УДК 004.9 DOI: 10.20998/2413-3000.2024.9.6

А. М. КОПП, І. С. НЕСТЕРЕНКО

МОДЕЛЬ ВИБОРУ ІНСТРУМЕНТІВ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ПІДТРИМКИ ПРОЦЕСІВ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Інтеграція інструментів штучного інтелекту (ШІ) у проєкти з розробки програмного забезпечення значно підвищує ефективність виконання різних завдань протягом життєвого циклу розробки програмного забезпечення (SDLC). Інструменти на основі ШІ, вбудовані в інтегровані середовища розробки (IDE), підвищують продуктивність розробників і якість коду, а також сприяють кращій взаємодії між учасниками проєкту та системами на основі ШІ. Основні напрямки досліджень у сфері інтеграції ШІ в процеси розробки програмного забезпечення включають адаптацію користувацьких інтерфейсів під конкретні завдання, підвищення довіри до систем на основі ШІ та покращення читабельності коду. ШІ покращує декілька етапів SDLC, зокрема автоматизовану генерацію коду, перегляд коду та прогнозування дефектів. Впровадження інструментів ШІ в IDE прискорює розробку, підвищує якість коду та зменшує кількість дефектів. Машинне навчання та обробка природної мови відіграють важливу роль у підвищенні якості програмного забезпечення завдяки класифікації вимог та прогнозуванню дефектів. Рішення на основі ШІ, такі як рекомендаційні системи та чат-боти, підтримують різні процеси розробки програмного забезпечення, включаючи збір вимог. Тому актуальним науково-практичним завданням є створення моделі для обґрунтованого вибору інструментів ШІ для підтримки процесів розробки програмного забезпечення з метою підвищення ефективності проєктів. У цьому дослідженні запропоновано математичну модель, яка мінімізує витрати на використання інструментів ШІ, забезпечуючи при цьому дотримання мінімальних вимог, що впливають на ефективність проєкту. Оптимізаційна модель враховує такі критерії, як вартість, можливості інтеграції, підтримки та функціональності, використовуючи нормалізовані оцінки на основі Gartner Peer Insights та інших відкритих джерел. Цільова функція мінімізує загальну вартість інструментів ШІ за умови дотримання обмежень, що забезпечують мінімально прийнятні оцінки. Розроблений підхід уможливлює системний вибір інструментів ШІ, що підвищує ефективність проєктів з розробки програмного забезпечення.

Ключові слова: інструменти штучного інтелекту, процеси розробки програмного забезпечення, проєкт з розробки програмного забезпечення, оптимізаційна модель.

A. M. KOPP, I. S. NESTERENKO

A MODEL FOR SELECTING ARTIFICIAL INTELLIGENCE TOOLS TO SUPPORT SOFTWARE DEVELOPMENT PROCESSES

Integrating artificial intelligence (AI) tools into software development projects significantly improves the efficiency of various tasks within the software development lifecycle (SDLC). AI-driven tools embedded in integrated development environments (IDEs) improve developer productivity and code quality, and facilitate better interaction between project participants and AI-based systems. The main research directions for integrating AI into software development processes include adapting user interfaces for specific tasks, increasing trust in AI-based systems, and improving code readability. AI enhances several SDLC stages, including automated code generation, code review and defect prediction. Implementing AI tools in IDEs accelerates development, improves code quality and reduces defects. Machine learning and natural language processing play a critical role in improving software quality through requirements classification and defect prediction. AI-based solutions, such as recommendation systems and chatbots, support various software development processes, including requirements gathering. Therefore, a relevant scientific and practical challenge is to create a model for the justified selection of AI tools to support software development processes in order to improve project efficiency. This study proposes a mathematical model that minimizes the cost of using AI tools, while ensuring compliance with minimum requirements that affect project efficiency. The optimization model takes into account criteria such as pricing, integration, support and functionality capabilities, using normalized evaluations based on Gartner Peer Insights and other open sources. The objective function minimizes the total cost of AI tools, subject to constraints that ensure minimum acceptable evaluation scores. The developed approach enables a systematic selection of AI tools, thus improving the efficiency of software development projects.

Keywords: artificial intelligence tools, software development processes, software development project, optimization model.

Вступ. Інтеграція різних інструментів штучного інтелекту (ШІ) в проєкти з розробки програмного забезпечення (ПЗ) значно підвищує ефективність виконання різних завдань у рамках життєвого циклу розробки програмного забезпечення (SDLC – Software Development Lifecycle).

Так, наприклад, інструменти ШІ, інтегровані в середовища розробки ПЗ (IDE – Integrated Development Environment) дозволяють підвищити продуктивність розробників і якість коду, сприяючи кращій взаємодії залучених до проєкту фахівців з інструментами на основі ШІ. Основні напрямки досліджень стосовно інтеграції ШІ в процеси розробки ПЗ включають розробку користувацьких інтерфейсів, пристосованих до виконання конкретних завдань, зміцнення довіри до ШІ-орієнтованих систем та покращення читабельності програмного коду [1].

Застосування ШІ дозволяє удосконалити різні етапи SDLC, включаючи автоматизовану генерацію програмного коду, перегляд програмного коду (так зване «code review») та передбачення виникнення помилок. Впровадження до IDE інструментів на основі ШІ призводить до прискорення процесів розробки, покращення якості коду та запобігання виникненню помилок [2].

Машинне навчання (ML – Machine Learning) та обробка природної мови (NLP – Natural Language Processing) відіграють ключову роль у підвищенні якості ПЗ, наприклад, завдяки класифікації вимог або прогнозування виникнення дефектів. Таким чином, широке застосування відповідних інструментів ML та NLP на всіх етапах розробки ПЗ дозволяє підвищити ефективність виконання технічних завдань членами проєктної команди [3].

Різноманітні рішення на основі ШІ, такі як рекомендаційні системи та чат-боти, забезпечують

© А. М. Копп, І. С. Нестеренко, 2024

підтримку різних процесів розробки ПЗ, таких як збір вимог і багатьох інших. Ці інструменти дозволяють підвищити ефективність праці як розробників ПЗ, так і інших фахівців, спрощуючи завдання завдяки наданню інтелектуальної підтримки своїм користувачам [4].

Отже, актуальною науково-практичною задачею є створення моделі для забезпечення обґрунтованого вибору ШІ-засобів підтримки процесів розробки ПЗ, що дозволить підвищити ефективність проєктів з розробки ПЗ.

Аналіз останніх досліджень і публікацій. Згідно дослідження [5], використання інструментів на основі ШІ, таких як GitHub Copilot, демонструє підвищення продуктивності на 21-28% при виконанні завдань в проєктах з розробки ПЗ. Автори відмічають, що це стає особливо корисним для розробників-початківців, завдяки скороченню часу на написання програмного коду та підвищенню задоволеності роботою [5].

Систематичний огляд досліджень [2] в сфері інтеграції ШІ в інженерію ПЗ підкреслює використання машинного навчання та глибокого навчання на етапах розробки та тестування ПЗ, що дозволяє підвищити якість ПЗ завдяки прогнозуванню виникнення помилок та рекомендаціям щодо покращення якості програмного коду [2].

У літературних джерелах, присвячених досвіду у сфері інтеграції ШІ до IDE, наприклад у [1], підкреслюється важливість розробки користувацьких інтерфейсів, які сприяють довірі розробників та покращують читабельність програмного коду, що має вирішальне значення для ефективної співпраці між розробниками та ШІ-засобами [1]. В іншій оглядовій статті [6] здійснюється аналіз 44 досліджень, які висвітлюють роль застосування інструментів на основі ШІ в підвищенні ефективності виконання фаз Agile-проєктів, пов’язаних з оцінками ризиків у плануванні, оцінками зусиль, розподілом завдань, комунікацією в команді, розв’язанням технічних задач та іншими аспектами.

У роботі [7] увага приділяється дослідженню впливу генеративного ШІ на ефективність розробки ПЗ, результати застосування якого демонструють скорочення часу на створення програмного коду на 45%, покращення тестового покриття програмного коду на 60% і зменшення кількості проблем після розгортання ПЗ на 30%, що підкреслює важливість застосування інструментів на основі ШІ для підвищення ефективності виконання завдань у проєктах з розробки ПЗ. Інше дослідження [8] розглядає застосування методів машинного навчання для розподілу ресурсів і планування, що демонструє значне підвищення ефективності проєктів з розробки ПЗ, в тому числі – підвищення ефективності розподілу ресурсів на 37% і збільшення частки вчасно виконаних проєктів на 25%.

У статті [9] також відзначається роль ШІ в підвищенні ефективності управління проєктами, зокрема за допомогою таких методів, як лінійне програмування, генетичні алгоритми та нейронні мережі, які дозволяють удосконалити процеси розподілу ресурсів, планування та прийняття рішень, що в кінцевому підсумку дозволить покращити результати проєктів з розробки ПЗ та задоволеність стейкхолдерів.

Стаття [10], присвячена ролі ШІ в удосконаленні Agile-методологій, розглядає застосування ШІ-засобів для підвищення якості ПЗ. В роботі [10] відмічається важливість вибору відповідних ШІ-асистентів для виконання конкретних завдань з метою удосконалення процесів розробки ПЗ. У роботі [11] розглядаються методи ШІ, такі як нейронні мережі, нечітка логіка та інтелектуальний аналіз даних, відмічається потенціал їх застосування для підвищення ефективності розробки ПЗ на різних етапах проєкту, зокрема, підкреслюється важливість використання інструментів ШІ, спеціально адаптованих до конкретних завдань.

Дослідження [12] присвячене інтелектуальним інструментам розробки ПЗ, які дозволяють підвищити ефективність за рахунок автоматизації генерації програмного коду на основі високорівневих описів бізнес-процесів, дозволяючи інженерам і замовникам зосередитися на вимогах, а не на програмуванні. Також у статті [13] розглядається інтеграція генеративного ШІ в гнучку розробку ПЗ – висвітлюється застосування ШІ в завданнях генерації програмного коду, автоматизованому тестуванні та пошуку дефектів, що дозволяє підвищити продуктивність і прискорити цикли доставки ПЗ, ефективно вирішуючи загальні проблеми управління проєктами.

Ще в одній статті [14] розглядається застосування методів ШІ для підвищення ефективності розробки ПЗ протягом усього життєвого циклу розробки ПЗ. При цьому, акцент робиться на автоматизацію таких завдань, як інженерія вимог, генерація програмного коду, проєктування ПЗ, тестування ПЗ, покращуючи таким чином результати проєкту та скорочуючи час на його реалізацію [14].

Мета дослідження. Підвищення ефективності проєктів з розробки ПЗ за рахунок обґрунтованого вибору ШІ-засобів підтримки процесів розробки ПЗ і подальшого їх впровадження.

Виклад основного матеріалу. Для забезпечення обґрунтованого вибору ШІ-засобів підтримки процесів розробки ПЗ, пропонується математична модель, яка передбачає мінімізацію вартості використання ШІ-інструментів при забезпеченні мінімальних вимог за аспектами, які впливають на ефективність проєктів з розробки ПЗ.

Нехай визначено множину завдань проєкту з розробки ПЗ $T = \{t_1, t_2, \dots, t_n\}$ та множину доступних ШІ-інструментів $A = \{a_1, a_2, \dots, a_m\}$.

Визначимо бінарну змінну x_{ij}, $i = \overline{1,n}$, $j = \overline{1,m}$:

$$x_{ij} = \begin{cases} 1, \text{якщо } a_j \text{ обрано для завдання } t_i; \\ 0, \text{в іншому випадку.} \end{cases} \quad (1)$$

Для кожного інструменту визначаються наступні оцінки, згідно методології Gartner Peer Insights [15]:

- $e_j, j = \overline{1,m}$ – оцінка інструмента a_j за ціновою політикою та взаємодією з клієнтами (Evaluation & Contracting);
- $d_j, j = \overline{1,m}$ – оцінка інструмента a_j за рівнем інтеграції та розгортання (Integration & Deployment);
- $s_j, j = \overline{1,m}$ – оцінка інструмента a_j за рівнем обслуговування та підтримки (Service & Support);
- $p_j, j = \overline{1,m}$ – оцінка інструмента a_j за функціональними можливостями (Product Capabilities);
- $c_j, j = \overline{1,m}$ – вартість використання інструмента a_j.

Оскільки за методологією Gartner Peer Insights оцінювання інструментів здійснюється за шкалою 1–5, пропонується провести нормалізацію за допомогою Min-Max методу [16] для приведення значень e_j, d_j, s_j, $p_j, j = \overline{1,m}$ до діапазону [0, 1]:

$$e_j^* = \frac{e_j - \min\limits_{j=\overline{1,m}} e_j}{\max\limits_{j=\overline{1,m}} e_j - \min\limits_{j=\overline{1,m}} e_j}, \quad (2)$$

$$d_j^* = \frac{d_j - \min\limits_{j=\overline{1,m}} d_j}{\max\limits_{j=\overline{1,m}} d_j - \min\limits_{j=\overline{1,m}} d_j}, \quad (3)$$

$$s_j^* = \frac{s_j - \min\limits_{j=\overline{1,m}} s_j}{\max\limits_{j=\overline{1,m}} s_j - \min\limits_{j=\overline{1,m}} s_j}, \quad (4)$$

$$p_j^* = \frac{p_j - \min\limits_{j=\overline{1,m}} p_j}{\max\limits_{j=\overline{1,m}} p_j - \min\limits_{j=\overline{1,m}} p_j}. \quad (5)$$

Для нормалізації значень вартості $c_j, j = \overline{1,m}$ пропонується застосувати обернену нормалізацію, оскільки менші значення вартості в умовах вибору ШІ-інструментів для виконання завдань проєкту з розробки ПЗ є переважними:

$$c_j^* = \frac{\max\limits_{j=\overline{1,m}} c_j - c_j}{\max\limits_{j=\overline{1,m}} c_j - \min\limits_{j=\overline{1,m}} c_j}. \quad (6)$$

Таким чином, після нормалізації всі оцінки ШІ-інструментів $a_j, j = \overline{1,m}$ за означеними критеріями (2)–(6) будуть приведені до однакового масштабу в діапазоні [0, 1], що дозволить використовувати їх у оптимізаційній моделі вибору ШІ-засобів підтримки процесів розробки ПЗ.

Цільова функція запропонованої оптимізаційної моделі передбачає мінімізацію загальної вартості C використання ШІ-засобів для виконання завдань проєкту з розробки ПЗ, що визначається через суму добутків нормалізованих коефіцієнтів вартості c_j^* та бінарних змінних x_{ij}:

$$\min C, C = \sum_{i=1}^{n} \sum_{j=1}^{m} c_j^* x_{ij}. \quad (7)$$

У вигляді обмежень оптимізаційної задачі представимо мінімальні вимоги до оцінок ШІ-інструментів згідно методології Gartner Peer Insights, виражені через нормалізовані коефіцієнти e_j^*, d_j^*, s_j^*, p_j^*, $j = \overline{1,m}$, які повинні бути виконані за кожним аспектом:

$$E \geq \hat{E}, E = \sum_{i=1}^{n} \sum_{j=1}^{m} e_j^* x_{ij}, \quad (8)$$

де E – загальна оцінка набору ШІ-засобів за ціновою політикою та взаємодією з клієнтами (Evaluation & Contracting);

$\hat{E}$ – мінімальне прийнятне значення оцінки набору ШІ-засобів за ціновою політикою та взаємодією з клієнтами (Evaluation & Contracting);

$$D \geq \hat{D}, D = \sum_{i=1}^{n} \sum_{j=1}^{m} d_j^* x_{ij}, \quad (9)$$

де D – загальна оцінка набору ШІ-засобів за рівнем інтеграції та розгортання (Integration & Deployment);

$\hat{D}$ – мінімальне прийнятне значення оцінки набору ШІ-засобів за рівнем інтеграції та розгортання (Integration & Deployment);

$$S \geq \hat{S}, S = \sum_{i=1}^{n} \sum_{j=1}^{m} s_j^* x_{ij}, \quad (10)$$

де S – загальна оцінка набору ШІ-засобів рівнем обслуговування та підтримки (Service & Support);

$\hat{S}$ – мінімальне прийнятне значення оцінки набору ШІ-засобів рівнем обслуговування та підтримки (Service & Support);

$$P \geq \hat{P}, P = \sum_{i=1}^{n} \sum_{j=1}^{m} p_j^* x_{ij}, \quad (11)$$

де P – загальна оцінка набору ШІ-засобів за функціональними можливостями (Product Capabilities);

$\hat{P}$ – мінімальне прийнятне значення оцінки набору ШІ-засобів за функціональними можливостями (Product Capabilities).

Додатково накладається обмеження з вибору для кожного завдання проєкту з розробки ПЗ лише одного ШІ-інструменту:

$$\sum_{j=1}^{m} x_{ij} = 1, i = \overline{1,n}. \quad (12)$$

Крім того, змінні оптимізації x_{ij}, $i = \overline{1,n}$, $j = \overline{1,m}$ повинні бути булевими:

$$x_{ij} \in \{0,1\}, i = \overline{1,n}, j = \overline{1,m}. \quad (13)$$

Запропонована математична модель являє собою задачу булевої оптимізації, яка дозволяє сформувати набір ШІ-інструментів для виконання набору завдань проєкту з розробки ПЗ з урахуванням аспектів якості (згідно методології Gartner Peer Insights) та вартості. Основною метою даної моделі є мінімізація загальних витрат на використання ШІ-засобів при одночасному забезпеченні виконання мінімальних оцінок цінової політики та взаємодії з клієнтами, рівня інтеграції та розгортання, обслуговування та підтримки, а також оцінок функціональних можливостей. Нормалізація відповідних коефіцієнтів надає змогу уникнути впливу різних масштабів величин (наприклад, оцінок від 1 до 5 та вартості використання ШІ-засобів у

грошових одиницях) при порівнянні різних інструментів.

Загальна схема розв'язання запропонованої задачі вибору ШІ-засобів підтримки процесів розробки ПЗ наведена на рис. 1.

Представлена процедура вибору ШІ-інструментів для проєктів з розробки ПЗ на основі принципів оптимізації (рис. 1) розпочинається з формування множини завдань проєкту $T = \{t_1, t_2, \dots, t_n\}$, що представляє різні види діяльності з розробки ПЗ, які потребують підтримки ШІ.

Згодом визначаються доступні інструменти ШІ, що формують множину $A = \{a_1, a_2, \dots, a_m\}$, яка включає різні рішення, що можуть допомогти на різних етапах життєвого циклу розробки ПЗ.

Для оцінки доступних інструментів на основі ШІ складаються оцінки на основі інформації з таких джерел, як Gartner Peer Insights. Ці значення оцінюють ШІ-інструменти за багатьма аспектами, зокрема за ціновою політикою та взаємодією з клієнтами E, рівнем інтеграції та розгортання D, рівнем обслуговування та підтримки S, а також функціональними можливостями P.

Крім того, витрати C, пов'язані з цими інструментами, визначаються з використанням відкритих джерел, що забезпечує комплексну оцінку як якісних, так і кількісних аспектів застосування ШІ-засобів у підтримці процесів розробки ПЗ.

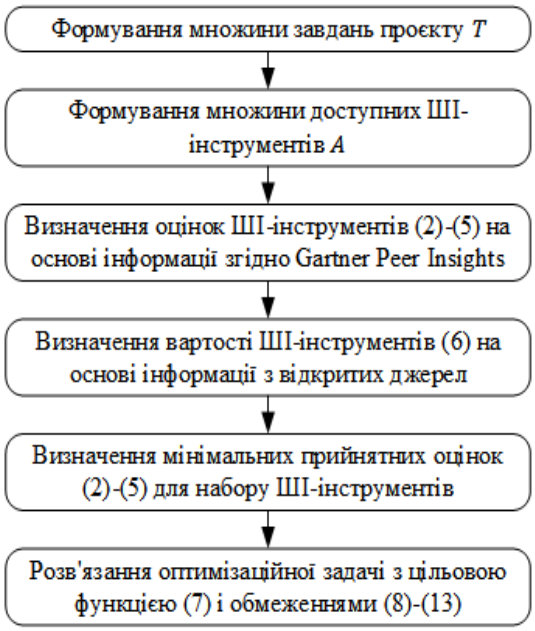

Рис. 1. Загальна схема розв'язання задачі

Крім того, наведена процедура (рис. 1) передбачає визначення мінімально прийнятних оцінок для інструментів ШІ за ціновою політикою та взаємодією з клієнтами $\hat{E}$, рівнем інтеграції та розгортання $\hat{D}$, рівнем обслуговування та підтримки $\hat{S}$, а також функціональними можливостями $\hat{P}$.

Введення обмежень дозволить забезпечити вибір лише тих інструментів, які відповідають визначеним пороговим значенням для кожного з аспектів якості ШІ-засобу.

Нарешті, формулюється оптимізаційна задача, цільова функція (7) якої мінімізує загальну вартість обраних інструментів ШІ, задовольняючи при цьому обмеження (8)–(13), пов'язані з мінімально допустимими оцінками ШІ-засобів, обмеженням з вибору для кожного завдання проєкту з розробки ПЗ лише одного ШІ-інструменту, а також обмеженням на булеві змінні оптимізації.

Отриманий розв'язок представленої задачі оптимізації забезпечить обґрунтований вибір інструментів ШІ, які дозволять підвищити ефективність проєкту з розробки ПЗ.

Висновки. В роботі було розглянуто задачу обґрунтованого вибору ШІ-засобів для підтримки процесів розробки ПЗ на основі моделі оптимізації. Запропонована процедура передбачає ідентифікацію множини завдань проєкту, формування множини доступних ШІ-інструментів, визначення їх оцінок за аспектами якості та вартістю, а також розв'язання задачі вибору ШІ-інструментів відповідно до заданих обмежень.

Основним результатом дослідження є формальна моделі вибору ШІ-інструментів, яка враховує якісні характеристики та вартість ШІ-засобів на основі інформації з відкритих джерел. Формалізація задачі вибору ШІ-інструментів для підтримки процесів розробки ПЗ у вигляді оптимізаційної моделі дозволить визначати рішення, що мінімізуватимуть загальні витрати, забезпечуючи при цьому відповідність вимогам до якості таких інструментів.

У подальшому, отримані результати дослідження зможуть бути використані для прийняття обґрунтованих рішень у процесі управління розробкою ПЗ, що сприятиме підвищенню ефективності проєктів з розробки ПЗ та зменшенню витрат на впровадження відповідних ШІ-рішень.

Список літератури

1. Sergeyuk A., Titov S., Izadi M. In-IDE Human-AI Experience in the Era of Large Language Models; A Literature Review. *Arxiv.org*, 2024. Available at : https://www.doi.org/10.48550/arxiv.2401.10739. (accessed 10.12.2024).
2. Goel P. K., Komal K., Vashishth N. AI-Driven Software Development Lifecycle Optimization. *Advancing Software Engineering Through AI, Federated Learning, and Large Language Models*, 2024. Available at : https://www.doi.org/10.4018/979-8-3693-3502-4.ch005. (accessed 10.12.2024).
3. Durrani U. K., Akpinar M., Adak M. F., Kabakus A. T., Öztürk M. M., Saleh M. A Decade of Progress: A Systematic Literature Review on the Integration of AI in Software Engineering Phases and Activities (2013-2023). *IEEE Access*, 2024. Available at : https://www.doi.org/10.1109/access.2024.3488904. (accessed 11.12.2024).
4. Del Carpio A. F., Angarita L. B. Assistant Solutions in Software Engineering: A Systematic Literature Review. *IEEE Xplore*, 2023. Available at : https://www.doi.org/10.1109/icsess58500.2023.10293029. (accessed 12.12.2024).
5. Ng K. K., Fauzi L., Leow L., Ng J. Harnessing the Potential of Gen-AI Coding Assistants in Public Sector Software Development. *Arxiv.org*, 2024. Available at : https://www.doi.org/10.48550/arxiv.2409.17434. (accessed 12.12.2024).

6. Lumbanraja H. L., Raharjo T., Fitriani A. N. Artificial Intelligence Implementation in Agile Project Management Addressing Challenges and Maximizing Impact. *The Indonesian Journal of Computer Science*, Vol. 3, No. 4, 2024. Available at : https://www.doi.org/10.33022/ijcs.v13i4.4155. (accessed 12.12.2024).
7. Modi M. D. B. Transforming Software Development Through Generative AI : A Systematic Analysis of Automated Development Practices. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, Vol. 10, No. 6. Available at : https://www.doi.org/10.32628/cseit24106197. (accessed 14.12.2024).
8. Hossain M. Z., Hasan L., Dewan M. A., Monira N. A. The impact of artificial intelligence on project management efficiency. *International Journal of Management Information Systems and Data Science*, Vol. 1, No. 05, 2024. Available at : https://www.doi.org/10.62304/ijmisds.v1i05.211. (accessed 15.12.2024).
9. Sravanthi J., Sobti R., Semwal A., Shravan M., Al-Hilali A. A., Alazzam M. B. AI-Assisted Resource Allocation in Project Management. *IEEE Xplore*, 2023. Available at : https://www.doi.org/10.1109/icacite57410.2023.10182760. (accessed 15.12.2024).
10. Narasimha M. M. Literature Review AI-Powered Agile: Reimagining Software Development through Automation, XAI, and Continuous Learning. *International Journal of All Research Education and Scientific Methods*, Vol. 11, No. 12, 2023. Available at : https://www.doi.org/10.56025/ijaresm.2023.1112231949. (accessed 15.12.2024).
11. Meziane F., Vadera S. Artificial Intelligence in Software Engineering: Current Developments and Future Prospects. *Machine Learning: Concepts, Methodologies, Tools and Applications*, 2012. Available at : https://www.doi.org/10.4018/978-1-60960-818-7.CH504. (accessed 15.12.2024).
12. Palacios-González E., Fernández-Fernández H., García-Díaz V., García-Bustelo B. C. P., Lovelle J. M. C. A review of Intelligent Software Development Tools. *Proceedings of the 2008 International Conference on Artificial Intelligence*, 2008. Available at : https://dblp.uni-trier.de/db/conf/icai/icai2008.html#Palacios-GonzalezFGGL08. (accessed 16.12.2024).
13. Bahi A., Gharib J., Gahi Y. Integrating Generative AI for Advancing Agile Software Development and Mitigating Project Management Challenges. *International Journal of Advanced Computer Science and Applications*, Vol. 15, No. 3, 2024. Available at : https://www.doi.org/10.14569/ijacsa.2024.0150306. (accessed 16.12.2024).
14. Sorte B. W., Joshi P. P., Jagtap V. Use of Artificial Intelligence in Software Development Life Cycle: A state of the Art Review. *International Journal of Advanced Engineering and Global Technology*, Vol. 03, No. 03, 2015. Available at : https://www.researchgate.net/profile/Vandana_Jagtap/publication/274247843_Use_of_Artificial_Intelligence_in_Software_Development_Life_Cycle_A_state_of_the_Art_Review/links/551925a20cf273292e70a6da.pdf. (accessed 17.12.2024).
15. *Gartner Peer Insights*. Available at : https://www.gartner.com/peer-insights/home. (accessed 17.12.2024).
16. Patro S. G. P., Sahu K. K. Normalization: A Preprocessing Stage. *International Advanced Research Journal in Science, Engineering and Technology*, Vol. 2, No. 3, 2015. Available at : http://dx.doi.org/10.17148/IARJSET.2015.2305. (accessed 17.12.2024).

Надійшла (received) 11.12.2024

Відомості про авторів / About the Authors

Копп Андрій Михайлович (Kopp Andrii) – доктор філософії, доцент, Національний технічний університет «Харківський політехнічний інститут», завідувач кафедри програмної інженерії та інтелектуальних технологій управління; e-mail: andrii.kopp@khpi.edu.ua. ORCID: http://orcid.org/0000-0002-3189-5623.

Нестеренко Іван Сергійович (Nesterenko Ivan) – Національний технічний університет «Харківський політехнічний інститут», аспірант; e-mail: ivan.s.nesterenko@cs.khpi.edu.ua. ORCID: https://orcid.org/0009-0002-5842-5426.