

Олександр Рюмін

Магістрант спеціальності комп'ютерні системи та мережі, Національний університет «Запорізька політехніка» м. Запоріжжя, Україна

0009-0000-5181-7422

sascha.rumin@gmail.com

Степан Скрупський

к.т.н., доцент, доцент кафедри комп'ютерних систем та мереж

Місце роботи: Національний університет «Запорізька політехніка», м. Запоріжжя, Україна

0000-0002-9437-9095

sskrupsky@gmail.com

СИСТЕМНИЙ ПІДХІД ДО ВИБОРУ ТЕХНОЛОГІЙ ВЕБРОЗРОБКИ

Анотація. У статті проведено аналіз підвищення ефективності вебсистем шляхом порівняння архітектур, мов програмування та фреймворків. Запропоновано системний підхід до вибору технологічного стеку, що враховує технічні та освітні критерії. Узагальнено сучасні тенденції, визначено проблеми вибору технологій і сформовано критерії оцінювання ефективності.

Розроблений вебзастосунок проводить автоматичне оцінювання продуктивності за показниками швидкодії, використання пам'яті та стабільності, дозволяє формувати власні технологічні стеки та зберігати результати для порівняльного аналізу. Підхід може використовуватись у навчальному процесі й наукових дослідженнях. Подальші роботи спрямовані на розширення критеріїв оцінювання та побудову дерева прийняття рішень для вибору оптимального стеку.

Ключові слова: веброзробка; архітектура вебсистем; продуктивність; підвищення ефективності; вибір технологій; фреймворки; системний підхід.

1. ВСТУП

Розвиток вебтехнологій протягом останнього десятиліття суттєво розширив можливості побудови вебсистем, охоплюючи різноманітні архітектурні моделі – від класичних монолітних до мікросервісних і серверлес-рішень. Зі зростанням складності проєктів розробники все частіше стикаються з необхідністю обрати оптимальні технології, що забезпечують баланс між продуктивністю, стабільністю та масштабованістю системи.

В цьому контексті актуальним є завдання аналізу та порівняння сучасних мов програмування, фреймворків і архітектурних підходів для підвищення ефективності процесів розробки вебсистем. Аналіз проведено на основі матеріалів порталів DOU, SoftServe та Wezom, що дозволив виявити основні труднощі, з якими стикаються початківці та студенти при виборі технологічного стеку.

Основна проблема полягає у відсутності систематизованого підходу до вибору архітектур та технологій, який враховував би як технічні критерії (швидкодія виконання коду, використання пам'яті, стабільність), так і освітні аспекти (складність освоєння, вартість навчання, доступність ресурсів).

Мета - це підвищення ефективності веброзробки шляхом ґрунтовного вибору технологій та архітектур, який можна здійснити за допомогою запропонованого системного підходу

2. АНАЛІЗ ПРОЦЕСУ ВИБОРУ ТЕХНОЛОГІЙ

Сучасний ринок веброзробки характеризується великою кількістю доступних мов програмування, фреймворків та архітектурних рішень. Для новачків і студентів вибір технологій є непростим завданням через різний рівень освоєння, потребу у балансі між продуктивністю та простотою, а також вплив на кар'єрні перспективи.

Аналіз проведено на основі кількох авторитетних джерел, що висвітлюють особливості вибору технологій у 2025 році. Таблиця нижче узагальнює ключові тези щодо процесу вибору та основні фактори, які варто враховувати при прийнятті рішень.

Таблиця 1

Аналіз змісту публікацій, присвячених вибору технологій у веброзробці

Джерело (назва статті / публікації)	Ключові висновки та проблематика
Найновіші технології веброзробки 2025	Вибір технологій ускладнюється великою кількістю фреймворків і архітектур; автори підкреслюють потребу оцінювати продуктивність, масштабованість та простоту освоєння [1].
30% українських ІТ-фахівців планують вивчати нову мову	Для новачків складно обрати мову через відсутність систематизованої інформації про популярність, попит на ринку праці та перспективи кар'єрного розвитку [2].
ІТ-ринок в Україні: аналітика DOU	Складність вибору пов'язана з високою конкуренцією на ринку та великою кількістю вакансій із різними вимогами; новачкам важко визначити, які технології дадуть перевагу у працевлаштуванні [4].
Архітектура ПЗ: ефективність і безпека	Вибір технологій впливає на продуктивність і безпеку системи; неправильний вибір архітектури може ускладнити масштабування та підтримку проєкту [5].
Порівнюємо React, Angular і Vue	Вибір між популярними фронтенд-фреймворками залежить від складності освоєння, підтримки спільноти та специфіки проєкту; кожен фреймворк має свої сильні та слабкі сторони [5].

Таблиця демонструє, що у різних джерелах виділяють подібні фактори процесу вибору технологій: велика кількість варіантів, баланс між продуктивністю та простотою освоєння, вплив на кар'єрні перспективи та підтримку проєкту. Ці аспекти є ключовими при розробці системного підходу до вибору технологічного стеку для новачків та студентів.

3. ЗАПРОПОНОВАНИЙ АЛГОРИТМ ВИРІШЕННЯ ПРОБЛЕМИ

Для забезпечення системного підходу до вибору технологій запропоновано алгоритм, який поєднує аналітичну та експериментальну складові. На першому етапі проводиться аналітичний відбір критеріїв оцінювання. Далі, в залежності від типу критерію (технічний або освітній), будуть проводитися відповідні дії. Якщо критерії технічні, то їх доцільно перевірити за допомогою вебзастосунку, який буде досліджувати обрані технології за відповідними критеріями технології. В інакшому випадку – буде проведений аналіз ринку з надання навчального матеріалу та курсів. Наступним є поєднання результатів досліджень з аналізом ринку. Якщо результати досліджень не відповідатимуть дійсності, тобто даним з аналізу статей чи інформації від освітніх платформ, то буде проводитися корегування застосунку. У результаті має бути отримане дерево прийняття рішень з вибору архітектур і технологій веброзробки. Алгоритм системного підходу до вибору технологій веброзробки наведений на рис. 1.

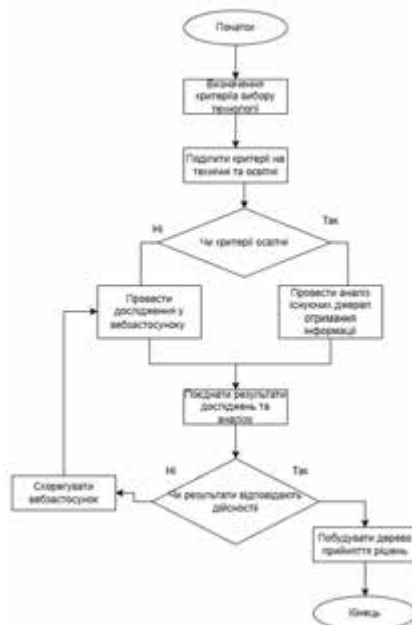


Рисунок 1. Алгоритм системного підходу до вибору технологій веброзробки

Алгоритм включає послідовні етапи – аналіз параметрів, дослідження, виконання замірів, статистичну обробку та формування рекомендацій.

4. РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

Для реалізації запропонованого підходу було розроблено вебзастосунок, який був обраний як інструмент для проведення дослідження технологій веброзробки за технічними критеріями. Дане рішення було прийнято для згрупування платформ для дослідження в один застосунок.

Основна ідея полягає у наданні користувачеві можливості обрати готовий технологічний шаблон або самостійно сформувати власний стек із доступних архітектур і мов програмування.

Після запуску, система автоматично проводить заміри за обраними критеріями (швидкодія, використання пам'яті, стабільність тощо) та зберігає результати у базу даних для подальшого аналізу і порівняння.

Даний вебзастосунок передбачає розширення шляхом створення нових шаблонів, додавання нових архітектур чи технологій веброзробки у наявну базу даних і корегування та доповнення її дослідницької частини. Дане рішення впровадить гнучкість у середовище, яка дозволить покращувати та підтримувати актуальність системи. Алгоритм роботи вебзастосунку наведений на рисунку 2.

Таким чином, даний алгоритм відповідає меті і здатний частково, оскільки потрібен аналіз не тільки за технічними критеріями, вирішити проблему, що поставлена у цій статті. Результати досліджень, що зроблені даним вебзастосунком, у сукупності з аналізом ринку праці матимуть більше конкретики і точності у підборі технологій веброзробки новачками та студентами.

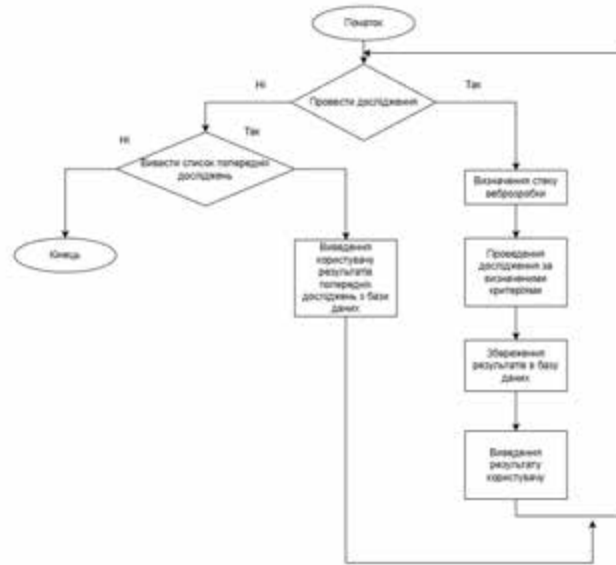


Рис. 2. Алгоритм роботи вебзастосунку

5 РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ

Вибір технологій і архітектури безпосередньо впливає на ключові показники роботи вебсистеми – латентність відгуку, пропускну здатність, використання оперативної пам'яті та здатність до масштабування. Агреговані бенчмарки демонструють, що різні реалізації одних і тих самих функціональних рішень можуть розрізнятися за пропускну здатністю та затримками у декілька разів, отже коректна підбірка стеку та його оптимізація дають реальні, кількісно відчутні вигоди при навантаженні. Оскільки абсолютні значення продуктивності залежать від конкретної апаратної конфігурації, версій програмного забезпечення та характеру навантаження, акцент зроблено на відносних змінах (%) та на перцентилях (p50, p95, p99), які краще відображають реальний вплив підвищення ефективності та архітектурних рішень.

Агреговані результати бенчмарків показують, що для різних реалізацій фреймворків пропускну здатність у «простих» сценаріях може відрізнятися в межах від кількох тисяч до десятків/сотень тисяч запитів за секунду; у прикладних порівняннях це часто відповідає співвідношенням порядку 2-5× між повільнішими і високопродуктивними реалізаціями [6]. Типові затримки для оптимізованих бекенд-запитів (p50) в простих сценаріях складають орієнтовно $\leq 5-20$ ms, тоді як високі перцентили (p95, p99) при реалістичних навантаженнях можуть зрости до десятків або сотень мілісекунд.

Особливої уваги заслуговує поведінка serverless-підходів (cold start): холодний старт функції може додавати затримку в діапазоні від десятків мілісекунд до понад 1 с, причому величина додаткової затримки залежить від рантайму та платформи; сучасні оптимізації та механізми продовження контейнерів знижують цей ефект у певних конфігураціях [7]. Кешування та edge/CDN-рішення показують практично відтворюваний позитивний ефект: прості HTTP-кеші дають звичайне зниження латенції в межах $\approx 20-40\%$ для повторюваних запитів, а комбіновані edge-стратегії у прикладах реалізацій можуть зменшувати сприйману затримку кінцевого користувача на $\approx 50\%$ і більше.

Практичні впливи оптимізацій та вибору мови/рантайму в дослідженнях узагальнено таким чином: конфігураційні оптимізації (впровадження кешування, мінімізація залежностей, підбір легших бібліотек) зазвичай дають двоцифрове покращення (зниження p50/p95 на приблизно 10-35%); у високопродуктивних

сценаріях використання компільованих мов / легких рантаймів (наприклад, Go або Rust) може забезпечити істотне підвищення пропускну здатності (в окремих бенчмарках – близько $\approx 2\times$ порівняно з типовими JS-реалізаціями) [6]. Водночас перехід на мікросервісну архітектуру дає переваги в масштабованості та гнучкості, але може підвищити сумарне споживання пам'яті і накладні мережеві затримки – їхній ефект залежить від якості декомпозиції і налаштування міжсервісної взаємодії.

Таким чином, правильний вибір архітектури та технологій веброзробки об'єктивно позитивно впливає на роботу вебзастосунку та його підтримку.

6. ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У ході дослідження було проаналізовано особливості сучасних технологій веброзробки. У якості результату систематичного підбору до вибору технологій було обрано створення дерева прийняття рішення.

Для створення дерева прийняття рішення вибору інструментів веброзробки для новачків та студентів було обрано двоетапний підхід. Першим підходом є проведення дослідження за технічними критеріями. Другим – аналіз ринку праці, що охоплює як пропозиції на ринку, так і надавачів освітніх програм. Розроблений підхід дозволяє систематизувати процес вибору технологічного стеку з урахуванням технічних та освітніх критеріїв.

Таким чином, запропонований системний підхід дозволяє підвищити ефективність веброзробки шляхом ґрунтовного вибору технологій та архітектур.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Євген Паталяк (2025) Інноваційні технології розробки веб-сайтів: тренди та рішення 2025. Available at: <https://wezom.com.ua/ua/blog/naynovishi-tehnologiyi-veb-rozrobki-2025-yak-stvoryuyutsya-suchasni-sayti> (Accessed: 10 November 2025);
2. Дмитро Сімагін (2025) 30% українських it-фахівців планують вивчати нову мову програмування у 2025 році, Highload.tech - медіа для розробників. Available at: <https://highload.tech/uk/30-ukrayinskyh-it-fahivtsiv-planuyut-vyvchaty-novu-movu-programuvannya-u-2025-rotsi/> (Accessed: 10 November 2025);
3. DOU (2025) Найкраще півріччя за всю повномасштабку. огляд it-ринку праці, червень 2025, DOU. Available at: <https://dou.ua/lenta/articles/it-job-market-june-2025/> (Accessed: 10 November 2025);
4. Гузенко Сергій (2025) Архітектура програмного забезпечення: Оптимізація коду, безпека та управління проєктами для успіху. Available at: <https://wezom.com.ua/ua/blog/arhitektura-programnogo-zabezpechennya> (Accessed: 10 November 2025);
5. Hutsuliak, O. (2022) Порівнюємо react, angular і Vue - найпопулярніші бібліотеки й фреймворки у 2022 році, DOU. Available at: <https://dou.ua/forums/topic/39933/> (Accessed: 10 November 2025);
6. TechEmpower (2025) TechEmpower framework benchmarks, TechEmpower. Available at: <https://www.techempower.com/benchmarks/#section=data-r23> (Accessed: 10 November 2025);
7. AWS (2025) Serverless function, Faas serverless - Aws Lambda - AWS, AWS Lambda. Available at: <https://aws.amazon.com/lambda/> (Accessed: 10 November 2025).

MINISTRY OF EDUCATION
AND SCIENCE OF UKRAINE

NATIONAL UNIVERSITY
OF LIFE AND ENVIRONMENTAL
SCIENCES OF UKRAINE

FACULTY OF INFORMATION
TECHNOLOGY

МІНІСТЕРСТВО ОСВІТИ
І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БІОРЕСУРСІВ І
ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

PROCEEDINGS

XIII International scientific
and practical conference

**GLOBAL AND
REGIONAL PROBLEMS OF
INFORMATIZATION IN
SOCIETY AND
NATURE USING
'2025**

13-14 November 2025

Kyiv, NULES of Ukraine

Kyiv 2025

МАТЕРІАЛИ

XIII Міжнародної науково-
практичної конференції

**ГЛОБАЛЬНІ ТА
РЕГІОНАЛЬНІ ПРОБЛЕМИ
ІНФОРМАТИЗАЦІЇ В
СУСПІЛЬСТВІ І
ПРИРОДОКОРИСТУВАННІ
'2025**

13-14 листопада 2025 року

Київ, НУБіП України

Київ 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ
І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

ХІІІ Міжнародної науково-практичної конференції

ГЛОБАЛЬНІ ТА РЕГІОНАЛЬНІ ПРОБЛЕМИ ІНФОРМАТИЗАЦІЇ В СУСПІЛЬСТВІ І ПРИРОДОКОРИСТУВАННІ '2025

13-14 листопада 2025 року

Київ, НУБіП України

Київ 2025

УДК 004

Рекомендовано до друку вченою радою факультету інформаційних технологій Національного університету біоресурсів і природокористування України (протокол № 4 від 18.12.2025).

Укладач: д.т.н., доцент Шкарупило В.В.

Збірник матеріалів XIII Міжнародної науково-практичної конференції "Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні '2025", 13–14 листопада 2025 року, НУБіП України, Київ. – К.: НУБіП України, 2025. – 206 с.

Відповідальність за зміст публікацій несуть автори.

© Національний університет біоресурсів
і природокористування України, 2025