., & Saeeda, H. (2025). Job satisfaction at risk: Measuring the role of process debt in agile software development. Journal of Systems and Software, 222, 112350. https://doi.org/10.1016/j.jss.2025.112350
15. Hai, S., Long, T., Honora, A., Japutra, A., & Guo, T. (2025). The dark side of employee-generative AI collaboration in the workplace: An investigation on work alienation and employee expediency. International Journal of Information Management, 83, 102905. https://doi.org/10.1016/j.ijinfomgt.2025.102905
16. Högemann, M., Hein, L., Britsche, J.-O., & Thomas, O. (2025). Technostress and generative AI in the workplace: A qualitative analysis of young professionals. Frontiers in Artificial Intelligence, 8, 1728881. https://doi.org/10.3389/frai.2025.1728881
17. Jussupow, E., Benbasat, I., & Heinzl, A. (2024). An integrative perspective on algorithm aversion and appreciation in decision-making. MIS Quarterly, 48(4), 1575–1590. https://doi.org/10.25300/MISQ/2024/18512
18. Kumar, A., Krishnamoorthy, B., & Bhattacharyya, S. S. (2024). Machine learning and artificial intelligence-induced technostress in organizations: A study on automation–augmentation paradox with socio-technical systems as coping mechanisms. International Journal of Organizational Analysis, 32(4), 681–701. https://doi.org/10.1108/IJOA-01-2023-3581
19. Li, Z., Sun, H., Yang, T., & Zeng, K. (2026). Generative AI in the workplace: How job demands and resources influence employee innovative performance—A JD–R theory perspective. European Journal of Innovation Management, 29(6), 1822–1845. https://doi.org/10.1108/EJIM-11-2025-1496
20. Liang, J. T., Yang, C., & Myers, B. A. (2024). A large-scale survey on the usability of AI programming assistants: Successes and challenges. Proceedings of the IEEE/ACM 46th International Conference on Software Engineering, 616–628. https://doi.org/10.1145/3597503.3608128
21. Moradi Dakhel, A., Majdinasab, V., Nikanjam, A., Khomh, F., Desmarais, M. C., & Jiang, Z. M. (2023). GitHub Copilot AI pair programmer: Asset or liability? Journal of Systems and Software, 203, 111734. https://doi.org/10.1016/j.jss.2023.111734

22. Mozannar, H., Bansal, G., Fourney, A., & Horvitz, E. (2024). Reading between the lines: Modeling user behavior and costs in AI-assisted programming. Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems, Article 142, 1–16. https://doi.org/10.1145/3613904.3641936
23. Noponen, N., Feshchenko, P., Auvinen, T., Luoma-aho, V., & Abrahamsson, P. (2024). Taylorism on steroids or enabling autonomy? A systematic review of algorithmic management. Management Review Quarterly, 74, 1695–1721. https://doi.org/10.1007/s11301-023-00345-5
24. Noy, S., & Zhang, W. (2023). Experimental evidence on the productivity effects of generative artificial intelligence. Science, 381(6654), 187–192. https://doi.org/10.1126/science.adh2586
25. Parent-Rocheleau, X., & Parker, S. K. (2022). Algorithms as work designers: How algorithmic management influences the design of jobs. Human Resource Management Review, 32(3), 100838. https://doi.org/10.1016/j.hrmr.2021.100838
26. Parker, S. K., & Grote, G. (2022). Automation, algorithms, and beyond: Why work design matters more than ever in a digital world. Applied Psychology, 71(4), 1171–1204. https://doi.org/10.1111/apps.12241
27. Perry, N., Srivastava, M., Kumar, D., & Boneh, D. (2023). Do users write more insecure code with AI assistants? Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security, 2785–2799. https://doi.org/10.1145/3576915.3623157
28. Podsakoff, P. M., Podsakoff, N. P., Williams, L. J., Huang, C., & Yang, J. (2024). Common method bias: It's bad, it's complex, it's widespread, and it's not easy to fix. Annual Review of Organizational Psychology and Organizational Behavior, 11, 17–61. https://doi.org/10.1146/annurev-orgpsych-110721-040030
29. Raisch, S., & Krakowski, S. (2021). Artificial intelligence and management: The automation–augmentation paradox. Academy of Management Review, 46(1), 192–210. https://doi.org/10.5465/amr.2018.0072
30. Russo, D., Hanel, P. H. P., Altnickel, S., & van Berkel, N. (2023). Satisfaction and performance of software developers during enforced work from home in the COVID-19 pandemic. Empirical Software Engineering, 28(2), Article 53. https://doi.org/10.1007/s10664-023-10293-z
31. Savolainen, I., Ylinen, L., Grönroos, R., & Oksanen, A. (2026). AI transformation in working life: A systematic review of usage and attitudes towards AI among workers. Digital Business, 6(1), 100162. https://doi.org/10.1016/j.digbus.2025.100162
32. Stack Overflow. (2025). Stack Overflow Annual Developer Survey 2025: Public data and methodology. https://survey.stackoverflow.co/2025/; https://github.com/StackExchange/Survey/tree/main/packages/archive/2025 (accessed 16 August 2026).
33. Vaccaro, M., Almaatouq, A., & Malone, T. W. (2024). When combinations of humans and AI are useful: A systematic review and meta-analysis. Nature Human Behaviour, 8, 2293–2303. https://doi.org/10.1038/s41562-024-02024-1
34. Verma, S., Singh, V., Tudoran, A. A., & Bhattacharyya, S. S. (2024). Elevating employees' psychological responses and task performance through responsible artificial intelligence. Information Technology & People, 37(7), 2551–2567. https://doi.org/10.1108/ITP-05-2023-0431
35. Wang, R., Cheng, R., Ford, D., & Zimmermann, T. (2024). Investigating and designing for trust in AI-powered code generation tools. Proceedings of the 2024 ACM Conference on Fairness, Accountability, and Transparency, 1475–1493. https://doi.org/10.1145/3630106.3658984
36. Wang, W., Ning, H., Zhang, G., Liu, L., & Wang, Y. (2024). Rocks coding, not development: A human-centric, experimental evaluation of LLM-supported software engineering tasks. Proceedings of the ACM on Software Engineering, 1(FSE), Article 32, 699–721. https://doi.org/10.1145/3643758
37. Weber, T., Brandmaier, M., Schmidt, A., & Mayer, S. (2024). Significant productivity gains through programming with large language models. Proceedings of the ACM on Human-Computer Interaction, 8(EICS), Article 256, 1–29. https://doi.org/10.1145/3661145
38. Yang, J., Jimenez, C. E., Wettig, A., Lieret, K., Yao, S., Narasimhan, K., & Press, O. (2024). SWE-agent: Agent-computer interfaces enable automated software engineering. Advances in Neural Information Processing Systems, 37. https://doi.org/10.52202/079017-1601
39. Zhang, X. (2026). Sustainability-oriented teaching in vocational higher education: How intrinsic motivation, programme relevance and learning commitment shape students' social value outcomes. Education + Training, 68(6), 956–972. https://doi.org/10.1108/ET-02-2026-0183
40. Zhang, X., & Murad, M. (2026). When collaboration meets responsibility: Responsible innovation as a pathway linking collaborative and responsible entrepreneurship to venture performance. Corporate Social Responsibility and Environmental Management, advance online publication, 1–18. https://doi.org/10.1002/csr.70764
41. Ziegler, A., Kalliamvakou, E., Li, X. A., Rice, A., Rifkin, D., Simister, S., Sittampalam, G., & Aftandilian, E. (2022). Productivity assessment of neural code completion. Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming, 21–29. https://doi.org/10.1145/3520312.3534864