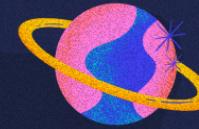




REALIZAÇÃO



## INTELIGÊNCIA ARTIFICIAL E DESENVOLVIMENTO DE SOFTWARE: Como ferramentas generativas afetam a produtividade dos desenvolvedores

**João P. ALVES<sup>1</sup>; Paulo C. dos SANTOS<sup>2</sup>;**

### RESUMO

Este artigo apresenta uma revisão bibliográfica sobre o impacto da Inteligência Artificial (IA) Generativa na produtividade de desenvolvedores de software. O objetivo é analisar como assistentes de codificação, como GitHub Copilot e Gemini Code Assist, estão remodelando o ciclo de vida do desenvolvimento de software. A pesquisa sintetiza estudos empíricos que indicam ganhos de até 55% na velocidade de conclusão de tarefas, mas também aponta para desafios emergentes, como o aumento da dívida técnica e a sobrecarga na revisão de código. Conclui-se que, embora a IA Generativa ofereça um potencial transformador para a produtividade, sua adoção bem-sucedida requer uma abordagem equilibrada que considere tanto os ganhos de eficiência quanto os novos desafios de qualidade e manutenção.

**Palavras-chave:** Codificação; Eficiência; Programação.

### 1. INTRODUÇÃO

A engenharia de software atravessa uma transformação impulsionada pela ascensão da Inteligência Artificial Generativa (IA Generativa), tecnologia capaz de criar conteúdo original, incluindo código-fonte. Diferentemente das IAs analíticas, focadas em extrair insights de dados existentes, a IA Generativa introduz uma capacidade de criação que está redefinindo as práticas de desenvolvimento (SALLAM, 2023). A rápida adoção dessas ferramentas, como o GitHub Copilot, reflete seu potencial para otimizar tarefas e acelerar o ciclo de vida de desenvolvimento de software (SDLC).

Estudos iniciais demonstram um impacto quantificável e significativo. Uma pesquisa do GitHub (2022) revelou que desenvolvedores que utilizam o Copilot concluem tarefas até 55% mais rápido. Essa aceleração, no entanto, não é isenta de complexidades. Pesquisas indicam que um aumento no volume de código gerado pode levar a novos gargalos, como um maior tempo gasto na integração e revisão de pull requests, evidenciando uma tensão entre a produtividade individual e a eficiência do sistema como um todo (PENG et al., 2023).

Diante deste cenário, este artigo busca analisar, por meio de uma revisão bibliográfica, o impacto multifacetado da IA Generativa na produtividade dos desenvolvedores. O objetivo é sintetizar os benefícios mensuráveis, comparar as principais ferramentas disponíveis e discutir os desafios críticos relacionados à qualidade do código, segurança e à necessária evolução das competências profissionais, fornecendo uma visão equilibrada do estado da arte.

<sup>1</sup>Discente de Bacharelado em Ciência da Computação, IFSULDEMINAS – Campus Muzambinho. E-mail: 12201000124@muz.ifsuldeminas.edu.br.

<sup>2</sup>Orientador, IFSULDEMINAS – Campus Muzambinho. E-mail: paulo.santos@muz.ifsuldeminas.edu.br.

## 2. FUNDAMENTAÇÃO TEÓRICA

A avaliação da produtividade em engenharia de software é uma tarefa complexa, que historicamente transcende métricas simplistas como "linhas de código" (LoC). Frameworks modernos, como o SPACE, propõem uma visão multidimensional que engloba Satisfação e bem-estar, Performance, Atividade, Comunicação e colaboração, e Eficiência e fluxo (FORSGREN et al., 2021). Essa abordagem holística é fundamental para compreender o impacto da IA Generativa, que afeta não apenas a velocidade da escrita de código (Atividade), mas também a satisfação do desenvolvedor e a qualidade do resultado final (Performance).

A IA Generativa aplicada à codificação funciona por meio de Grandes Modelos de Linguagem (LLMs) treinados em vastos conjuntos de dados de código-fonte, como os do GitHub. Ferramentas como o GitHub Copilot e o Amazon CodeWhisperer utilizam esse conhecimento para prever e sugerir trechos de código, funções inteiras ou até mesmo converter comentários em linguagem natural para código executável (GITHUB, 2022). O valor dessas ferramentas reside em sua capacidade de se integrar diretamente ao Ambiente de Desenvolvimento Integrado (IDE), reduzindo a carga cognitiva e o tempo gasto em tarefas repetitivas.

Contudo, a qualidade e a segurança do código gerado são pontos de atenção. Um estudo da Universidade de Stanford revelou que o código produzido por assistentes de IA pode conter vulnerabilidades de segurança (SANDOVAL et al., 2021). Além disso, a facilidade de gerar código pode inadvertidamente aumentar a "dívida técnica" — o custo implícito de retrabalho futuro causado pela escolha de uma solução fácil agora em vez de usar uma abordagem melhor que levaria mais tempo. Relatórios da indústria já apontam para um aumento na duplicação de código em projetos que adotaram IA, o que pode comprometer a manutenibilidade a longo prazo (GITCLEAR, 2024).

## 3. MATERIAL E MÉTODOS

Este estudo é uma revisão bibliográfica da literatura para sintetizar o conhecimento atual sobre o impacto da IA Generativa na produtividade dos desenvolvedores. A busca foi realizada entre os meses de Abril e Junho de 2025 nas bases de dados IEEE Xplore, ACM Digital Library e Google Scholar, com a string **(("generative AI" OR "AI code assistant" OR "GitHub Copilot") AND ("software development" OR "software engineering") AND ("productivity" OR "efficiency" OR "impact"))**. Os critérios de inclusão foram artigos empíricos e relatórios técnicos publicados entre janeiro de 2021 e junho de 2025 garantindo a relevância dos dados em um campo de rápida evolução. Foram excluídos artigos de opinião, editoriais, trabalhos com foco distinto e publicações duplicadas.

O processo de filtragem iniciou com 50 artigos. Após a remoção de 7 duplicatas, 43 foram

triados por título e resumo, restando 15 para leitura completa. Desses, 3 foram descartados por razões como metodologia insuficiente ou foco divergente. A amostra final consistiu em 12 estudos, que foram submetidos a uma síntese qualitativa para extrair dados sobre ganhos de produtividade, funcionalidades de ferramentas e desafios relacionados à segurança e qualidade do código.

#### **4. RESULTADOS E DISCUSSÃO**

A literatura aponta para um impacto consistentemente positivo na velocidade de codificação e satisfação do desenvolvedor. Estudos da Amazon (2022) e do GitHub (2022) relatam que desenvolvedores usando CodeWhisperer e Copilot foram, respectivamente, 57% e 55% mais rápidos. Qualitativamente, os profissionais relatam maior foco e menos frustração, o que se alinha com a dimensão de "Satisfação" do framework SPACE (FORSGREN et al., 2021).

O mercado de assistentes de codificação é dominado por ferramentas com posicionamentos estratégicos distintos. O GitHub Copilot destaca-se pela integração e flexibilidade, enquanto o Gemini Code Assist (Google) e o Amazon Code Whisperer focam em seus ecossistemas de nuvem e funcionalidades empresariais, como resumido na Tabela 1. O Tabnine, por sua vez, diferencia-se pelo foco em privacidade, oferecendo implementações locais.

*Tabela 1 – Comparativo dos Principais Assistentes de Codificação por IA*

Característica	GitHub Copilot	Gemini Code Assist	Amazon CodeWhisperer
<b>Modelos de IA</b>	Multi-modelo (GPT-4, etc.)	Família Gemini	Modelo proprietário Amazon
<b>Diferencial</b>	Integração com ecossistema; Flexibilidade	Integração com Google Cloud; contexto de todo o repositório.	Integração com AWS; rastreamento de licenças.
<b>Foco Principal</b>	Desenvolvedor individual e times	Empresas e ecossistema Google.	Empresas e ecossistema AWS.
<b>Segurança</b>	Filtros de conteúdo; Gestão de políticas	Segurança de nível empresarial; código privado.	Análise de vulnerabilidades; indenização de PI.

*Fonte: Elaborado pelo autor (2025), com base em Sallam (2023) e GitHub (2022).*

Apesar dos ganhos de velocidade, a produtividade real é mais complexa. O estudo de Peng et al. (2023) descobriu que o aumento no volume de código sobrecarrega as etapas colaborativas, com o tempo para integrar pull requests subindo 41.6%. Isso sugere que a aceleração individual não se traduz em maior velocidade de entrega, mas cria gargalos na fase de revisão. Essa geração massiva de código também eleva o risco de dívida técnica, com estudos mostrando um aumento na duplicação de código, o que onera a manutenção futura (GITCLEAR, 2024).

## 5. CONCLUSÃO

A integração da Inteligência Artificial Generativa na engenharia de software representa um avanço transformador, com assistentes de codificação demonstrando capacidade de aumentar significativamente a velocidade das tarefas de desenvolvimento e a satisfação dos profissionais. No entanto, este estudo conclui que os ganhos de produtividade não devem ser medidos apenas pela aceleração da escrita de código individual. A análise da literatura evidencia que os benefícios vêm acompanhados de desafios críticos, como o aumento da carga sobre os processos de revisão, o risco de introduzir vulnerabilidades de segurança e a potencial acumulação de dívida técnica, que podem comprometer a qualidade e a sustentabilidade dos projetos a longo prazo.

A escolha da ferramenta ideal depende do alinhamento com o ecossistema tecnológico e o perfil de risco de cada organização. A limitação deste estudo reside na natureza emergente do campo, com poucos estudos longitudinais sobre os efeitos a longo prazo. Recomenda-se, para trabalhos futuros, a realização de pesquisas que acompanhem projetos ao longo de vários meses para medir a evolução da dívida técnica e investigações sobre novas metodologias de revisão de código e testes adaptadas a um fluxo de trabalho auxiliado por IA.

## REFERÊNCIAS

AMAZON. AWS study shows developers are 27% more likely to complete tasks successfully and 57% faster on average with Amazon CodeWhisperer. 2022. Disponível em: <https://sonix.ai/resources/pt/fonte-com-captacoes-fechadas/>. Acesso em: 20 jun. 2025.

FORSGREN, N.; SUTTER, D.; STOREY, M-A.; MADISON, J.; ZIMMERMANN, T. The SPACE of Developer Productivity: There's more to it than you think. **Queue**, v. 19, n. 1, p. 20-53, 2021.

GITCLEAR. The 2024 State of Code Quality & AI. 2024. Disponível em: <https://sonix.ai/resources/pt/fonte-com-captacoes-fechadas/>. Acesso em: 20 jun. 2025.

GITHUB. Research: quantifying GitHub Copilot's impact on developer productivity and happiness. 2022. Disponível em: <https://github.blog/2022-09-07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/>. Acesso em: 20 jun. 2025.

PENG, S.; ZHAO, Y.; WANG, C.; YIN, P.; LIN, D. The Impact of AI on Developer Productivity: Evidence from GitHub Copilot. **arXiv preprint arXiv:2302.06590**, 2023.

SALLAM, R. The CIO's Guide to Generative AI. Gartner, 2023. Disponível em: <https://sonix.ai/resources/pt/fonte-com-captacoes-fechadas/>. Acesso em: 20 jun. 2025.

SANDOVAL, G.; NGUYEN, H.; NADI, S.; SITARAMAN, H.; CHEUNG, S.; Solar-Lezama, A. Security Implications of Code Generated by Artificial Intelligence. In: 2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP). Anais...2021, p. 231-240.