

# THE INFLUENCE OF INTELLIGENT AGENTS ON THE SPEED AND QUALITY OF SOFTWARE DEVELOPMENT

Vinnitsia National Technical University

## Анотація

*Інтеграція інтелектуальних ШІ-агентів у сучасні середовища розробки являє собою зміну парадигми в інженерії програмного забезпечення. Замість традиційного автодоповнення окремих рядків, новітні інструменти, такі як редактор Cursor, здатні аналізувати глобальний контекст усього проєкту, автономно генерувати комплексні правки та виявляти логічні помилки в реальному часі. У цій роботі розглядається вплив автономних помічників на робочий процес програміста, висвітлюється оптимізація швидкості написання коду та підвищення його загальної якості під час розробки та рефакторингу.*

**Ключові слова:** інтелектуальні агенти, розробка програмного забезпечення, штучний інтелект, великі мовні моделі, Cursor, рефакторинг.

## Abstract

*The integration of intelligent AI agents into modern development environments is a paradigm shift in software engineering. Rather than the usual single-line autocomplete, there are more sophisticated tools available, such as the Cursor Editor, which can analyse the global context of an entire project, generate complex modifications autonomously and detect logical errors in real time. The focus of this paper is on the impact of autonomous assistants on the workflow of programmers. It looks at how coding speed can be optimised and how the quality of code can be improved during development and refactoring.*

**Keywords:** intelligent agents, software development, artificial intelligence, large language models and Cursor refactoring.

## Introduction

In recent years, the field of software engineering has experienced a significant transformation, driven by the rapid advancement of artificial intelligence (AI). Traditionally, developers relied on basic code completion tools that predicted the next word or line based on limited local context. While these tools did improve typing speed, they lacked an understanding of broader project architecture and complex logic.

To address these limitations, a new generation of intelligent AI agents has emerged that are integrated directly into modern development environments like the Cursor editor. Unlike standard autocomplete, these agents use Large Language Models (LLMs) to analyse the global context of entire software projects. This enables them to autonomously generate complex, multi-file modifications, assist with large-scale refactoring and detect logical errors in real time. This paper examines the role of such autonomous assistants in modern programming workflows, detailing how they optimise development speed and enhance the quality of software.

## Research results

The main benefit of using intelligent agents in software development is their ability to maintain comprehensive 'project-wide context'. Unlike standard AI interfaces, tools such as the Cursor editor utilise advanced indexing techniques, such as vector embeddings, to map the entire repository. [1] This allows the agent to understand the dependencies between different modules and files, ensuring that any generated code aligns with the existing architecture and naming conventions.

Significant improvements in development speed are observed through 'agentic workflows', where the AI can perform multi-file edits simultaneously. For example, when a developer requests a change to a data structure, the agent can autonomously update all related service calls and test cases across the codebase. [2] Research suggests that such automation can reduce the time spent on boilerplate code and repetitive refactoring tasks by 40-60%.

Furthermore, software quality is enhanced through real-time error detection and proactive debugging. Intelligent agents can analyse code as it is written, identifying potential logical flaws, security vulnerabilities or performance bottlenecks before the compilation stage. [3] The integration of 'Composer' modes enables iterative prompting, whereby the developer and the agent collaborate to refine complex algorithms, resulting in more robust and maintainable code. [4] Ultimately, the transition from simple autocomplete to autonomous agents shifts the programmer's role from manual coding to high-level architectural oversight.

## Conclusion

In conclusion, integrating intelligent AI agents into the software development process significantly improves engineering efficiency and code reliability. Leveraging tools such as the Cursor editor, which offer comprehensive codebase indexing and autonomous agentic workflows, enables developers to overcome the limitations of traditional autocomplete. These technologies accelerate the implementation of complex features and large-scale refactoring, while also enhancing the overall quality of software by minimising logical errors through deep context awareness. As AI capabilities evolve, software engineers' roles will increasingly shift from manual coding to high-level system orchestration, making intelligent agents indispensable components of the modern development ecosystem.

## REFERENCES

1. Cursor – The AI Code Editor. URL: <https://docs.cursor.com/context/codebase-indexing> (accessed 17/03/2026).
2. E. Kalliamvakou, 'Research: Quantifying GitHub Copilot's Impact on Developer Productivity and Happiness'. URL: <https://github.blog/2022-09-07-research-quantifying-github-copilots-impact-on-developer-productivity-and-happiness/> (accessed 17/03/2026).
3. AI-Powered Coding: The impact on code quality and security. URL: <https://www.sonarsource.com/blog/ai-code-quality-security/> (date of access: 17/03/2026).
4. Ng, A. 'Why Agentic Workflows are the Next Big Step for AI'. URL: <https://www.deeplearning.ai/the-batch/agentic-workflows-next-big-step-for-ai/> (date of access: 17/03/2026).

**Кудінов Андрій Васильович** – студент групи 5ПІ-25б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: [kudinov446@gmail.com](mailto:kudinov446@gmail.com).

Науковий керівник: **Чопляк Вікторія Володимирівна** – викладач англійської мови, кафедра іноземних мов, Вінницький національний технічний університет, м. Вінниця, e-mail: [nikavnuchkova@gmail.com](mailto:nikavnuchkova@gmail.com).

**Andrii V. Kudinov** – student in Group 5PI-25b, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: [kudinov446@gmail.com](mailto:kudinov446@gmail.com).

Scientific Supervisor: **Victoria V. Chopliak** – teacher of English, Foreign Languages Department, Vinnytsia National Technical University, Vinnytsia, e-mail: [nikavnuchkova@gmail.com](mailto:nikavnuchkova@gmail.com).