

МАТЕРІАЛИ

ІХ науково-практичної конференції студентів,
курсантів, аспірантів, докторантів та молодих учених
“АКТУАЛЬНІ ПИТАННЯ
ЗАСТОСУВАННЯ СПЕЦІАЛЬНИХ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМ”

з нагоди 20-ї річниці утворення
Державної служби спеціального зв'язку та захисту
інформації України

23 червня 2026 року

Матеріали IX науково-практичної конференції студентів, курсантів, аспірантів, докторантів та молодих учених “Актуальні питання застосування спеціальних інформаційно-комунікаційних систем”. Київ : ІСЗЗІ КПІ ім. Ігоря Сікорського, 2026. 581с.

У матеріалах IX науково-практичної конференції студентів, курсантів, аспірантів, докторантів та молодих учених “Актуальні питання застосування спеціальних інформаційно-комунікаційних систем” опубліковано тези доповідей, в яких висвітлюються питання дослідження, аналізу та узагальнення нових теоретичних і практичних наукових результатів у сферах кібербезпеки, кіберзахисту, інформаційних технологій та електронних комунікацій. Підвищення професійної підготовки майбутніх офіцерів сектору безпеки і оборони шляхом формування лідерських якостей і фахових компетентностей в умовах сучасних безпекових викликів, зокрема правового режиму воєнного стану. Залучення здобувачів вищої освіти до активної наукової діяльності з використанням новітніх інформаційних технологій і штучного інтелекту в сфері кібербезпеки.

РЕЦЕНЗЕНТИ:

Олександр ПУЧКОВ	к.філос.н., професор
Сергій КОНЮШОК	к.т.н., доцент
Владислав ГОЛЬ	к.т.н., професор
Вадим РОМАНЕНКО	к.т.н., доцент
Дмитро МОГИЛЕВИЧ	д.т.н., професор
Ігор СУБАЧ	д.т.н., професор
Ярослав ЗІНЧЕНКО	к.т.н., с.н.с.

*Рекомендовано до друку Вченою радою ІСЗЗІ КПІ ім. Ігоря Сікорського
(протокол № 13 від 22.06.2026).*

Владислав СЕМЕНОВ; Володимир СОКОЛОВ	
Виявлення фішингових посилань на основі аналізу ознак та машинної класифікації.....	553
Вадим СИВАШ, Ігор ЯКОВІВ	
Система контролю доступу до вебресурсів на основі політики атрибутів	555
Вікторія СТЕПАНЮК; Володимир СОКОЛОВ	
Система статичного аналізу безпеки вихідного коду з використанням штучного інтелекту	557
Адріана ТЕСЛЕНКО; Дмитро ШАРАДКІН	
Виявлення ключових часових параметрів клавіатурного почерку для виявлення несанкціонованого доступу	559
Нікіта ТРОЦЬКИЙ; Дмитро ЛАНДЕ	
Методика безкодового програмування в середовищі великих мовних моделей.....	561
Діана УНЄГОВА; Ольга ШЕВЧУК	
Тестування безпеки програмного забезпечення	563
Валерій УСЕНКО; Сергій МІТІН	
Програмний модуль контролю дотримання політики безпеки в службовому чаті месенджера обміну повідомлень в Signal	564
Віталій ЦИГАНОК	
Колективне формулювання головної цілі при моделюванні безпекового середовища.....	565
Назар ЦИНКО; Дмитро ЛАНДЕ	
Інтеграція інформаційно-пошукових систем та великих мовних моделей для семантичного пошуку з питань кібербезпеки	566
Володимир ЧОРНОМОРЕЦЬ, Ігор ЯКОВІВ	
Система керування інформацією інфраструктури кіберзахисту	568
Михайло ШЕЛЕЛЬО; Іван ГОРНІЙЧУК	
Побудова відтворюваних навчальних сценаріїв кіберполігону з імітацією дій користувачів, систем та порушника.....	569
Анастасія ШЛЯХОВА; Ольга ШЕВЧУК	
Система виявлення кібератак з використанням алгоритмів машинного навчання	571

Нікіта ТРОЦЬКИЙ;
Дмитро ЛАНДЕ, д.т.н., професор

МЕТОДИКА БЕЗКОДОВОГО ПРОГРАМУВАННЯ В СЕРЕДОВИЩІ ВЕЛИКИХ МОВНИХ МОДЕЛЕЙ

Анотація. У роботі проаналізовано підходи до безкодового (no-code / low-code) програмування та розглянуто можливості великих мовних моделей (LLM) як засобу розробки програмних рішень. На основі порівняльного аналізу сучасних LLM обґрунтовано вибір оптимальної моделі для реалізації запропонованої методики. Сформовано методику безкодового програмування в середовищі LLM для вирішення прикладних задач.

Summary. The paper analyzes no-code / low-code programming approaches and examines LLMs as a tool for developing software solutions. Based on a comparative analysis of modern LLMs, the optimal model for implementing the proposed methodology is selected. A methodology for no-code programming in the LLM environment is formed for solving applied problems.

Ключові слова: безкодове програмування, великі мовні моделі, промпт-інжиніринг, текстова аналітика, RAG, формальні примітиви управління.

Сучасний ринок програмного забезпечення демонструє стрімке зростання попиту на no-code та low-code платформи, що дозволяють нефахівцям створювати програмні рішення без глибоких знань мов програмування. Аналіз існуючих підходів виявив ключові категорії: візуальні конструктори (Bubble, Webflow, AppGyver), системи автоматизації бізнес-процесів (Zapier, Make, Power Automate) та LLM-орієнтовані платформи (GitHub Copilot, Cursor, Replit Ghostwriter). Принципова відмінність LLM-орієнтованих рішень полягає у використанні природної мови як основного інтерфейсу взаємодії, що знімає обмеження традиційних no-code платформ у частині гнучкості та виразності задач і відкриває нові можливості для автоматизації складних сценаріїв.

Порівняльний аналіз сучасних великих мовних моделей – GPT-4o, Claude 3.7 Sonnet, Gemini 1.5 Pro та Llama 3.3 – проводився за критеріями точності розуміння природномовних інструкцій, підтримки україномовного вводу, доступності через API та якості структурованої відповіді. Особливу увагу приділено підтримці контекстного вікна та обробки документів. Для реалізації прототипу обрано модель Llama

3.3-70b (через безкоштовний API Groq) завдяки відкритому доступу та підтримці україномовних запитів; Claude 3.7 Sonnet і GPT-4o визначено як теоретично потужніші, проте обмежені вартістю альтернативи в навчальному середовищі.

Методика ґрунтується на концепції промпт-програм – природномовних інструкцій із трьох базових примітивів: ЦИКЛ (For), УМОВА (If/Else) та ФУНКЦІЯ. ЦИКЛ задає перебір елементів колекції документа; УМОВА описує розгалуження логіки «якщо – то – інакше»; ФУНКЦІЯ задає параметризовану дію і формат виводу. Конвеєр обробки складається з п'яти кроків: завантаження документа (PDF, TXT, CSV) як RAG-контексту → конструювання промпт-програми з примітивів → трансляція у природномовний запит → виконання LLM → відображення структурованого результату. Ключові вимоги до методики: підтримка завантаження документів як контексту, збереження сесійної історії та виведення результатів у форматах таблиць і списків Markdown.

Апробація методики проводилась на задачах аналізу текстових документів: витягування структурованої інформації (імена, дати, цитати), класифікації абзаців за змістом та побудови зведених таблиць на основі вхідних даних. Реалізований веб-додаток (Flask + Groq API) надає два режими роботи: конструктор промпт-програм із примітивами та вільний текстовий чат із підтримкою завантаження документів. Апробація підтвердила доступність підходу для нефахівців: побудова промпт-програми займає 1–3 хвилини без жодного рядка коду. Виявлено обмеження: залежність якості результату від точності формулювань у примітивах, складність верифікації відповіді та чутливість до обсягу вхідного документа. Подолання цих обмежень є предметом подальших досліджень.

Висновки. Проведений аналіз підходів до no-code / low-code програмування та можливостей LLM підтвердив перспективність використання великих мовних моделей як основи методики безкодового програмування. Запропонована методика, що ґрунтується на конструюванні природномовних промпт-програм із примітивів ЦИКЛ, УМОВА та ФУНКЦІЯ, дозволяє нефахівцю формулювати прикладні задачі й отримувати структуровані результати без написання традиційного коду. Реалізований прототип підтвердив практичну застосовність підходу в умовах реальних задач. Подальші дослідження мають бути спрямовані на розширення набору примітивів, підвищення якості верифікації результатів та адаптацію методики до спеціалізованих предметних областей.