GENERATING JAVA CODE WITH AI TOOLS. USAGE AND IMPLICATIONS

Alexandru TĂBUȘCĂ[1]

Andrei LUCHICI[2]

Mihai BOTEZATU[3]

Silvia TĂBUȘCĂ[4]

Abstract

Java programming, a high-level, general-purpose language renowned for its "Write Once, Run Anywhere" (WORA) capability, has (re)gained notable traction as developers increasingly integrate artificial intelligence (AI) generative tools into their workflows. Java's platform independence, robust security features, and extensive libraries have made it a preferred choice for a wide range of applications, from mobile apps to large-scale enterprise systems. The advent of AI generative tools, such as ChatGPT, GitHub Copilot or Amazon CodeWhisperer, has further enhanced Java programming by automating mundane tasks, improving code quality, and fostering creativity in the development process. In today's world, solid knowledge related to AI code generation tools is a must for all developers and software engineers. AI tools for generating Java code have also started an entire new set of debates related to copyright issues. Currently, the relevant legal frameworks, at international level, are not harmonized and in some cases even antagonistic.

Keywords: ai code generation; java ai tools; java; ai code legal issues; ai copyright issues

JEL Classification: C8, O31, O33, O34, O39

1. Introduction

Java programming, a high-level, general-purpose language renowned for its "Write Once, Run Anywhere" (WORA) capability, has gained notable traction as developers increasingly integrate artificial intelligence (AI) generative tools into their workflows. Java's platform independence, its' very robust security features, as well as the extensive libraries have made it an excellent and preferred choice for a very wide range of different applications, starting with mobile apps and going to large-scale enterprise systems. The advent of AI generative tools, such as GitHub Copilot and Amazon CodeWhisperer, has further enhanced Java

[1]PhD, Associate Professor, Romanian-American University, Romania, alex.tabusca@rau.ro, corresponding author
[2]PhD, Lecturer, Center for Research in AI, Romania, andrei.luchici@rau.ro
[3]PhD, Professor Habil, Romanian-American University, Romania, mihai.botezatu@rau.ro
[4]PhD, Lecturer, Center for Human Rights and Migration, Romania, silvia.tabusca@rau.ro

programming by automating mundane tasks, improving (usually) the quality of code, and increasing creativity within the entire software development process [1][2][3][4]. The integration of AI generative tools is reshaping the software development landscape, promoting efficiency and enabling developers to focus on complex problem-solving. These tools assist in various capacities, including code generation, debugging, and project documentation, thereby streamlining workflows and enhancing productivity [5][6][7]. Moreover, studies have shown a positive correlation between the use of these AI tools and perceived productivity among developers, as they consolidate multiple functions into a single application and provide quick access to critical information [5][8]. However, the incorporation of AI into Java programming is not without its challenges. Developers face issues related to the reliability of generated outputs, the need for precise prompting, and concerns regarding data security and governance. Ethical considerations, such as algorithmic bias and intellectual property rights, further complicate the landscape as the technology matures [9][10][11]. As the Java community continues to embrace AI solutions, it is imperative to address these challenges through transparency, ethical guidelines, and best practices to ensure that the advantages of AI can be fully realized without compromising quality or security [12][13][14]. Looking to the future, the collaboration between AI and developers is expected to deepen, with increasing use cases for AI in Java development, particularly in specialized areas such as natural language processing and dynamic pricing systems. As developers adapt to these evolving technologies, ongoing education and a commitment to ethical standards will be crucial in harnessing the full potential of AI while mitigating associated risks [15][16][14].

2. Java Programming Paradigm

Java is a high-level, general-purpose programming language that is both memory-safe and object-oriented, making it a prominent choice among software developers. Designed with the principle of "Write Once, Run Anywhere" (WORA), Java allows compiled code to run on any platform that supports Java without needing recompilation. This is achieved through the use of the Java Virtual Machine (JVM), which interprets Java bytecode across different operating systems and hardware architectures [1][2].

2.1. Features of Java

Object-Oriented Principles

Java is fundamentally an object-oriented programming (OOP) language, supporting core OOP concepts such as classes, objects, inheritance, encapsulation, and polymorphism. Unlike some other programming languages, all code in Java is encapsulated within classes, and every data item is treated as an object, except for the primitive data types which are optimized for performance [3][2]. The syntax of Java is influenced heavily by C and C++,

yet it avoids complex features like pointers and multiple inheritance, promoting simplicity and enhancing its usability for beginners [4][17].

Platform Independence

One of Java's most significant advantages is its platform independence. Java programs are first compiled into bytecode, a platform-neutral intermediate representation. This bytecode can run on any system equipped with a JVM, making Java applications remarkably versatile across different environments [3][2][17].

Performance and Scalability

Despite being an interpreted language, Java's performance is impressive, thanks in part to its Just-In-Time (JIT) compiler, which optimizes bytecode execution. Additionally, Java supports multithreading, allowing concurrent execution of code, which improves efficiency and resource management [17][18]. These features, combined with Java's inherent scalability, make it suitable for a wide range of applications from mobile devices to enterprise-level systems [4][17].

Security and Robustness

Java is designed with security in mind, incorporating features that protect against common programming errors, such as memory leaks and buffer overflows. Its strong type-checking at compile time and runtime contributes to its reliability and robustness, ensuring that Java applications are both secure and stable [3][2]. The Java ecosystem includes a rich set of libraries for various tasks, including internationalization, database connectivity, and remote method invocation, further enhancing its utility for developers [1][4].

Development Environment

The Java development environment is extensive, comprising several frameworks and libraries. The Java Standard Edition (Java SE) provides the core functionality, while Java Enterprise Edition (Java EE) extends this with capabilities for building large-scale, distributed applications [1]. Various tools and integrated development environments (IDEs) facilitate coding in Java, allowing developers to leverage the language's features effectively in their projects [30].

3. AI Generative Tools for Java Programming

AI generative tools have become instrumental in enhancing the efficiency and effectiveness of Java programming. These tools leverage artificial intelligence to assist developers in various capacities, streamlining workflows and improving code quality.

The adoption (more or less generalized at this time) of AI in code generation offers several benefits, including faster development cycles, improved productivity, and the ability to automate mundane tasks. Nevertheless, challenges persist, particularly concerning the

quality of the generated code and potential security risks. Developers must remain vigilant against issues such as security vulnerabilities that may arise from AI-generated outputs, which can lead to exploitable weaknesses in software systems [45][31]. Moreover, the copyright implications of using AI-generated code are under scrutiny. As AI tools produce outputs based on their training datasets—which may include copyrighted material - the legal status of the code generated raises questions about ownership and intellectual property rights [46][43]. The intersection of AI-generated content and copyright law continues to evolve, necessitating further examination of how these technologies interact with existing legal frameworks. The advent of artificial intelligence (AI) tools for code generation has prompted significant legal discourse surrounding copyright protections. Two primary issues stand out: the risk of copyright infringement and the difficulty in obtaining copyright protection for AI-generated content [35].

At the USA level, the U.S. Copyright Office maintains a firm stance that copyright law protects only works created with human authorship. This principle was emphasized in the 2025 Report on Copyright and Artificial Intelligence, which clarifies that works generated entirely by AI, devoid of meaningful human intervention, do not qualify for copyright protection [40][47]. Consequently, code produced solely by AI cannot be registered for copyright, placing it in a category that may be freely used by anyone unless protected under another legal framework, such as trade secrets [32][47]. However, when human developers engage in structuring, selecting, and refining AI-assisted outputs, they can establish sufficient human authorship to secure copyright for those contributions [40][47]. This distinction is critical for businesses that integrate AI tools into their development processes, as it underscores the importance of human involvement in obtaining copyright protections.

Uses of Generative AI Tools

Generative AI tools are utilized in multiple ways within the realm of software development.

- Generating and Reviewing Artifacts: AI tools facilitate the creation and refinement of project documents, including requirements specifications and design documents, ensuring accuracy and completeness in project deliverables [5].
- Supporting Ideation Processes: AI assists in brainstorming sessions and design thinking by generating novel ideas and concepts, thereby fostering creativity during the development phase [5].
- Resolving Doubts in Code Construction: Developers can leverage AI to troubleshoot and resolve technical issues encountered during programming, enhancing problem-solving capabilities [5].

Key Features of Generative AI Tools

Generative AI tools are equipped with advanced functionalities designed to aid developers in their coding tasks.

- Code Completion and Generation: Tools like GitHub Copilot, ChatGPT and Amazon CodeWhisperer enable developers to write code more efficiently by suggesting code completions and even entire functions based on their coding style and intent [19].
- Language Understanding: These tools utilize sophisticated natural language processing algorithms to interpret the intent behind code, allowing developers to work across various programming languages without hindrance [19].
- Integration with IDEs: Many generative AI tools seamlessly integrate with popular integrated development environments (IDEs) such as Visual Studio Code and IntelliJ, providing developers with immediate access to AI capabilities without disrupting their workflow [19].

Challenges of Generative AI Tools

Despite the advantages, the integration of generative AI tools in software development also presents several challenges. These include issues related to explainability and interpretability of AI decisions, as well as concerns regarding data governance and the quality of training data used in these models [9][20]. Addressing these challenges requires a comprehensive approach that emphasizes transparency and ethical guidelines in the use of generative AI technologies [12].

3.1 Integration of AI Generative Tools with Java

Overview of AI in Java Development

The integration of AI generative tools in Java programming is reshaping the software development landscape, enhancing productivity, and enabling developers to focus on more complex tasks. Generative AI tools such as Co-Pilot and Codex can generate Java code from natural language descriptions, significantly reducing the time developers spend on producing boilerplate code, thus improving overall efficiency and output quality [6][21]. As Java remains a leading platform for AI projects due to its flexibility, comprehensive libraries, and large-scale data processing capabilities, the collaboration between AI and Java is becoming increasingly vital [7][22].

At this moment, we can conclude that AI tools for code generation have significantly transformed the software development landscape, offering a new approach to writing and optimizing code. These intelligent systems leverage advanced technologies such as Artificial Intelligence (AI) and Machine Learning (ML) to assist developers by generating code based on natural language descriptions and existing code contexts.

Enhancements in Developer Productivity

Participants in studies have reported a positive correlation between the use of generative AI tools and their perceived productivity. These tools not only streamline workflows by consolidating multiple functions into a single application but also provide quick access to information that enhances team communication and collaboration [5]. Despite challenges related to reliability and the need for refinement of generated outputs, developers noted that generative AI tools facilitate gains in efficiency and flow, allowing them to produce relevant content such as code, reports, and design models more effectively [5][8].

Challenges in Integration

While generative AI tools present numerous advantages, developers also face challenges such as ensuring the reliability of generated outputs and the need for precise prompting to achieve accurate results. The absence of sources to validate the information provided by these tools further complicates their integration into sensitive projects [5][21]. Moreover, there are security concerns that can restrict the use of generative AI tools when dealing with sensitive data in software development contexts [5].

Future Directions and Best Practices

Looking ahead, the Java community is actively embracing AI-driven solutions, with a high percentage of developers expressing interest in building AI-powered applications using Java [22]. As Java developers prepare for the future, they emphasize the importance of essential elements like Retrieval-Augmented Generation (RAG) and embeddings for advanced knowledge retrieval [23][22]. These approaches will be crucial in developing practical, enterprise-ready features that are reliable and secure, enabling developers to navigate complex decision-making scenarios effectively.

How AI Code Generation Works

At its core, AI code generation involves using sophisticated algorithms that are trained on extensive datasets sourced from publicly available code repositories. Large Language Models (LLMs) employ Deep Learning (DL) techniques to understand programming patterns and best practices, enabling them to suggest or create code snippets that meet user-defined functionalities [42][43]. Developers input plain text prompts that outline their coding requirements, and the AI tools respond with relevant code suggestions or full functions, streamlining the development process and reducing manual effort [44][45].

Current AI code generation tools are capable of producing code in various programming languages based on natural language input. Developers can articulate the desired functionality, and the AI translates these instructions into contextually appropriate code, enhancing productivity and accessibility for programmers of all skill levels [42][44]. As already mentioned, notable tools in this domain include GitHub Copilot, Amazon Code

Whisperer or ChatGPT, which offer features such as code completion, snippets, and even entire functions to assist in the coding workflow [45][31].

In addition to generating code, AI tools also conduct automatic reviews of existing codebases to identify security vulnerabilities, bugs, and common coding errors. This process not only helps maintain code quality but also allows developers to save time by automating repetitive tasks that would otherwise require significant manual intervention [42][43]. However, it is important to note that the code generated by these tools should still undergo rigorous review by human developers to mitigate the risks associated with inaccuracies and security vulnerabilities [44][45].

4. Case Studies and Examples

Pilot Case Study on Generative AI in Software Development

A pilot case study was conducted to investigate the impact of generative AI tools on software development productivity. The study focused on a single company and involved various software development roles, including developers, quality assurance (QA) professionals, and designers. The primary objectives were to explore how different professionals utilize generative AI tools in their tasks and to understand the tools' perceived effects on productivity across diverse project configurations and methodologies [5].

Methodology and Data Collection

The research utilized a case study methodology that involved questionnaires with open-ended questions and observations. The questionnaires aimed to gather insights from participants who volunteered to use generative AI tools, ensuring minimal disruption to their work routines [5]. Observational data from company communication channels, such as Slack, were also collected to identify potential participants and assess discussions surrounding generative AI [5].

Findings on Perceived Productivity

Participants reported a positive effect of generative AI tools on their perceived productivity, particularly in terms of efficiency gains across their software development activities. The tools facilitated time optimization by consolidating multiple individual tools into a single streamlined workflow. Despite facing some challenges, such as concerns over reliability, most software professionals acknowledged improvements in their ability to create relevant and insightful outputs, including code, design models, and documentation [5].

Indirect Impacts on Communication and Collaboration

While not explicitly stated by the participants, the use of generative AI tools appeared to indirectly enhance communication and collaboration within development teams. By providing quick access to information and facilitating knowledge sharing, these tools allowed team members to align their understanding and work towards common goals more effectively [5].

Broader Implications and Applications of Generative AI Tools

Generative AI tools offer a variety of applications that can significantly aid software developers. These applications include refactoring and code improvement, documentation generation, learning assistance, prototype development, and data transformation and analysis [24]. For example, developers can use AI to automate data cleansing tasks or rapidly develop proof-of-concept applications, thereby reducing the upfront investment of time and resources [24]. Bogdan Mykhaylovych, Technical Director at Softjourn, emphasized the importance of integrating tailored AI models into specific business domains to further enhance productivity for developers and product owners [24]. This integration highlights the adaptability of generative AI tools across various aspects of the software development lifecycle, reinforcing their potential to redefine traditional programming methodologies.

Ethical Considerations

The integration of AI generative tools into Java programming raises several ethical concerns, particularly regarding ownership and originality in AI-generated code. Issues related to intellectual property rights and copyright are increasingly becoming significant in discussions about the ethical use of such technologies [11].

Algorithmic Bias

One of the most pressing ethical considerations in machine learning and AI is algorithmic bias, which refers to systematic errors that lead to discrimination against certain groups based on the outputs of a program [10][25]. This bias can arise when training datasets are unrepresentative, leading the AI to produce results that favor specific demographics over others [26]. For instance, algorithms may exhibit racial bias in healthcare risk assessments or show discrimination in hiring practices, as evidenced by cases involving biased recruitment tools [27].

Mitigating Bias

To address algorithmic bias, developers and operators of AI algorithms are urged to implement best practices at various stages of the AI system's lifecycle [28]. This includes examining the training data for representation and ensuring that the model does not perpetuate existing inequalities. Additionally, efforts such as using AI Fairness 360, Fairlearn, and other fairness tools can help identify and reduce bias in AI systems [27].

The Role of Policymakers

Policymakers also play a critical role in addressing these issues. The establishment of regulatory sandboxes could provide a framework for testing AI technologies while developing appropriate regulations to curb biases [13]. Such environments allow for innovation in technology alongside regulatory evolution, helping ensure that ethical standards are upheld without stifling technological advancement.

Broader Ethical Implications

Beyond technical concerns, the ethical implications of AI-generated tools extend to their societal impacts. Developers must consider how their models may influence individuals and communities, particularly in high-stakes areas like job applicant suitability assessments [11]. Discussions surrounding the ethical framework of machine learning emphasize the need for guardrails to prevent the amplification of systemic discrimination and other negative consequences [13].

5. Copyright Issues for Java Code Generated with AI Tools

The copyright implications of Java code generated by artificial intelligence (AI) tools have sparked significant legal debate as the integration of these technologies into software development raises questions about authorship and intellectual property rights. Historically, U.S. copyright law mandates that only works created by human beings are eligible for protection, a principle reaffirmed by the U.S. Copyright Office and recent court rulings such as Thaler v. Perlmutter, which ruled that AI-generated works without meaningful human involvement cannot be copyrighted [31][32][33].

As a result, developers and companies face considerable risks when utilizing AI-generated code, as such outputs may not only be unprotected by copyright but could also infringe on existing copyrighted works. The uniqueness of the situation is compounded by the nature of AI code generation tools, which rely on vast datasets that may include copyrighted material. This raises the potential for legal conflicts when companies modify or deploy AI-generated code, as they could inadvertently violate the rights of original copyright holders [34][32].

Legal experts emphasize that businesses must navigate these complexities carefully, documenting their creative processes and consulting with legal professionals to mitigate infringement risks and ensure compliance with copyright law [35][36]. Moreover, the evolving legal landscape surrounding AI-generated content necessitates ongoing discourse on the definition of authorship and the applicability of copyright in the digital age. Current legislative efforts and high-profile lawsuits are beginning to address these challenges, but many uncertainties remain. The juxtaposition of rapid technological advancement with existing copyright frameworks highlights the need for reform to address the implications of generative AI on intellectual property rights, particularly in the realm of software development [37][38][39].

The use of AI in generating code also introduces substantial copyright infringement risks. As copyright automatically applies to original source code, modifying existing works with AI could infringe on the exclusive rights held by the original copyright owner [34]. Businesses must seek permission from copyright holders when using AI to modify protected works, much like any traditional method of content alteration. Additionally, the landscape of copyright law necessitates that organizations document their independent creation processes and consult legal professionals about copyright risks before reusing code. This proactive approach is essential given the complex nature of fair use determinations and the legal ramifications of infringement claims [36]. The current legal framework, in all countries, requires businesses to critically assess their reliance on AI in creative fields and to navigate the copyright landscape with very much care. While the US Copyright Office and other similar bodies worldwide have yet to provide definitive guidance on the future of copyright in relation to AI-generated works, it has been made clear that human authorship remains a cornerstone of copyright protection [32].

The legal landscape regarding the copyright of AI-generated works, particularly Java code, is evolving as courts address the implications of artificial intelligence on existing copyright law. As of now, the legality of using copyrighted content to train AI models remains unsettled, with outcomes heavily reliant on jurisdiction and specific case circumstances [37][48]. This uncertainty has led to a growing demand for legislative or regulatory solutions to clarify when training on copyrighted content is permissible [37]. This places an added responsibility on developers and companies to ensure that their use of AI tools complies with copyright law to mitigate potential legal and financial repercussions. The U.S. Copyright Office has released reports addressing the copyrightability of outputs created using generative AI, affirming that existing copyright principles can accommodate these new technologies. The Office maintains that AI outputs may be copyrightable when a human author contributes significant creative elements, such as perceptible modifications or creative arrangements, rather than merely providing prompts [41][50]. This stance sets a high bar for the recognition of AI-generated works under copyright law, leaving many

boundaries regarding protectable and unprotectable works still undefined [50]. Courts have also pointed out that AI-generated outputs could infringe copyright if they closely resemble existing works, raising significant legal questions regarding originality and authorship [49][50].

As AI tools continue to transform coding practices, their legal and ethical ramifications warrant thorough examination. The lack of a clear path to copyright protection for AI-generated Java code emphasizes the importance of human involvement in the creative process to secure intellectual property rights. The outcome of ongoing legal battles and potential legislative changes will be pivotal in shaping the future of copyright in relation to AI technologies, ultimately influencing how developers and businesses engage with these innovative tools [40][37][41].

5.1 The Impact of Generative AI on Copyright

The advent of generative artificial intelligence (AI) technologies has raised new challenges for traditional notions of authorship in copyright law. Current interpretations of copyright require that an original work of authorship must be created by a human being, a stance upheld by the U.S. Copyright Office [32]. Recent court rulings in the USA, such as Thaler v. Perlmutter, have reinforced this position, stating that works generated by AI without human intervention do not qualify for copyright protection [33]. This limitation has significant implications for the use of AI in creative domains, particularly when businesses aim to utilize AI-generated content, as they may not enjoy the typical intellectual property protections if such content is challenged by rivals [33].

5.2 Current Legislative Landscape and Ongoing Legal Battles

The legislative landscape concerning copyright and AI-generated Java code is rapidly evolving as governments worldwide grapple with the implications of generative AI technologies. In March 2024, the European Parliament approved the EU AI Act, which mandates that providers of general-purpose AI models comply with existing copyright laws, particularly regarding the reservation of rights for training data [39]. This act signifies a proactive approach by the EU in addressing copyright concerns related to AI, emphasizing the need for transparency and compliance with copyright norms. In the USA, various states have introduced their own legislation targeting generative AI, with notable bills emerging in Colorado and California in 2024. These state-level initiatives reflect a recognition of the need to establish a regulatory framework that balances innovation with the protection of intellectual property rights [39]. However, the U.S. federal stance remains somewhat

ambiguous, with courts and legislators still deliberating on the copyrightability of AI-generated content [53].

The approaches to copyright for AI-generated works differ significantly among jurisdictions. In the U.S., the Copyright Office maintains a stringent requirement for human authorship, stating that copyright entitlement can only be established if the work originates from a human agent [33]. This is particularly relevant to Java code generated by AI tools, as the lack of a human author would preclude such code from receiving copyright protection under current U.S. law [34]. Conversely, the EU has initiated frameworks that could potentially allow for broader interpretations of AI-generated works. The recent introduction of exceptions for text and data mining (TDM) indicates a willingness to adapt existing laws to facilitate AI development while safeguarding creator rights [37]. This distinction highlights a significant divergence from U.S. legislation, where the fair use doctrine provides a more flexible, albeit less clear-cut, framework for utilizing copyrighted material for AI training.

Experts advocate for international harmonization of copyright laws concerning AI, proposing common principles that could facilitate a global approach to managing AI-generated content [37]. Such harmonization could involve agreements that permit the use of works for AI training under specific conditions, thereby balancing the interests of creators with the need to foster innovation in AI technologies. One suggestion includes establishing a global data licensing framework that allows AI firms to contribute to a fund distributing payments to content creators in exchange for licenses to train on published content [37]. The ongoing discussions around AI and copyright reflect a critical juncture for policymakers. Achieving a consensus that aligns with technological advancements while protecting the rights of creators will likely require extensive international dialogue, possibly facilitated through organizations like the World Intellectual Property Organization (WIPO) or within trade agreements. The outcome of these discussions will be pivotal in shaping the future of copyright as it relates to AI-generated Java code and other creative works.

As AI continues to advance, high-profile lawsuits have emerged against companies like OpenAI and Stability AI, with plaintiffs alleging unauthorized use of copyrighted material to train AI models. The ongoing case of *New York Times Co. v. Microsoft* serves as a critical example of how courts are grappling with these issues, setting a precedent for the intersection of copyright law and AI technology. These cases could have far-reaching implications for the future of copyright and AI-generated content, as the outcomes may influence both the interpretation of fair use and the rights of creators in the digital age [38][51][52].

5.3 Implications for Java Code Generated with AI

When considering Java code generated by AI tools, the existing legal framework suggests that if the code lacks direct human authorship, it may not be eligible for copyright protection. This situation necessitates that developers and businesses exercise caution, as using or modifying AI-generated code could potentially infringe upon the copyright rights of existing works, unless proper permissions are obtained from the original copyright holders [34]. As the legal landscape continues to evolve alongside advancements in AI technologies, ongoing discourse regarding the definitions of authorship and the applicability of copyright law will be crucial in shaping the future of intellectual property rights in the realm of software development and beyond.

Looking ahead, it is anticipated that a more settled equilibrium will emerge between AI developers, content creators, and consumers. This will likely involve clearer guidelines on permissible data usage for AI training, ensuring that creators can benefit from AI utilization without facing undue harm. Additionally, it is crucial to foster an environment where diverse creative content continues to thrive, balancing innovation with the protection of human authorship [37]. Ultimately, while challenges exist, the potential for AI to enhance programming and creativity remains substantial, provided that stakeholders actively engage with the legal and ethical implications [19].

6. Conclusions and Future Trends

The landscape of Java programming with AI generative tools is poised for significant evolution in the coming years. As the integration of AI technologies into software development becomes more prevalent, we can expect several transformative trends to emerge.

Enhanced Collaboration Between AI and Developers

Rather than viewing AI as a competitor, the future of programming will increasingly focus on collaboration. Developers will leverage AI tools to automate mundane tasks such as code generation, debugging, and testing, allowing them to concentrate on more complex problem-solving aspects of software development [15][29]. This symbiotic relationship is anticipated to boost productivity, with AI tools offering developers real-time information and support during coding [8].

Expanding Use Cases for AI in Java Development

AI's capabilities will likely extend into more specialized areas of Java development. Use cases such as natural language processing (NLP) and dynamic pricing systems will benefit

from advanced AI models embedded in Java applications, enhancing their functionality and efficiency [22]. The integration of AI frameworks into existing systems will also enable developers to implement machine learning models more effectively, addressing complex challenges like customer churn prediction and recommendation systems.

Addressing Ethical and Security Challenges

As AI becomes integral to coding practices, addressing ethical and security concerns will be paramount. Developers will need to navigate potential pitfalls related to security vulnerabilities, intellectual property issues, and code quality management [14]. Robust risk management strategies will be essential as organizations seek to balance the advantages of AI tools with the associated risks.

Continuous Learning and Adaptation

The rapid pace of AI development necessitates continuous learning and adaptability among programmers. Professionals will need to stay updated on evolving AI technologies and methodologies to maximize their effectiveness in collaboration with AI tools [16][29]. Educational institutions and training programs are likely to adapt their curricula to incorporate AI literacy, equipping future developers with the necessary skills to thrive in this changing environment.

Acknowledgment

The paper is based on research carried out in part within the Center for Computational Science and Machine Intelligence (CSMI) of the Romanian-American University's School of Computer Science for Business Management, and the Center for Human Rights and Migration (CDOM) of the Romanian-American University's School of Law.

References

[1] Java (programming language) – Wikipedia; https://en.wikipedia.org/wiki/Java_(programming_language); last access: May 04, 2025

[2] Java Features; March 24, 2025; https://www.geeksforgeeks.org/java-features/; last access: May 04, 2025

[3] How Java Is Different From Other Programming Languages; https://www.fita.in/how-java-is-different-from-other-programming-languages/; last access: May 04, 2025

[4] Bello Adeshola, Amorighoye Otse; Java Programming Language; https://amorserv.com/insights/java-programming-language

[5] Kumar Vinod; Exploring the Key Features of Java Programming Language; https://medium.com/@vinodkumarbheel61/exploring-the-key-features-of-java-programming-language-315f92e58869

[6] Bloch-Wehba Hannah; Transparency's AI Problem; June 17, 2021; https://knightcolumbia.org/content/transparencys-ai-problem

[7] Coutinho Mariana, Marques Lorena, Santos Anderson, Dahia Marcio, Franca Cesar, de Souza Santos Ronnie; The Role of Generative AI in Software Development Productivity: A Pilot Case Study; Proceedings of the 32nd ACM Symposium on the Foundations of Software Engineering (FSE '24), November 15–19, 2024, Porto de Galinhas, Brazil; https://arxiv.org/html/2406.00560v1

[8] Generative AI Tools for Software Development | Coding Temple; https://www.codingtemple.com/blog/generative-ai-tools-for-software-development/; last access: May 04, 2025

[9] AI Transparency: Fundamental pillar for ethical and safe AI; April 08, 2025; https://www.plainconcepts.com/ai-transparency/; last access: May 04, 2025

[10] Hassani Hesam Sheikh; AI Governance: Frameworks, Tools, Best Practices; September 04, 2024; https://www.datacamp.com/blog/ai-governance

[11] Ruiz Julie Tyler; Generative AI Ethics: AI Risks, Benefits, and Best Practices; October 27, 2024; https://www.coursera.org/articles/generative-ai-ethics

[12] Laforge Guillaume; Generative AI text generation in Java with PaLM and LangChain4J; December 04, 2023; https://codelabs.developers.google.com/codelabs/genai-text-gen-java-palm-langchain4j

[13] The Impact of Generative AI on Java Development: From Code to Deployment; November 11, 2024; https://medium.com/@speckyfox.technologies/the-impact-of-generative-ai-on-java-development-from-code-to-deployment-8d1e3d2a35d8; last access: May 04, 2025

[14] The Future of Java Development: 13 AI Tools Transforming Productivity; January 15; 2025; https://groovetechnology.com/blog/technologies/ai-for-java-developers/; last access: May 04, 2025

[15] Sakovich Natallia; How to Implement AI in Java: A Step-by-Step Guide; May 20, 2025 https://sam-solutions.com/blog/how-to-implement-ai-in-java/

[16] The Power of AI-Assisted Coding: Streamlining Rapid Development in Java; September 11, 2024; https://blog.kodezi.com/the-power-of-ai-assisted-coding-streamlining-rapid-development-in-java/; last access: May 04, 2025

[17] Selvasingh Asir; The State of Coding the Future with Java and AI; May 12, 2025; https://devblogs.microsoft.com/java/the-state-of-coding-the-future-with-java-and-ai/

[18] Neville Meghan; Can AI Really Code? Understanding the Strengths and Limitations of AI Coding Tools; April 25, 2025; https://softjourn.com/insights/ai-coding-tools

[19] Nasimov Salmany; Taming the code generation beast — How responsible is your AI adoption in Java; October 29, 2024; https://digma.ai/taming-the-code-generation-beast-how-responsible-is-your-ai-adoption-in-java/

[20] Chen, Z.; Ethics and discrimination in artificial intelligence-enabled recruitment practices. Humanit Soc Sci Commun 10, 567 (2023). https://doi.org/10.1057/s41599-023-02079-x; https://www.nature.com/articles/s41599-023-02079-x

[21] Wood Maria; What Are the Risks of Algorithmic Bias in Higher Education?; May 2021; https://www.everylearnereverywhere.org/blog/what-are-the-risks-of-algorithmic-bias-in-higher-education/

[22] Levin Cait; AI Prompt Engineering: How to Instruct AI Models to Avoid AI Bias and Get The Results You Want; September 4, 2024; https://www.glideapps.com/blog/ai-prompt-engineering

[23] Sinha Swarnabha; How to Reduce Bias in AI? - Scaler Topics; December 8, 2023; https://www.scaler.com/topics/artificial-intelligence-tutorial/ai-bias/

[24] Awan Abid Ali;l What is Algorithmic Bias?; July 17, 2023; https://www.datacamp.com/blog/what-is-algorithmic-bias

[25] Turner Lee Nicol, Resnick Paul, and Barton Genie; Algorithmic bias detection and mitigation: Best practices and policies to reduce consumer harms; May 22, 2019; https://www.brookings.edu/articles/algorithmic-bias-detection-and-mitigation-best-practices-and-policies-to-reduce-consumer-harms/

[26] Siedsma Andrea; Will AI Replace Programmers? Navigating the Future of Coding; March 22, 2024; https://extendedstudies.ucsd.edu/news-events/extended-studies-blog/will-ai-replace-programmers-navigating-the-future-of-coding

[27] Jasińska Dorota, Puczyk Andrzej; Top Generative AI Platforms and Tools for Software Development; May 07, 2025; https://neontri.com/blog/generative-ai-platforms/

[28] Zaman Sian; December 29. 2024; Risks Of Using AI In Software Development - Is It All Bad?; https://impalaintech.com/blog/risks-of-ai-software-development/

[29] Zeichick Alan; 7 Ways GenAI Can Help Improve Software Development; October 8, 2024; https://www.oracle.com/artificial-intelligence/generative-ai/generative-ai-software-development/

[30] 10 Best Java IDE For Developers in 2025; December 13, 2024; https://www.geeksforgeeks.org/best-java-ide-for-developers/; last access: May 04, 2025

[31] Caldwell Mackenzie; What Is an "Author"?-Copyright Authorship of AI Art Through a Philosophical Lens; Houston Law Review, Vol. 61, Issue 2, 2023; https://houstonlawreview.org/article/92132-what-is-an-author-copyright-authorship-of-ai-art-through-a-philosophical-lens

[32] Artificial Intelligence; January 30, 2025; https://libguides.washburn.edu/c.php?g=1377087&p=10201459/; last access: May 04, 2025

[33] Laney Douglas B.; Copyright or Copywrong? AI's Intellectual Property Conundrum; Forbes Contributors, February 11, 2025; https://www.forbes.com/sites/douglaslaney/2025/02/11/copyright-or-copywrong-ais-intellectual-property-paradox/

[34] Vana Jim; Court Rules AI Can't Author a Copyrighted Work; March 19, 2025; https://www.schwabe.com/publication/court-rules-ai-cant-author-a-copyrighted-work/

[35] What is AI code generation?; https://aws.amazon.com/what-is/ai-coding/; last access: May 04, 2025

[36] James Kay; How to Use AI Code Generation to Enhance Developer Productivity; https://www.getambassador.io/blog/ai-code-generation-boost-developer-productivity

[37] AI code generation; https://www.sonarsource.com/learn/ai-code-generation/; last access: May 04, 2025

[38] AI Code Generation: The Risks and Benefits of AI in Software; https://www.legitsecurity.com/aspm-knowledge-base/ai-code-generation-benefits-and-risks; last access: May 04, 2025

[39] Artificial intelligence and copyright; https://en.wikipedia.org/wiki/Artificial_intelligence_and_copyright; last access: May 04, 2025

[40] Pelkonen Tuomas; Legal Insights into AI-Driven Game Development; October 16, 2024; https://nordialaw.com/legal-insights-into-ai-driven-game-development/

[41] Mathur Atreya; Recent Developments in AI, Art & Copyright: Copyright Office Report & New Registrations; March 4, 2025; https://itsartlaw.org/2025/03/04/recent-developments-in-ai-art-copyright-copyright-office-report-new-registrations/

[42] Furdock Brandon; Navigating the Legal Landscape of AI-Generated Code: Ownership and Liability Challenges; May 19, 2025; https://www.mbhb.com/intelligence/snippets/navigating-the-legal-landscape-of-ai-generated-code-ownership-and-liability-challenges/

[43] Sumner Michael; Protecting Source Code from Copyright Infringement: A Guide; February 09, 2025; https://www.scoredetect.com/blog/posts/protecting-source-code-from-copyright-infringement-a-guide

[44] Masood Adnan; Intellectual Property Rights and AI-Generated Content — Issues in Human Authorship, Fair Use Doctrine, and Output Liability; April 04, 2025; https://medium.com/@adnanmasood/intellectual-property-rights-and-ai-generated-content-issues-in-human-authorship-fair-use-8c7ec9d6fdc3

[45] Gewirtz David; If ChatGPT produces AI-generated code for your app, who does it really belong to?; December 21, 2024; https://www.zdnet.com/article/if-chatgpt-produces-ai-generated-code-for-your-app-who-does-it-really-belong-to/

[46] Zirpoli Christopher T.; Generative Artificial Intelligence and Copyright Law; September 29, 2023; https://www.congress.gov/crs-product/LSB10922

[47] Oratz Lisa T.; Copyright Office Solidifies Stance on the Copyrightability of AI-Generated Works; March 19, 2025; https://perkinscoie.com/insights/update/copyright-office-solidifies-stance-copyrightability-ai-generated-works

[48] Copyright Office Releases Part 2 of Artificial Intelligence Report; https://www.copyright.gov/newsnet/2025/1060.html; last access: May 04, 2025

[49] Artificial Intelligence and the Creative Double Bind - Chapter 2; Harvard Law Review; Vol. 138, Issue 6, April 2025; https://harvardlawreview.org/print/vol-138/artificial-intelligence-and-the-creative-double-bind/

[50] Klosek Katherine, Blumenthal Marjory S.; Training Generative AI Models on Copyrighted Works Is Fair Use; January 23, 2025; https://www.arl.org/blog/training-generative-ai-models-on-copyrighted-works-is-fair-use/

[51] Kim Rachel; AI and Copyright in 2023: In the Courts; January 4, 2024; https://copyrightalliance.org/ai-copyright-courts/

[52] Stalnaker Trevor, Wintersgill Nathan, Chaparro Oscar, Heymann Laura A., Di Penta Massimiliano, German Daniel M., Poshyvanyk Denys; Developer Perspectives on Licensing and Copyright Issues Arising from Generative AI for Coding; November 2024; https://arxiv.org/html/2411.10877v1

[53] Bondari Negar; AI, Copyright, and the Law: The Ongoing Battle Over Intellectual Property Rights; February 04, 2025; https://sites.usc.edu/iptls/2025/02/04/ai-copyright-and-the-law-the-ongoing-battle-over-intellectual-property-rights/

Bibliography

Awan Abid Ali;l What is Algorithmic Bias?; July 17, 2023; https://www.datacamp.com/blog/what-is-algorithmic-bias

Bello Adeshola, Amorighoye Otse; Java Programming Language; https://amorserv.com/insights/java-programming-language

Bloch-Wehba Hannah; Transparency's AI Problem; June 17, 2021; https://knightcolumbia.org/content/transparencys-ai-problem

Bondari Negar; AI, Copyright, and the Law: The Ongoing Battle Over Intellectual Property Rights; February 04, 2025; https://sites.usc.edu/iptls/2025/02/04/ai-copyright-and-the-law-the-ongoing-battle-over-intellectual-property-rights/

Caldwell Mackenzie; What Is an "Author"?-Copyright Authorship of AI Art Through a Philosophical Lens; Houston Law Review, Vol. 61, Issue 2, 2023; https://houstonlawreview.org/article/92132-what-is-an-author-copyright-authorship-of-ai-art-through-a-philosophical-lens

Chen, Z.; Ethics and discrimination in artificial intelligence-enabled recruitment practices. Humanit Soc Sci Commun 10, 567 (2023). https://doi.org/10.1057/s41599-023-02079-x; https://www.nature.com/articles/s41599-023-02079-x

Coutinho Mariana, Marques Lorena, Santos Anderson, Dahia Marcio, Franca Cesar, de Souza Santos Ronnie; The Role of Generative AI in Software Development Productivity: A Pilot Case Study; Proceedings of the 32nd ACM Symposium on the Foundations of Software Engineering (FSE '24), November 15–19, 2024, Porto de Galinhas, Brazil; https://arxiv.org/html/2406.00560v1

Furdock Brandon; Navigating the Legal Landscape of AI-Generated Code: Ownership and Liability Challenges; May 19, 2025; https://www.mbhb.com/intelligence/snippets/navigating-the-legal-landscape-of-ai-generated-code-ownership-and-liability-challenges/

Gewirtz David; If ChatGPT produces AI-generated code for your app, who does it really belong to?; December 21, 2024; https://www.zdnet.com/article/if-chatgpt-produces-ai-generated-code-for-your-app-who-does-it-really-belong-to/

Hassani Hesam Sheikh; AI Governance: Frameworks, Tools, Best Practices; September 04, 2024; https://www.datacamp.com/blog/ai-governance

James Kay; How to Use AI Code Generation to Enhance Developer Productivity; https://www.getambassador.io/blog/ai-code-generation-boost-developer-productivity

Jasińska Dorota, Puczyk Andrzej; Top Generative AI Platforms and Tools for Software Development; May 07, 2025; https://neontri.com/blog/generative-ai-platforms/

Kim Rachel; AI and Copyright in 2023: In the Courts; January 4, 2024; https://copyrightalliance.org/ai-copyright-courts/

Klosek Katherine, Blumenthal Marjory S.; Training Generative AI Models on Copyrighted Works Is Fair Use; January 23, 2025; https://www.arl.org/blog/training-generative-ai-models-on-copyrighted-works-is-fair-use/

Kumar Vinod; Exploring the Key Features of Java Programming Language; https://medium.com/@vinodkumarbheel61/exploring-the-key-features-of-java-programming-language-315f92e58869

Laforge Guillaume; Generative AI text generation in Java with PaLM and LangChain4J; December 04, 2023; https://codelabs.developers.google.com/codelabs/genai-text-gen-java-palm-langchain4j

Laney Douglas B.; Copyright Or Copywrong? AI's Intellectual Property Conundrum; Forbes Contributors, February 11, 2025; https://www.forbes.com/sites/douglaslaney/2025/02/11/copyright-or-copywrong-ais-intellectual-property-paradox/

Levin Cait; AI Prompt Engineering: How to Instruct AI Models to Avoid AI Bias and Get The Results You Want; September 4, 2024; https://www.glideapps.com/blog/ai-prompt-engineering

Masood Adnan; Intellectual Property Rights and AI-Generated Content — Issues in Human Authorship, Fair Use Doctrine, and Output Liability; April 04, 2025; https://medium.com/@adnanmasood/intellectual-property-rights-and-ai-generated-content-issues-in-human-authorship-fair-use-8c7ec9d6fdc3

Mathur Atreya; Recent Developments in AI, Art & Copyright: Copyright Office Report & New Registrations; March 4, 2025; https://itsartlaw.org/2025/03/04/recent-developments-in-ai-art-copyright-copyright-office-report-new-registrations/

Nasimov Salmany; Taming the code generation beast — How responsible is your AI adoption in Java; October 29, 2024; https://digma.ai/taming-the-code-generation-beast-how-responsible-is-your-ai-adoption-in-java/

Neville Meghan; Can AI Really Code? Understanding the Strengths and Limitations of AI Coding Tools; April 25, 2025; https://softjourn.com/insights/ai-coding-tools

Oratz Lisa T.; Copyright Office Solidifies Stance on the Copyrightability of AI-Generated Works; March 19, 2025; https://perkinscoie.com/insights/update/copyright-office-solidifies-stance-copyrightability-ai-generated-works

Pelkonen Tuomas; Legal Insights into AI-Driven Game Development; October 16, 2024; https://nordialaw.com/legal-insights-into-ai-driven-game-development/

Ruiz Julie Tyler; Generative AI Ethics: AI Risks, Benefits, and Best Practices; October 27, 2024; https://www.coursera.org/articles/generative-ai-ethics

Sakovich Natallia; How to Implement AI in Java: A Step-by-Step Guide; May 20, 2025 https://sam-solutions.com/blog/how-to-implement-ai-in-java/

Selvasingh Asir; The State of Coding the Future with Java and AI; May 12, 2025; https://devblogs.microsoft.com/java/the-state-of-coding-the-future-with-java-and-ai/

Siedsma Andrea; Will AI Replace Programmers? Navigating the Future of Coding; March 22, 2024; https://extendedstudies.ucsd.edu/news-events/extended-studies-blog/will-ai-replace-programmers-navigating-the-future-of-coding

Sinha Swarnabha; How to Reduce Bias in AI? - Scaler Topics; December 8, 2023; https://www.scaler.com/topics/artificial-intelligence-tutorial/ai-bias/

Sumner Michael; Protecting Source Code from Copyright Infringement: A Guide; February 09, 2025; https://www.scoredetect.com/blog/posts/protecting-source-code-from-copyright-infringement-a-guide

Stalnaker Trevor, Wintersgill Nathan, Chaparro Oscar, Heymann Laura A., Di Penta Massimiliano, German Daniel M., Poshyvanyk Denys; Developer Perspectives on Licensing and Copyright Issues Arising from Generative AI for Coding; November 2024; https://arxiv.org/html/2411.10877v1

Turner Lee Nicol, Resnick Paul, and Barton Genie; Algorithmic bias detection and mitigation: Best practices and policies to reduce consumer harms; May 22, 2019; https://www.brookings.edu/articles/algorithmic-bias-detection-and-mitigation-best-practices-and-policies-to-reduce-consumer-harms/

Vana Jim; Court Rules AI Can't Author a Copyrighted Work; March 19, 2025; https://www.schwabe.com/publication/court-rules-ai-cant-author-a-copyrighted-work/

Wood Maria; What Are the Risks of Algorithmic Bias in Higher Education?; May 2021; https://www.everylearnereverywhere.org/blog/what-are-the-risks-of-algorithmic-bias-in-higher-education/

Zaman Sian; December 29. 2024; Risks Of Using AI In Software Development - Is It All Bad?; https://impalaintech.com/blog/risks-of-ai-software-development/

Zeichick Alan; 7 Ways GenAI Can Help Improve Software Development; October 8, 2024; https://www.oracle.com/artificial-intelligence/generative-ai/generative-ai-software-development/

Zirpoli Christopher T.; Generative Artificial Intelligence and Copyright Law; September 29, 2023; https://www.congress.gov/crs-product/LSB10922

*** 10 Best Java IDE For Developers in 2025; December 13, 2024; https://www.geeksforgeeks.org/best-java-ide-for-developers/; last access: May 04, 2025

*** AI code generation; https://www.sonarsource.com/learn/ai-code-generation/; last access: May 04, 2025

*** AI Code Generation: The Risks and Benefits of AI in Software; https://www.legitsecurity.com/aspm-knowledge-base/ai-code-generation-benefits-and-risks; last access: May 04, 2025

*** AI Transparency: Fundamental pillar for ethical and safe AI; April 08, 2025; https://www.plainconcepts.com/ai-transparency/; last access: May 04, 2025

*** Artificial Intelligence; January 30, 2025; https://libguides.washburn.edu/c.php?g=1377087&p=10201459/; last access: May 04, 2025

*** Artificial intelligence and copyright; https://en.wikipedia.org/wiki/Artificial_intelligence_and_copyright; last access: May 04, 2025

*** Artificial Intelligence and the Creative Double Bind - Chapter 2; Harvard Law Review; Vol. 138, Issue 6, April 2025; https://harvardlawreview.org/print/vol-138/artificial-intelligence-and-the-creative-double-bind/

*** Copyright Office Releases Part 2 of Artificial Intelligence Report; https://www.copyright.gov/newsnet/2025/1060.html; last access: May 04, 2025

*** Generative AI Tools for Software Development | Coding Temple; https://www.codingtemple.com/blog/generative-ai-tools-for-software-development/; last access: May 04, 2025

*** How Java Is Different From Other Programming Languages; https://www.fita.in/how-java-is-different-from-other-programming-languages/; last access: May 04, 2025

*** Java (programming language) – Wikipedia; https://en.wikipedia.org/wiki/Java_(programming_language); last access: May 04, 2025

*** Java Features; March 24, 2025; https://www.geeksforgeeks.org/java-features/; last access: May 04, 2025

*** The Future of Java Development: 13 AI Tools Transforming Productivity; January 15; 2025; https://groovetechnology.com/blog/technologies/ai-for-java-developers/; last access: May 04, 2025

*** The Impact of Generative AI on Java Development: From Code to Deployment; November 11, 2024; https://medium.com/@speckyfox.technologies/the-impact-of-generative-ai-on-java-development-from-code-to-deployment-8d1e3d2a35d8; last access: May 04, 2025

*** The Power of AI-Assisted Coding: Streamlining Rapid Development in Java; September 11, 2024; https://blog.kodezi.com/the-power-of-ai-assisted-coding-streamlining-rapid-development-in-java/; last access: May 04, 2025

*** What is AI code generation?; https://aws.amazon.com/what-is/ai-coding/; last access: May 04, 2025