ISSN: 2623-1662 (online), 2460-092X (printed)
DOI: https://doi.org/10.19109/jusifo.v11i1.25082
Vol. 11, No. 1 (2025), p.11-20

JUSIFO
JURNAL SISTEM INFORMASI

Review Article [OPEN ACCESS]

The Impact of Artificial Intelligence on Programmer Jobs: Threat or Opportunity?

Agustia Hananto, Nadya Susanti Hasibuan*, Stevi Olivia, Tukino Tukino, Elfina Novalia

ABSTRACT

The integration of artificial intelligence (AI) into software development has significantly reshaped the programming profession. This study investigates the dual impact of AI—particularly Generative AI tools such as ChatGPT and Codex—on programming jobs, examining both the potential threats and emerging opportunities. Using a Systematic Literature Review (SLR) approach, 18 peer-reviewed articles published within the last five years were analyzed. The findings reveal that while AI automates routine tasks such as code generation, debugging, and testing—raising concerns about the reduction of entry-level roles—it simultaneously enables programmers to assume more strategic functions in AI integration, system design, and ethical governance. Additionally, AI fosters inclusive programming environments by empowering non-experts through accessible development tools. The study emphasizes the need for continuous skill enhancement, human–AI collaboration, and policy support to ensure sustainable adaptation. This research contributes to a deeper understanding of AI's transformative influence on programming and proposes strategies for thriving in an AI-augmented digital workforce.

Keyword: Generative AI, programmer adaptation, software automation

Received: October 30, 2024; Revised: April 18, 2025; Accepted: June 16, 2025

Corresponding Author: Nadya Susanti Hasibuan, Department of Information System, Universitas Buana Perjuangan Karawang, Indonesia, nadyasusantihasibuan@gmail.com
Authors: Agustia Hananto, Department of Information System, Universitas Buana Perjuangan Karawang, Indonesia, agustia.hananto@ubpkarawang.ac.id; Stevi Olivia, Department of Information System, Universitas Buana Perjuangan Karawang, Indonesia, steviolivia10@gmail.com; Tukino Tukino, Department of Information System, Universitas Buana Perjuangan Karawang, Indonesia, tukino@ubpkarawang.ac.id; Elfina Novalia, Department of Information System, Universitas Buana Perjuangan Karawang, Indonesia, elfina.novalia@ubpkarawang.ac.id

1. INTRODUCTION

Artificial intelligence (AI) has emerged as a transformative force across various sectors, particularly in information technology (IT), by enhancing automation, decision-making, and operational efficiency (Alief & Nurmiati, 2022; Ray, 2023; Sihananto et al., 2024). Leveraging advanced algorithms, AI improves performance through applications such as predictive maintenance in manufacturing and fraud detection in financial services. More recently, the development of generative AI tools—including GitHub Copilot, OpenAI Codex, and TabNine—has further revolutionized software development by providing real-time code suggestions, intelligent error detection, and natural language-based code generation (Aarti, 2024). In parallel, low-code and no-code platforms such as Microsoft Power Apps and Google AppSheet are enabling non-programmers to develop software through intuitive interfaces, dramatically lowering entry barriers (Roach, 2022).

Despite these promising advancements, the rise of AI in programming brings forth substantial concerns. Automation of routine coding tasks raises the possibility of job displacement, particularly among early-career software developers (Ferdiana, 2024). Furthermore, AI's increasing capabilities, as

emphasized by NVIDIA CEO Jensen Huang, reduce the need for deep technical proficiency, allowing broader segments of society to engage in software creation without mastering programming languages (Choquette et al., 2021). This shift challenges the traditional role of programmers and creates uncertainty regarding the skills and professional relevance of developers in an AI-driven environment. As AI systems continue to evolve, it becomes critical to assess how they reshape not only technical workflows but also the economic and strategic positioning of programmers in the software industry.

While existing literature acknowledges AI's capacity to enhance productivity and simplify development tasks, it tends to focus either on the technological potentials or on generalized impacts on employment. Few studies offer a comprehensive analysis of the dual nature of AI as both a disruptor and an enabler within programming careers. Moreover, there is limited empirical exploration into how programmers can proactively adapt, upgrade their competencies, and reposition themselves in response to AI integration (Afandi & Kurnia, 2023; Kumar et al., 2024; Mutaqin et al., 2024). This research gap hampers our understanding of how the evolving human–AI interaction can be leveraged to ensure mutual augmentation rather than replacement.

To address this issue, the present study examines the impact of AI on programming careers by investigating both its potential threats and emerging opportunities. This study explores how AI alters work processes, reshapes required competencies, and challenges traditional notions of expertise in software development. By identifying the strategic approaches developers can adopt in the face of accelerating automation, this research contributes to a nuanced understanding of how AI can act not only as a source of disruption but also as a catalyst for innovation, collaboration, and human-centered growth in the programming profession.

2. LITERATURE REVIEW AND METHODS

This study adopts a descriptive qualitative approach, with a primary emphasis on literature review as the method of data collection. The use of qualitative description allows for an in-depth understanding of existing concepts, trends, and theoretical developments related to the research topic. By synthesizing insights from prior scholarly works, this method establishes a conceptual foundation to support the analysis and interpretation of findings. To ensure a rigorous and systematic examination of existing research, the study employs the Systematic Literature Review (SLR) methodology (Jones, 2004; Snyder, 2019). The SLR process comprises clearly defined stages: formulating research questions, identifying relevant literature using predetermined inclusion and exclusion criteria, assessing the quality of selected studies, and systematically synthesizing findings. This methodology is particularly suitable for achieving transparency, reproducibility, and comprehensiveness in the review process.

Data sources were carefully curated to maintain academic integrity and relevance. Scientific articles, reports, and other credible publications concerning programmers, information technology, and artificial intelligence form the core of the data set. These sources were obtained from Semantic Scholar, Google Scholar, and other peer-reviewed databases recognized for their reliability and scholarly value. Following data collection, the selected literature was subjected to analytical procedures aimed at extracting key findings that align with the research objectives. The analysis focused on identifying patterns, thematic categories, and critical insights that inform the ongoing discourse on the impact of artificial intelligence on programming professions.

2.1 Research Question

The rapid advancement of artificial intelligence (AI) has generated significant discourse regarding its implications for the labor market, particularly within the domain of software programming. As AI-powered tools increasingly demonstrate the capacity to automate various coding tasks, concerns about the potential displacement of programmers have become prominent. Nonetheless, these developments also introduce new possibilities for enhancing productivity, streamlining workflows, and expanding the technical skillsets required in modern software development.

In light of these dual perspectives—automation-related threats and innovation-driven opportunities—this study seeks to investigate how AI, especially Generative AI models, is reshaping the programming profession. The research aims to analyze not only the technical transformations but also the broader pedagogical and professional consequences in the digital era. Accordingly, this study formulates the following research question:

RQ: How does the adoption of Artificial Intelligence, particularly Generative AI such as ChatGPT and Codex, influence programming practices, technology education, and digital-era work dynamics?

2.2 Search Process

The search process is a critical step in ensuring the relevance and quality of the literature included in this study. At this stage, the objective is to identify and retrieve scholarly journals and research articles from reputable online databases that directly relate to the research topic. All selected sources must meet clearly defined inclusion criteria, which were established to ensure alignment with the study's objectives and to maintain methodological consistency. Following the identification of potential sources, a systematic retrieval and evaluation process is undertaken. This involves analyzing the content of each article to determine its relevance to the research questions formulated in the previous phase. Only studies that offer substantial insights and meet the inclusion standards are incorporated into the final synthesis. This approach ensures that the literature used contributes meaningfully to answering the research questions and supports the overall reliability and depth of the findings.

2.3 Inclusion and Exclusion Criteria

To ensure the relevance and rigor of the literature reviewed, this study applies predefined inclusion and exclusion criteria during the selection process. Only papers that explicitly examine the impact of artificial intelligence (AI) on programmers, including its practical applications across various domains, are considered. Furthermore, the scope of the review is restricted to studies published within the last five years to maintain alignment with the most recent developments and trends in AI research. Studies that fall outside this publication window or fail to address the research focus are excluded from the final analysis.

2.4 Quality Assessment

Following the selection process, a quality assessment is conducted to evaluate whether the included studies meet the standards required for academic reliability and thematic relevance. This assessment ensures that only high-quality, peer-reviewed sources contribute to the synthesis of findings. Three key criteria guide the evaluation process:

QA1: Does the selected research directly address Artificial Intelligence (AI) in programmers or a specific field?

QA2: Are the research methods in the paper well-defined?

QA3: Was the research published within the last five years in a reputable, peer-reviewed journal or conference proceedings, ensuring its relevance to current AI advances?

Each study is assessed based on these questions, and responses are coded as Y (Yes) or N (No). Only studies that satisfy all three criteria are retained for further analysis, thereby upholding the methodological integrity of the review.

3. RESULTS AND DISCUSSION

3.1 Process and Inclusion–Exclusion Criteria

To streamline the identification of relevant studies that address the formulated research questions, the selected papers were initially categorized based on their respective journals or publishers. This

classification provides a structured overview of the source distribution and ensures the diversity and credibility of the literature included. A total of 18 papers focusing on the intersection of Artificial Intelligence (AI) and programming were compiled from 11 distinct journals or publishers, as summarized in Table 1.

Table 1. Distribution of selected papers by journal or publisher

Journal/Publisher	Quantity	Reference
Association for Computing Machinery (ACM)	8	(Chen, 2024; Feldman & Anderson, 2024; Finnie-Ansley et al., 2022; Kazemitabaar et al., 2023; Mozannar et al., 2024; Ross et al., 2023; Weisz et al., 2021, 2022)
Empirical Software Engineering	1	(Mailach et al., 2025)
Bulletin of Electrical Engineering and Informatics	1	(Hasan et al., 2024)
Proceedings of the ACM on Programming Languages	1	(Barke et al., 2023)
SSRN Electronic Journal	1	(Baidoo-Anu & Owusu Ansah, 2023)
Management Studies and Business Journal (Productivity)	1	(Sucipto, 2024)
Computing and Artificial Intelligence	1	(Akhtar, 2025)
Information Fusion	1	(Barredo Arrieta et al., 2020)
Science of Computer Programming	1	(Kuhail et al., 2024)
Applied Sciences	1	(Rahman & Watanobe, 2023)
Education Sciences	1	(Lo, 2023)

Following the initial classification, each of the 18 collected papers was evaluated using the inclusion and exclusion criteria established in Section 2.3. This screening process was conducted to ensure that only studies directly aligned with the research objectives—namely, those addressing the impact of AI on programmers and programming practices—were retained for further analysis. The application of these criteria enhances the methodological rigor and relevance of the study by eliminating sources that fall outside the defined scope.

3.2 Assessment Outcomes of Selected Literature

This stage verifies that every study selected through the inclusion–exclusion filter meets the methodological standards required for a rigorous synthesis. Each paper is appraised against the three quality-assessment questions (QA1–QA3) outlined in Section 2.4. A score of “Y” (Yes) indicates full compliance, whereas “N” (No) signals non-conformity; only papers earning “Y” across all criteria are retained.

Table 2 lists the 18 papers evaluated, recording author names, article titles, publication years, and the outcome for each quality item. All studies satisfy QA1 (direct relevance to AI in programming or a specific application domain), QA2 (clearly defined and sound research methods), and QA3 (publication within the past five years in reputable, peer-reviewed outlets). Consequently, every paper received a “Y” for all three criteria and is classified as “Accepted” for the final analysis. This result confirms that the evidence base is both current and methodologically robust, providing a solid foundation for the subsequent synthesis of findings.

Table 2. Quality-assessment outcomes for the 18 retained studies

No.	Author(s)	Title	QA1	QA2	QA3	Result
1	Mailach et al. (2025)	""Ok pal, we have to code that now": interaction patterns of programming beginners with a conversational chatbot"	Y	Y	Y	Accepted
2	Hasan et al. (2024)	"A novel approach to analyzing the impact of ai, chatgpt, and chatbot on education using machine learning algorithms"	Y	Y	Y	Accepted
3	Sucipto (2024)	"The impact of artificial intelligence (ai) on human resource management practices"	Y	Y	Y	Accepted
4	Akhtar (2025)	"Generative artificial intelligence (gai): from large language models (llms) to multimodal applications towards fine tuning of models, implications, investigations"	Y	Y	Y	Accepted
5	Kuhail et al. (2024)	""Will i be replaced?" assessing chatgpt's effect on software development and programmer perceptions of ai tools"	Y	Y	Y	Accepted
6	Chen (2024)	"The impact of ai-pair programmers on code quality and developer satisfaction: evidence from timi studio"	Y	Y	Y	Accepted
7	Feldman & Anderson (2024)	"Non-expert programmers in the generative ai future"	Y	Y	Y	Accepted
8	Mozannar et al. (2024)	"Reading between the lines: modeling user behavior and costs in ai-assisted programming"	Y	Y	Y	Accepted
9	Ross et al. (2023)	"The programmer's assistant: conversational interaction with a large language model for software development"	Y	Y	Y	Accepted
10	Barke et al. (2023)	"Grounded copilot: how programmers interact with code-generating models"	Y	Y	Y	Accepted
11	Baidoo-Anu & Owusu Ansah (2023)	"Education in the era of generative artificial intelligence (ai): understanding the potential benefits of chatgpt in promoting teaching and learning"	Y	Y	Y	Accepted
12	Kazemitabaar et al. (2023)	"Studying the effect of ai code generators on supporting novice learners in introductory programming"	Y	Y	Y	Accepted
13	Rahman & Watanobe (2023)	"Chatgpt for education and research: opportunities, threats, and strategies"	Y	Y	Y	Accepted
14	Lo (2023)	"What is the impact of chatgpt on education? a rapid review of the literature"	Y	Y	Y	Accepted
15	Weisz et al. (2022)	"Better together? an evaluation of ai-supported code translation"	Y	Y	Y	Accepted
16	Finnie-Ansley et al. (2022)	"The robots are coming: exploring the implications of openai codex on introductory programming"	Y	Y	Y	Accepted
17	Weisz et al. (2021)	"Perfection not required? human-ai partnerships in code translation"	Y	Y	Y	Accepted
18	Barredo Arrieta et al. (2020)	"Explainable artificial intelligence (xai): concepts, taxonomies, opportunities and challenges toward responsible ai"	Y	Y	Y	Accepted

3.3 Results Related to the Research Question

This section presents the synthesized findings of the study and directly addresses the formulated Research Question (RQ). The analysis is structured to ensure alignment between the empirical evidence drawn from the reviewed literature and the stated research objectives. By systematically categorizing the thematic patterns found in the selected studies, this section explains how Artificial Intelligence (AI), particularly Generative AI technologies such as ChatGPT and Codex, is influencing programming practices, technology education, and the broader dynamics of work in the digital era.

To address RQ, the selected papers were grouped according to the specific thematic reasons describing the influence and implications of AI on programming and digital transformation. These groupings reflect the most commonly identified effects of AI adoption, as outlined in Table 3. Each theme is associated with a cluster of articles that highlight consistent patterns across the literature.

Table 3. Thematic grouping of AI's influence on programming and related fields

No.	Thematic Reason	Paper Numbers	Total
1	AI enhances productivity and code quality in programming	[9], [10], [15], [17]	4
2	AI supports interactive and personalized learning in programming education	[2], [11], [12], [16]	4
3	AI transforms work dynamics and redefines the role of programmers in the digital ecosystem	[5], [6]	2
4	AI increases accessibility to technology for diverse user groups	[1], [7], [8]	3
5	AI raises ethical concerns, emphasizing transparency and accountability in intelligent systems	[3], [18]	2
6	AI drives transformation in education, research, and the role of educators	[4], [13], [14]	3

3.4 Discussion

Artificial intelligence (AI) exerts a dual, contrasting influence on organisational life. On the risk side, AI adoption introduces fresh cyber-security vulnerabilities, exacerbates data-privacy concerns, and automates tasks that once provided secure employment, thereby raising the spectre of workforce displacement. Conversely, the same technologies enrich knowledge sharing, expand employee autonomy, stimulate creativity, and measurably boost job performance (Malik et al., 2022). Managing AI's integration therefore demands a balanced strategy that heightens its advantages while vigilantly containing its downsides through robust governance and ethical oversight.

AI-related threats to programming roles centre on automation's encroachment into core development activities. Empirical studies reveal that tools such as GitHub Copilot already solve roughly half of programming problems on the first attempt and 60% after simple natural-language restatements, effectively shifting how coding skills are acquired and exercised (Denny et al., 2023). As AI-driven test suites autonomously locate defects, routine quality-assurance positions risk obsolescence. Competitive pressure intensifies as firms with advanced AI capabilities deliver software faster and at lower cost, compelling developers to continually upgrade their expertise. Entry-level positions are especially vulnerable: tasks once reserved for junior staff are increasingly off-loaded to intelligent systems, forcing newcomers to pivot towards higher-level responsibilities such as architectural design and systems oversight.

To remain professionally relevant, programmers must adopt proactive survival strategies. First, deepening technical proficiency in machine learning, natural-language processing, and AI system design is essential; language-learning platforms can accelerate this upskilling, provided users cultivate the contextual judgment those systems cannot supply (Rebolledo et al., 2023). Second, developers should channel AI's capabilities into novel product and service concepts, thereby positioning themselves at the forefront of

innovation. Third, critical thinking and holistic problem-solving—skills that translate business requirements into elegant technical solutions—will continue to differentiate human talent from automated code generation. Fourth, collaborating with AI rather than competing against it enhances efficiency; controlled experiments show AI pair-programming reduces task-completion time by over 55% (Peng et al., 2023). Finally, refined soft skills—communication, project management, and cross-functional teamwork—remain indispensable because AI cannot fully replicate nuanced human interaction.

At the same time, AI opens significant new opportunities for adaptable developers. Integrating intelligent modules into mainstream applications generates fresh business value, as captured by the Implications–Applications–Methods (IAM) framework (Sestino & De Mauro, 2022). Automating repetitive chores—boilerplate generation, debugging, and test construction—via systems such as IntelliCode, CodeWhisperer, and Tabnine allows engineers to concentrate on complex, high-impact workstreams (Ferdiana, 2024). Market demand for AI-literate programmers continues to outpace supply, translating into attractive career prospects. Finally, productivity gains from trustworthy, context-aware AI promise faster development cycles and accelerated innovation, underscoring the profession's evolution rather than its extinction.

Together, these perspectives confirm that AI's transformative power is neither uniformly threatening nor uniformly beneficial; instead, its net effect depends on the extent to which organisations and individuals strategically mitigate risks while exploiting new possibilities.

4. CONCLUSION

This study set out to clarify artificial intelligence's dual impact on programming careers—simultaneously disruptive and enabling—and the results corroborate that premise. Generative-AI tools now automate routine activities such as boilerplate code generation, unit-test creation, and initial debugging, thereby shrinking demand for many entry-level tasks. Yet the same technologies enlarge the professional horizon by creating roles in algorithm design, machine-learning system management, and intelligent-system integration. Programmers who continuously upgrade complementary skills—data-driven architecture, ethical-security validation, and advanced algorithmic thinking—can pivot toward these higher-value functions and retain strategic relevance in an AI-saturated landscape.

The investigation is limited by its reliance on qualitative synthesis; interpretive findings drawn from recent literature may not capture sector- or region-specific variations, nor keep pace with rapid AI advances. Future research should therefore adopt empirical, longitudinal designs to trace how programmers' competencies evolve, identify the most critical skill sets, and evaluate educational models that prepare new talent for AI-rich work environments. Such studies would refine our understanding of AI's labour-market effects and guide targeted policy or curriculum interventions.

Overall, Generative AI—exemplified by ChatGPT and Codex—should be viewed less as a replacement for human developers and more as a catalyst that redefines their remit. By off-loading repetitive, procedural tasks, AI frees programmers to concentrate on architectural strategy, ethical oversight, and creative problem-solving. It simultaneously broadens participation by enabling non-experts to prototype software under professional guidance, positioning experienced developers as facilitators, integrators, and innovators within human-machine teams. Far from rendering human skills obsolete, the AI era elevates programmers to stewards of a more intelligent, inclusive, and sustainable digital transformation.

To fully leverage this transformation, collaboration between academia, industry, and policymakers is essential. Education systems must adapt by embedding AI literacy and interdisciplinary problem-solving into curricula, while governments and organizations must support lifelong learning and reskilling initiatives. Only through this coordinated response can societies harness AI not as a threat to employment, but as a lever for inclusive growth, innovation, and digital equity in the evolving world of work.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

REFERENCES

Aarti, A. (2024). Generative ai in software development: an overview and evaluation of modern coding tools. *IJFMR - International Journal for Multidisciplinary Research*, *6*(3). https://doi.org/10.36948/IJFMR.2024.V06I03.23271

Afandi, A. R., & Kurnia, H. (2023). Revolusi teknologi: masa depan kecerdasan buatan (ai) dan dampaknya terhadap masyarakat. *Academy of Social Science and Global Citizenship Journal*, *3*(1), 9–13. https://doi.org/10.47200/AOSSAGCJ.V3I1.1837

Akhtar, Z. Bin. (2025). Generative artificial intelligence (gai): from large language models (llms) to multimodal applications towards fine tuning of models, implications, investigations. *Computing and Artificial Intelligence*, *2025*(1), 1498–1498. https://doi.org/10.59400/CAI.V3I1.1498

Alief, R., & Nurmiati, E. (2022). Penerapan kecerdasan buatan dan teknologi informasi pada efisiensi manajemen pengetahuan. *Jurnal Masyarakat Informatika*, *13*(1), 59–69. https://doi.org/10.14710/JMASIF.13.1.43760

Baidoo-Anu, D., & Owusu Ansah, L. (2023). Education in the era of generative artificial intelligence (ai): understanding the potential benefits of chatgpt in promoting teaching and learning. *SSRN Electronic Journal*. https://doi.org/10.2139/SSRN.4337484

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). https://doi.org/10.1145/3586030

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2020). Explainable artificial intelligence (xai): concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, *58*, 82–115. https://doi.org/10.1016/J.INFFUS.2019.12.012

Chen, T. (2024). The impact of ai-pair programmers on code quality and developer satisfaction: evidence from timi studio. *ACM International Conference Proceeding Series*, 201–205. https://doi.org/10.1145/3665348.3665383

Choquette, J., Gandhi, W., Giroux, O., Stam, N., & Krashinsky, R. (2021). Nvidia a100 tensor core gpu: performance and innovation. *IEEE Micro*, *41*(2), 29–35. https://doi.org/10.1109/MM.2021.3061394

Denny, P., Kumar, V., & Giacaman, N. (2023). Conversing with copilot: exploring prompt engineering for solving cs1 problems using natural language. *SIGCSE 2023 - Proceedings of the 54th ACM Technical Symposium on Computer Science Education*, *1*, 1136–1142. https://doi.org/10.1145/3545945.3569823

Feldman, M. Q., & Anderson, C. J. (2024). Non-expert programmers in the generative ai future. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3663384.3663393

Ferdiana, R. (2024). *The impact of artificial intelligence on programmer productivity*. https://www.researchgate.net/publication/378962192_The_Impact_of_Artificial_Intelligence_on_Programmer_Productivity

Finnie-Ansley, J., Denny, P., Becker, B. A., Luxton-Reilly, A., & Prather, J. (2022). The robots are coming: exploring the implications of openai codex on introductory programming. *ACM International Conference Proceeding Series*, 10–19. https://doi.org/10.1145/3511861.3511863

Hasan, N., Polin, J. A., Ahmmed, M. R., Sakib, M. M., Jahin, M. F., & Rahman, M. M. (2024). A novel approach to analyzing the impact of ai, chatgpt, and chatbot on education using machine learning algorithms. *Bulletin of Electrical Engineering and Informatics*, *13*(4), 2951–2958. https://doi.org/10.11591/EEI.V13I4.7158

Jones, M. L. (2004). Application of systematic review methods to qualitative research: practical issues. *Journal of Advanced Nursing*, *48*(3), 271–278. https://doi.org/10.1111/J.1365-2648.2004.03196.X

Kazemitabaar, M., Chow, J., Ma, C. K. T., Ericson, B. J., Weintrop, D., & Grossman, T. (2023). Studying the effect of ai code generators on supporting novice learners in introductory programming. *Conference on Human Factors in Computing Systems - Proceedings*. https://doi.org/10.1145/3544548.3580919

Kuhail, M. A., Mathew, S. S., Khalil, A., Berengueres, J., & Shah, S. J. H. (2024). "Will i be replaced?" assessing chatgpt's effect on software development and programmer perceptions of ai tools. *Science of Computer Programming*, *235*, 103111. https://doi.org/10.1016/J.SCICO.2024.103111

Kumar, Ch. N. S., Nukala, L. D., Aswani, P., & Sujana, Y. (2024). The impact of ai tools on employment trends and skill demands. *International Research Journal on Advanced Engineering Hub (IRJAEH)*, *2*(10), 2538–2545. https://doi.org/10.47392/IRJAEH.2024.0348

Lo, C. K. (2023). What is the impact of chatgpt on education? a rapid review of the literature. *Education Sciences*, *13*(4), 410. https://doi.org/10.3390/EDUCSCI13040410

Mailach, A., Gorgosch, D., Siegmund, N., & Siegmund, J. (2025). "Ok pal, we have to code that now": interaction patterns of programming beginners with a conversational chatbot. *Empirical Software Engineering*, *30*(1), 1–33. https://doi.org/10.1007/S10664-024-10561-6

Malik, N., Tripathi, S. N., Kar, A. K., & Gupta, S. (2022). Impact of artificial intelligence on employees working in industry 4.0 led organizations. *International Journal of Manpower*, *43*(2), 334–354. https://doi.org/10.1108/IJM-03-2021-0173

Mozannar, H., Fourney, A., Bansal, G., & Horvitz, E. (2024). Reading between the lines: modeling user behavior and costs in ai-assisted programming. *Conference on Human Factors in Computing Systems - Proceedings*, 16. https://doi.org/10.1145/3613904.3641936

Mutaqin, R., Subroto, D. E., & Warman, C. (2024). Analisis penggunaan chat gpt (ai) dan modul pemrograman terhadap motivasi belajar dan kreativitas mahasiswa dalam mata kuliah pemrograman pada program studi pendidikan teknologi informasi, universitas bina bangsa. *Jurnal Saintifik (Multi Science Journal)*, *22*(1), 1–8. https://doi.org/10.58222/JS.V22I1.214

Peng, S., Kalliamvakou, E., Cihon, P., & Demirer, M. (2023). *The impact of ai on developer productivity: evidence from github copilot*. https://doi.org/10.48550/arXiv.2302.06590

Rahman, M. M., & Watanobe, Y. (2023). Chatgpt for education and research: opportunities, threats, and strategies. *Applied Sciences 2023, Vol. 13, Page 5783*, *13*(9), 5783. https://doi.org/10.3390/APP13095783

Ray, B. (2023). Programming language processing: how ai can revolutionize software development? *ISEC '23: Proceedings of the 16th Innovations in Software Engineering Conference*, 1–1. https://doi.org/10.1145/3578527.3581766

Rebolledo, R., De La Vall, F., & González Araya, F. (2023). Exploring the benefits and challenges of ai-language learning tools. *International Journal of Social Sciences and Humanities Invention*, *10*(01), 7569–7576. https://doi.org/10.18535/IJSSHI/V10I01.02

Roach, J. (2022). *How ai makes developers' lives easier, and helps everybody learn to develop software*. News.microsoft.com.

Ross, S. I., Martinez, F., Houde, S., Muller, M., & Weisz, J. D. (2023). The programmer's assistant: conversational interaction with a large language model for software development. *International Conference on Intelligent User Interfaces, Proceedings IUI*, 491–514. https://doi.org/10.1145/3581641.3584037

Sestino, A., & De Mauro, A. (2022). Leveraging artificial intelligence in business: implications, applications and methods. *Technology Analysis and Strategic Management*, *34*(1), 16–29. https://doi.org/10.1080/09537325.2021.1883583

Sihananto, A. N., Atmaja, P. W., & Sugiarto, S. (2024). Pemanfaatan ai dalam pembelajaran pemrograman untuk mahasiswa. *Prosiding TAU SNARS-TEK Seminar Nasional Rekayasa Dan Teknologi*, *4*(1), 8–11. https://doi.org/10.47970/SNARSTEK.V2I1.706

Snyder, H. (2019). Literature review as a research methodology: An overview and guidelines. *Journal of Business Research*, *104*, 333–339. https://doi.org/10.1016/J.JBUSRES.2019.07.039

Sucipto, H. (2024). The impact of artificial intelligence (ai) on human resource management practices. *Management Studies and Business Journal (Productivity)*, *1*(1), 138–145. https://doi.org/10.62207/XEY9MX18

Weisz, J. D., Muller, M., Houde, S., Richards, J., Ross, S. I., Martinez, F., Agarwal, M., & Talamadupula, K. (2021). Perfection not required? human-ai partnerships in code translation. *International Conference on Intelligent User Interfaces, Proceedings IUI*, 402–412. https://doi.org/10.1145/3397481.3450656

Weisz, J. D., Muller, M., Ross, S. I., Martinez, F., Houde, S., Agarwal, M., Talamadupula, K., & Richards, J. T. (2022). Better together? an evaluation of ai-supported code translation. *International Conference on Intelligent User Interfaces, Proceedings IUI*, 369–391. https://doi.org/10.1145/3490099.3511157