

УДК 004.415.2

*Мельничук К. В., здобувач вищої освіти;  
Зелінська О. В., канд. техн. наук, доцент,  
в. о. завідувача кафедри інформаційних технологій,  
Донецький національний університет імені Василя Стуса*

## **ЕФЕКТИВНІСТЬ АРІ ДЛЯ ОПТИМІЗАЦІЇ НАПИСАННЯ ТА ВИКОРИСТАННЯ СКЛАДНИХ ВЕБПРОГРАМ**

Ключові слова: АРІ, веб, база даних, вебпроект, вебзастосунок.

**Вступ.** У сучасному інформаційному суспільстві вебпрограмування відіграє ключову роль у розробці та підтримці вебзастосунків і систем. Завдяки постійному росту складності та обсягу вебпроект, зростає і потреба в ефективних інструментах для оптимізації процесу розробки та використання вебпрограм. Одним із важливих компонентів цієї оптимізації є використання програмованих інтерфейсів застосунків (АРІ), які дають можливість спрощення написання коду та покращення взаємодії між різними компонентами вебзастосунків. У цій роботі ми маємо на меті дослідити, як АРІ впливають на ефективність написання та використання складних вебпрограм, які фактори визначають ефективність АРІ які методи та метрики можна використовувати для оцінки ефективності АРІ, та які виклики й обмеження можуть виникати під час використання АРІ.

**Актуальність.** Конкуренція в галузі веброзробки зростає, і розробники постійно шукають способи підвищення ефективності та якості своїх продуктів. Використання АРІ дає можливість створити вебзастосунки швидше та ефективніше. Використання АРІ стає необхідним елементом стратегії розробки вебзастосунків для досягнення конкурентних переваг та задоволення потреб сучасного інформаційного ринку.

1. Роль АРІ у вебпрограмуванні [1, 3]: АРІ є інтерфейсами, які дають можливість взаємодії між різними компонентами вебзастосунків. Вони складаються з набору правил, методів, параметрів та форматів обміну даними, які дають змогу розробникам використовувати функціональні можливості інших програм або сервісів без необхідності знову розробляти весь функціонал. Наприклад, АРІ можуть надавати доступ до баз даних, картографічних сервісів, соціальних мереж, платіжних систем тощо.

2. Використання АРІ дає змогу розробникам [1, 5]: зменшити зусилля, необхідні для написання коду; підвищити швидкість розробки; забезпечити більшу стабільність та безпеку застосунків, використовуючи перевірені АРІ; покращити можливості розширення та масштабування проєктів, оскільки АРІ дають змогу додавати новий функціонал без значного впливу на наявний код.

Деякими з найпопулярніших АРІ у вебпрограмуванні є [3]: Google Maps АРІ, Facebook Graph АРІ, Stripe АРІ, YouTube Data АРІ тощо.

3. Ефективність АРІ для оптимізації написання коду. АРІ можуть бути ефективними для оптимізації написання коду завдяки таким аспектам [1]:

**Інтеграція:** API дають змогу легко інтегрувати сторонні сервіси та бібліотеки у вебзастосунки, зменшуючи потребу в розробці власного функціоналу. Розробники можуть використовувати готовий функціонал, наданий API, не потребуючи написання коду з нуля для кожної задачі. Наприклад, для отримання координат місцеперебування користувача за його IP-адресою розробник може використати IP Geolocation API, передавши IP-адресу як параметр і отримавши JSON-об'єкт із координатами широти та довготи.

**Повторне використання:** використовуючи API, розробники можуть створювати код, який може бути використаний в інших проєктах, що сприяє повторному застосуванню та уникненню дублювання коду.

**Покращення продуктивності:** API допомагають використовувати готові рішення для рутинних задач, даючи змогу розробникам зосередитися на складніших аспектах проєкту.

Завдяки API розробники можуть легко додавати новий функціонал до своїх проєктів, використовуючи наявний API або створюючи власні. Також API дають змогу розділити логіку та інтерфейс вебзастосунків, що сприяє модульності та гнучкості коду. Наприклад, розробник може створити [3] RESTful API для свого вебзастосунку, який надає доступ до даних та бізнес-логіки через HTTP-запити. Потім розробник може створити різні клієнтські інтерфейси для вебу, мобільних або настільних пристроїв, які використовуватимуть цей API для отримання та надсилання даних.

4. Виклики та обмеження використання API [2, 5]. Незважаючи на багато переваг, використання API також вносить виклики та обмеження:

- залежність від сторонніх сервісів: використання зовнішніх API робить застосунок залежним від доступності цих API та може призвести до проблем, якщо сторонні сервіси перестануть підтримувати свої API;

- безпека: використання API може створити потенційні точки атаки, тому необхідно ретельно контролювати доступ до API та дотримуватися найкращих практик забезпечення безпеки.

### **Висновки**

Використання API є важливим інструментом для оптимізації написання та використання складних вебпрограми. Вони сприяють підвищенню продуктивності розробки, зменшенню зусиль, необхідних для написання коду, та покращенню якості веб-застосунків. Однак важливо ретельно розглядати вибір API та дотримуватися найкращих практик забезпечення безпеки та стабільності.

### **Список використаних джерел**

1. The Role of APIs in Web Development. URL: <https://appmaster.io/blog/apis-in-web-development/>
2. Simplify Your API: A Path to Efficiency and Ease URL: <https://softwaremill.com/simplify-your-api-a-path-to-efficiency-and-ease/>
3. Web API in Application Development: Concepts & Practical Uses URL: <https://medium.com/theymakedesign/web-api-in-application-development-4f11a466da6b/>

4. Дев'ять переваг розробки API у сучасній веб розробці URL: <https://www.artfans.info/9-api-development-benefits-in-modern-web-development/>

5. Інтеграція API у веб-сайт: як використовувати дані сторонніх сервісів для поліпшення функціоналу сайту URL: <https://it-rating.ua/integratsiya-api-v-veb-sayt-kak-ispolzovat-dannye-storonnih-servisov-dlya-uluchsheniya-funktsionala-sayta>