

УДК 004.41:004.8

Бармак Р. – ст. гр. СП-41, Дегодюк І. – ст. гр. СП-31

Тернопільський національний технічний університет імені Івана Пулюя

МЕТОДОЛОГІЯ СПЕЦИФІКАЦІЙНО-ОРІЄНТОВАНОЇ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЗА ДОПОМОГОЮ ІНСТРУМЕНТІВ ШТУЧНОГО ІНТЕЛЕКТУ

Науковий керівник: PhD Бревус В. М.

Barmak R., Dehodiuk I.

Ternopil Ivan Puluj National Technical University

METHODOLOGY OF SPECIFICATION-ORIENTED SOFTWARE DEVELOPMENT USING ARTIFICIAL INTELLIGENCE TOOLS

Supervisor: PhD Brevus V. M.

Ключові слова: специфікаційно-орієнтована розробка, програмування з використанням штучного інтелекту, автоматизація розробки програмного забезпечення.

Keywords: Spec-Driven Development, AI-assisted Programming, Software Development Automation

Вступ. Стрімкий розвиток великих мовних моделей відкриває нові можливості автоматизації процесу розробки програмного забезпечення. Проте, згідно з даними галузевих звітів, безпосередня генерація коду інструментами штучного інтелекту може призводити до проблем: порушення архітектурних принципів, відхилення від встановлених конвенцій кодової бази та генерація нерелевантного або некоректного коду[1].

Мета роботи – дослідження і формалізація методології специфікаційно-орієнтованої розробки, що забезпечує структуроване управління контекстом для ефективної розробки програмного забезпечення з використанням інструментів штучного інтелекту.

Основна частина. Методологія специфікаційно-орієнтованої розробки базується на принципі «пріоритет специфікації». Ця концепція реалізує необхідний в епоху ШІ-асистентів перехід від парадигми написання коду до розробки на основі контрактів[2]. Методологія включає три ключові компоненти:

1. Конституція проєкту – нормативний документ, що визначає архітектурні принципи, конвенції кодування та обов'язкові перевірки якості. Як свідчать результати досліджень, такий підхід дозволяє впроваджувати принцип «безпеки за проєктуванням» безпосередньо в процесі автоматизованої генерації коду[3]. Конституція інтегрується у контекст роботи асистента штучного інтелекту за допомогою спеціальних файлів інструкцій, які автоматично завантажуються самими інструментами розробки.

2. Ієрархія специфікацій – структурована система документів:

- План – технічний контекст, архітектурні рішення, фази виконання;
- Специфікація – користувацькі сценарії з виставленими пріоритетами для фокусу, функціональні вимоги, крайні випадки;
- Завдання – декомпозовані атомарні завдання з чіткими критеріями прийняття.

3. Зворотний зв'язок та ітерації – механізми перевірки згенерованого коду через автоматичні перевірки типів у кодї та якості написаного коду.

Ключовою перевагою методології є інтеграція контексту безпосередньо в репозиторій проекту, що дозволяє асистенту штучного інтелекту автоматично враховувати архітектурні обмеження та конвенції при генерації коду.

На відміну від класичної методології Agile, де документація часто мінімізується на користь робочого коду, специфікаційно-орієнтована розробка встановлює специфікацію як основний артефакт. Такий підхід узгоджується з принципами формальних методів розробки, водночас залишаючись практичним і гнучким до змін вимог. Порівняно з традиційним підходом розробки через тестування, специфікаційно-орієнтований підхід розширює концепцію контракту за межі модульних тестів, охоплюючи архітектурні рішення, користувацький досвід та бізнес-логіку.

Перехід до специфікаційно-орієнтованої розробки вимагає трансформації процесів команди розробки. Критичним фактором успіху є створення культури документування перед написанням коду, що традиційно може зустрічати опір від розробників, звиклих до безпосередньої імплементації. Важливим елементом є інтеграція специфікацій у систему контролю версій поряд з кодом, що забезпечує синхронізацію документації та реалізації. Додатковою перевагою такого підходу є можливість проводити перевірку коду на рівні специфікацій до початку розробки, що дозволяє виявляти архітектурні проблеми та непорозуміння на ранніх етапах.

Результати. Застосування методології специфікаційно-орієнтованої розробки продемонструвало:

- зменшення кількості виправлень коду під час під час перевірки коду, згенерованого асистентом штучного інтелекту;
- збереження і дотримання архітектурної цілісності проекту під час генерації коду штучним інтелектом;
- можливість паралельної роботи над різними функціональними модулями без конфліктів;
- підвищення передбачуваності результатів генерації коду штучним інтелектом;

Висновки. Методологія специфікаційно-орієнтованої розробки ефективно вирішує проблему контекстного управління в розробці, забезпечуючи структурований підхід до формулювання завдань та перевірки результатів.

- [1] D. DeBellis, K. Storer, N. Harvey, M. Beane, R. Edwards, E. Fraser et al, "DORA 2025 State of AI-Assisted Software Development Report," DORA, Google, 2025. [Online]. Available: <https://dora.dev/dora-report-2025> [Accessed: Mar. 30, 2026].
- [2] B. P. Deepak, "Spec-Driven Development: From Code to Contract in the Age of AI Coding Assistants," 2026. [Online]. Available: <https://arxiv.org/abs/2602.00180> (accessed Mar. 30, 2026).
- [3] R. M. Srinivas, "Constitutional Spec-Driven Development: Enforcing Security by Construction in AI-Assisted Code Generation," 2026. [Online]. Available: <https://arxiv.org/abs/2602.02584> (accessed Mar. 30, 2026).