

УДК 004.738.5

DOI <https://doi.org/10.32782/2663-5941/2026.1.2/16>**Квітка Д.І.**<https://orcid.org/0009-0008-5096-2714>

Криворізький національний університет

**Музика І.О.**<https://orcid.org/0000-0002-9202-2973>

Криворізький національний університет

**Кузнєцов Д.І.**<https://orcid.org/0000-0002-2021-5207>

Криворізький національний університет

## МЕТОДИ ОПТИМІЗАЦІЇ БЛОГУ НА БАЗІ СУЧАСНИХ ТЕХНОЛОГІЙ FRONTEND РОЗРОБКИ

У статті розглянуто методи оптимізації інформаційного блогу на основі сучасних технологій frontend-розробки з метою підвищення продуктивності, зручності використання та якості користувацького досвіду. Актуальність дослідження зумовлена стрімким розвитком вебтехнологій і зростанням вимог до швидкодії, адаптивності та доступності вебінтерфейсів. Об'єктом дослідження є процес розробки та оптимізації клієнтської частини вебзастосунків, а предметом – методи та інструменти оптимізації інтерфейсу блогу з використанням сучасних frontend-технологій. Авторами проаналізовано етапи frontend-розробки, принципи проєктування інтерфейсу та підходи до побудови інформаційної архітектури блогівих платформ. Особливу увагу приділено використанню шаблонізатора Blade та фреймворку Bootstrap 5 для створення адаптивного інтерфейсу, а також методам оптимізації стилів і ресурсів. Розглянуто практичні аспекти застосування кешування CSS, мінімізації та структурування стилів, оптимізації завантаження статичних ресурсів і впровадження адаптивного дизайну. Показано, що комплексне застосування зазначених методів дозволяє скоротити час завантаження сторінок, зменшити навантаження на браузер і підвищити стабільність роботи вебінтерфейсу. У межах дослідження використано методи аналізу та узагальнення наукових джерел, порівняльний аналіз frontend-технологій, проєктно-конструкторські методи та виконано практичну реалізацію інтерфейсу блогу. Отримані результати підтверджують доцільність застосування сучасних frontend-інструментів у поєднанні з методами оптимізації для створення ефективних інформаційних блогів. Практичне значення роботи полягає у можливості використання запропонованих підходів під час розробки та модернізації вебресурсів, орієнтованих на публікацію та поширення контенту в умовах високих вимог до продуктивності та зручності користування.

**Ключові слова:** frontend-розробка, вебінтерфейс, оптимізація, блог, адаптивний дизайн, кешування стилів, Blade, Bootstrap 5.

**Постановка проблеми.** У сучасних умовах стрімкого розвитку цифрових технологій інформаційні вебресурси, зокрема блогіві платформи, відіграють важливу роль у поширенні знань, аналітичних матеріалів та корпоративної інформації. Водночас зростання обсягів контенту, різноманіття пристроїв доступу та підвищені очікування користувачів щодо швидкодії й зручності використання висувають нові вимоги до клієнтської частини вебзастосунків. Недостатня оптимізація

frontend-складової призводить до збільшення часу завантаження сторінок, перевантаження браузера, погіршення користувацького досвіду та, як наслідок, зниження ефективності функціонування інформаційного ресурсу.

Особливо гостро ця проблема проявляється у блогах, де основний акцент зроблено на споживанні текстового та мультимедійного контенту. Неоптимізовані стилі, надмірна кількість CSS і JavaScript файлів, відсутність механізмів



кешування та некоректна реалізація адаптивного дизайну негативно впливають на продуктивність вебінтерфейсу. Це ускладнює доступ до інформації для користувачів мобільних пристроїв, погіршує показники доступності та знижує конкурентоспроможність вебресурсу в умовах сучасного інформаційного середовища.

Незважаючи на наявність значної кількості frontend-фреймворків і бібліотек, практичні питання їх ефективного поєднання та оптимального використання залишаються недостатньо систематизованими. Зокрема, актуальною є проблема впровадження комплексного підходу до оптимізації блогівих платформ на основі сучасних технологій frontend-розробки з урахуванням продуктивності, адаптивності та зручності користування. Виникає потреба в дослідженні методів оптимізації інтерфейсу блогу, які поєднують сучасні інструменти розробки з практичними рішеннями щодо скорочення часу завантаження, підвищення стабільності роботи та покращення користувацького досвіду. Саме розв'язанню цієї проблеми присвячено наукову статтю.

**Аналіз останніх досліджень і публікацій.** Питання оптимізації вебінтерфейсів у сучасній frontend-розробці впродовж останніх років перебуває у фокусі як академічних досліджень, так і практичних рекомендацій провідних технологічних компаній. Найбільш впливовим напрямом є вимірювання та покращення реального користувацького досвіду (UX) через показники продуктивності. У цьому контексті вагому роль відіграє підхід Core Web Vitals, який систематизує метрики, пов'язані зі швидкістю завантаження, інтерактивністю та візуальною стабільністю сторінки. Документація Google підкреслює, що ці метрики відображають «реальний» досвід користувачів і рекомендуються як орієнтир для досягнення якісної роботи сайтів, зокрема з позицій сторінкового досвіду та загальної зручності взаємодії [1, 2, 3].

Другий помітний блок досліджень і публікацій присвячено оптимізації критичного шляху рендерингу (Critical Rendering Path), адже затримки на етапі обробки стилів і скриптів є типовою причиною повільного відображення контенту. Відповідно до матеріалів web.dev, одним із поширених підходів виступає виділення критичного CSS (Critical CSS) – тобто стилів, необхідних для відображення контенту «першого екрану». Це дозволяє прискорити первинне відображення контенту, а некритичні стилі відкладати або підвантажувати асинхронно [4]. Додатково варто відзначити, що

методи оптимізації доставки CSS можуть бути реалізовані як за допомогою ручних підходів, які передбачає проограміст, так і через інструменти чи модулі оптимізації, що автоматизують визначення критичних і некритичних стилів та їх підключення [5].

Окремий напрям публікацій формують рекомендації з HTTP-кешування та керування повторним використанням ресурсів. Для контентних платформ, зокрема блогів, де значна частина статичних ресурсів (CSS, шрифти, зображення) змінюється відносно рідко, механізми кешування є одним із найефективніших способів зменшити кількість повторних завантажень і скоротити час відкриття сторінок. MDN детально описує призначення директив заголовка Cache-Control для керування кешуванням у браузерях та проміжних кешах (CDN/проксі), а також практику «примусової перевірки актуальності» через поєднання Cache-Control із валідаторами на кшталт ETag [6, 7]. У сукупності ці підходи вважаються базовими для побудови стабільної та передбачуваної політики кешування статичних ресурсів.

Не менш важливою є тематика доступності вебконтенту, оскільки оптимізація не зводиться лише до швидкодії. Доступність прямо впливає на здатність користувачів взаємодіяти з контентом, зокрема на мобільних пристроях або з використанням допоміжних технологій. Актуальним нормативним підґрунтям виступає стандарт WCAG 2.2, опублікований W3C як рекомендація та такий, що охоплює широкий спектр вимог до доступності [8, 9].

У прикладній площині дослідження оптимізації часто поєднують архітектурні рішення та інструменти UI-розробки. Для блогівих платформ типовими є адаптивна верстка, повторне використання компонентів, мінімізація дублювання стилів, організація шаблонів і прискорення завантаження. У межах розробки блогу для кафедри комп'ютерних систем та мереж, який слугує підґрунтям для цієї статті, акцент зроблено на сучасних інструментах frontend-реалізації (Laravel Blade і Bootstrap 5) та практиках підвищення ефективності інтерфейсу, включно із застосуванням кешування стилів.

Отже, аналіз останніх публікацій дозволяє виокремити ключові напрями, що визначають сучасну оптимізацію блогівих вебінтерфейсів: орієнтація на метрики реального UX (Core Web Vitals), оптимізація рендерингу через критичні або некритичні ресурси (насамперед CSS), коректна політика кешування на рівні HTTP, а також

забезпечення доступності відповідно до WCAG. Саме інтеграція цих підходів у практичну реалізацію блогової платформи має формувати основу подальшого дослідження.

**Постановка завдання.** Метою статті є дослідження та обґрунтування методів оптимізації інформаційного блогу на основі сучасних технологій frontend-розробки з метою підвищення продуктивності, адаптивності та зручності користування вебінтерфейсом. Для досягнення поставленої мети необхідно проаналізувати сучасні підходи до оптимізації клієнтської частини вебзастосунків, визначити ефективні методи оптимізації стилів і статичних ресурсів, а також оцінити доцільність їх практичного застосування під час розробки та модернізації на прикладі блогу кафедр комп'ютерних систем та мереж.

**Виклад основного матеріалу.** Сучасні інформаційні блоги є важливою складовою вебпростору, оскільки вони забезпечують поширення аналітичного, навчального та корпоративного контенту для широкої аудиторії. В умовах зростання конкуренції між вебресурсами ключовими чинниками їх успішного функціонування стають швидкодія, адаптивність, стабільність роботи та зручність користування. Саме тому оптимізація клієнтської частини блогу на основі сучасних технологій frontend-розробки набуває особливої актуальності.

Frontend-складова вебзастосунку безпосередньо впливає на перший контакт користувача з ресурсом, швидкість відображення контенту та загальний користувацький досвід. Для блогових платформ це має особливе значення, оскільки основна взаємодія користувача зводиться до читання матеріалів, навігації між публікаціями та пошуку інформації. Надмірно складна структура інтерфейсу, неефективне завантаження стилів або відсутність оптимізації статичних ресурсів можуть призвести до зниження швидкодії та втрати аудиторії.

У межах дослідження встановлено, що оптимізація frontend-рівня має розглядатися не як окремий етап після завершення розробки, а як інтегрований процес, що охоплює всі стадії життєвого циклу вебінтерфейсу: від проектування структури та прототипування до реалізації, тестування й подальшої підтримки. Такий підхід дозволяє уникнути архітектурних помилок і забезпечити стабільну роботу блогу в умовах зростання обсягів контенту.

Початковим етапом оптимізації є грамотне проектування інтерфейсу та інформаційної архі-

тектури блогу (рис. 1). На цьому етапі визначається структура сторінок, логіка навігації, розташування ключових елементів і сценарії взаємодії користувача з системою. Для інформаційних блогів доцільним є використання ієрархічної структури, що включає головну сторінку зі стрічкою публікацій, сторінки категорій, окремі сторінки статей та допоміжні розділи.



Рис. 1. Процес прототипування інтерфейсу

Рациональне проектування інтерфейсу зменшує кількість зайвих елементів, спрощує навігацію та створює передумови для подальшої технічної оптимізації. На рис. 2 продемонстровано взаємозв'язок між аналітичними та практичними етапами проектування.

Фундамент сучасного frontend-інтерфейсу становлять HTML, CSS і JavaScript. Як показує аналіз, проведений авторами, їх ефективне поєднання дозволяє забезпечити структурованість, візуальну привабливість і динамічність блогу. Особливу увагу приділено застосуванню семантичної HTML-розмітки, яка покращує доступність контенту та спрощує індексацію сторінок.

Для реалізації інтерфейсу блогу доцільним є використання шаблонізатора Blade у складі фреймворку Laravel. Blade дозволяє не лише уникати дублювання коду завдяки шаблонам і секціям, а також централізовано керувати структурою сторінок, що спрощує підтримку та масштабування інтерфейсу.

Фреймворк Bootstrap 5 використовується як інструмент швидкої реалізації адаптивного дизайну. Основні функції технології Bootstrap 5 та можливості інтегрування її у верстку блогу наведено у табл. 1.

Варто також відзначити, що сіткова система та готові UI-компоненти забезпечують коректне

<b>Етап 1. Аналіз потреб та цільової аудиторії</b>			
Визначення груп користувачів	Аналіз їхніх сценаріїв взаємодії	Огляд освітніх сайтів і блогів	
<b>Етап 2. Формування інформаційної структури</b>			
Структура блогу	Основні розділи та сторінки	Логіка навігації та переходів	
<b>Етап 3. Створення прототипів</b>			
Wireframes головної сторінки	Макет сторінки статті	Меню та стрічка публікацій	Тестування прототипу
<b>Етап 4. Візуальне проєктування (UI Design)</b>			
Кольорова палітра	Система шрифтів	Стилі UI-елементів	Адаптивність інтерфейсу
<b>Етап 5. Реалізація та програмна розробка</b>			
HTML5, CSS3, JavaScript	Інтеграція CMS	Реалізація пошуку, коментарів	Оптимізація швидкодії (кешування, мінімізація)
<b>Етап 6. Тестування</b>			
Функціональне тестування	Usability тестування	Кросбраузерне тестування	Кросплатформне тестування
<b>Етап 7. Впровадження</b>			
Розгортання на сервері	Підключення аналітики та моніторинг	Регулярне оновлення дизайну та контенту	Додавання нового функціоналу

Рис. 2. Схема етапів проєктування та розробки блогу

Таблиця 1

**Основні можливості Bootstrap 5**

Можливість Bootstrap 5	Теоретичне призначення	Практичне використання в блозі
Grid System	Адаптивна багатостовпкова верстка	Формування структури списку постів і блоку контенту
Компоненти UI	Стандартизовані елементи інтерфейсу	Кнопки, модальні вікна, форми, сповіщення
Типографіка	Естетичне та читабельне відображення тексту	Оформлення заголовків статей і текстового вмісту
CSS-змінні	Гнучке налаштування стилів	Власна кольорова палітра блогу
JavaScript-компоненти	Інтерактивна поведінка елементів	Модальні вікна та динамічні форми
Кросбраузерність	Гарантована підтримка різних браузерів	Стабільний доступ із різних пристроїв студентів

відображення блогу на різних типах пристроїв без необхідності створення окремих версій сторінок (рис. 3).

Одним із ключових напрямів оптимізації блогівих платформ є робота зі стилями та статичними ресурсами. У процесі дослідження було встановлено, що надмірна кількість CSS-файлів і відсутність їх структуризації негативно впливають на час завантаження сторінок. Тому ефективними можна вважати передусім такі методи:

- мінімізація та об'єднання CSS-файлів;
- використання модульної структури стилів;
- уникнення дублювання правил оформлення.

Окрему увагу приділено кешуванню стилів, яке дозволяє повторно використовувати вже завантажені ресурси без додаткових мережевих запитів. Запровадження кешування CSS особливо ефективно для блогів, де структура сторінок є відносно сталою, а оновлення стилів відбувається

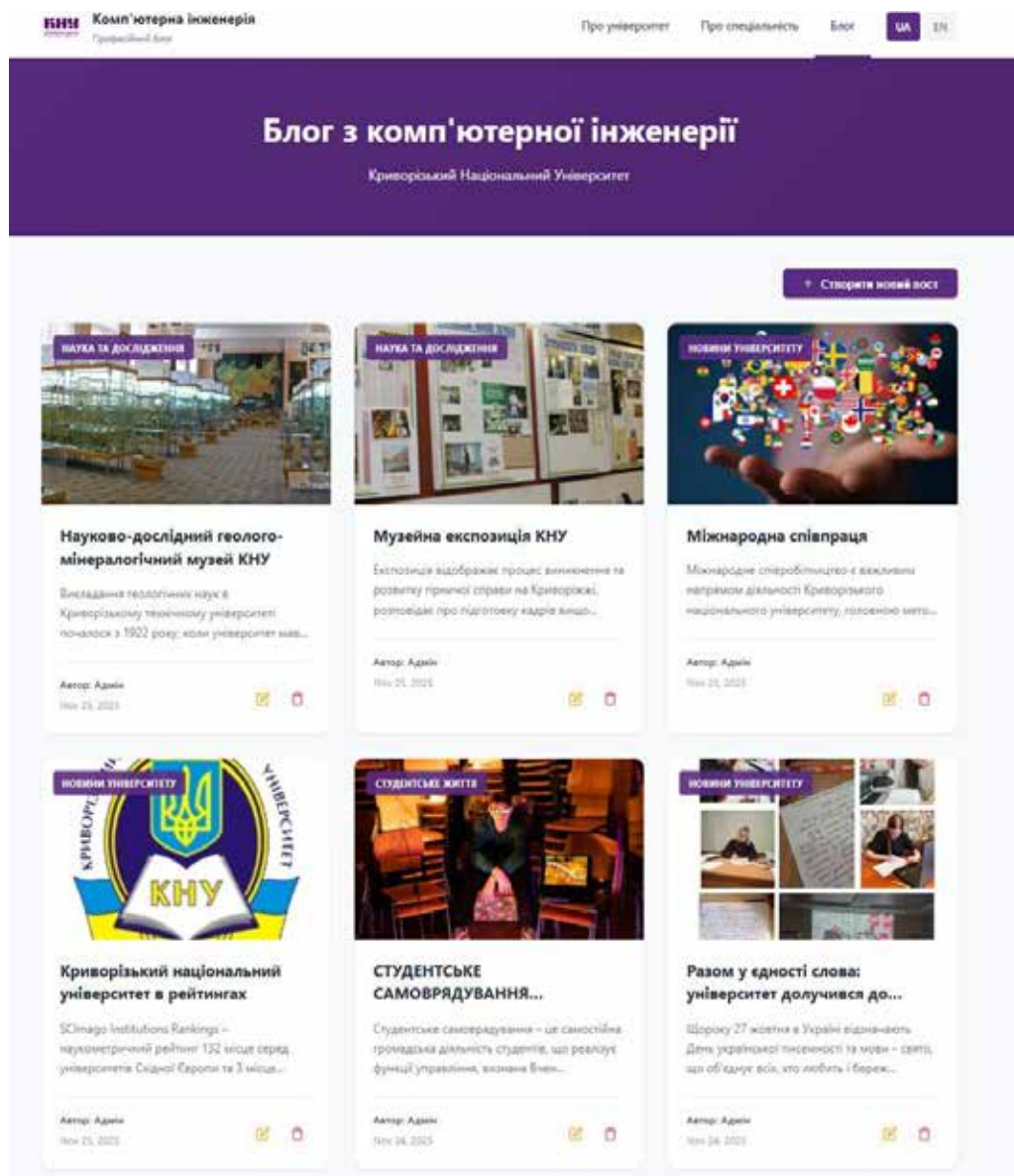


Рис. 3. Приклади реалізації адаптивного дизайну

нечасто. Деякі етапи оптимізації фронтенду представлено на рис. 4.

Оптимізація сучасного блогу не обмежується лише швидкодією. Важливими критеріями якості є адаптивність і доступність інтерфейсу. Значна частина користувачів споживає контент з мобільних пристроїв, тому інтерфейс повинен коректно відображатися на екранах різних розмірів. Bootstrap 5 і сучасні можливості CSS (Flexbox, Grid) забезпечують реалізацію адаптивних макетів без істотного ускладнення коду.

Доступність інтерфейсу передбачає врахування потреб користувачів з обмеженими можливостями. Доцільно наголосити на важливості дотримання принципів контрастності, коректного використання семантичних тегів і забезпечення

навігації з клавіатури. Це не лише розширює аудиторію блогу, але й позитивно впливає на загальну якість користувацького досвіду.

Практична реалізація інтерфейсу блогу, виконана в межах оновлення вебресурсів кафедри комп'ютерних систем та мереж, підтвердила доцільність комплексного підходу до frontend-оптимізації. Поєднання сучасних технологій розробки з методами оптимізації стилів, кешування ресурсів і адаптивного дизайну дозволило не лише скоротити час завантаження сторінок та підвищити стабільність роботи інтерфейсу, але й покращити зручність навігації та сприйняття контенту.

Узагальнюючи викладений матеріал, можна стверджувати, що оптимізація блогу на базі



Рис. 4. Етапи оптимізації фронкенду

сучасних frontend-технологій має розглядатися як системний процес, що поєднує проектні, технологічні та експлуатаційні аспекти. Саме така інтеграція забезпечує створення ефективних, конкурентоспроможних і зручних для користувача інформаційних блогів, здатних відповідати сучасним вимогам вебсередовища.

**Висновки.** У статті розглянуто сучасні методи оптимізації інформаційного блогу на

основі frontend-технологій та проаналізовано їх вплив на продуктивність і якість користувацького досвіду. Дослідження підтвердило, що ефективність блогівих платформ значною мірою залежить від оптимізації клієнтської частини, зокрема від раціональної організації стилів, коректного завантаження ресурсів і реалізації адаптивного дизайну.

У ході дослідження встановлено, що застосування сучасних інструментів frontend-розробки у поєднанні з методами кешування CSS, мінімізації та структурування стилів дозволяє суттєво скоротити час завантаження сторінок і зменшити навантаження на браузер користувача. Використання шаблонізатора Blade та фреймворка Bootstrap 5 для створення адаптивного й функціонального інтерфейсу блогу дозволяє забезпечити відповідність сучасним вимогам зручності та доступності.

Отримані результати свідчать, що комплексний підхід до оптимізації frontend-складової забезпечує підвищення стабільності роботи вебресурсу, покращення користувацького досвіду та конкурентоспроможності блогівих платформ. Запропоновані підходи можуть бути використані під час розробки нових та модернізації існуючих інформаційних блогів з високими вимогами до продуктивності й зручності використання.

#### Список літератури:

1. Google Search Central. Core Web Vitals. URL: <https://developers.google.com/search/docs/appearance/core-web-vitals> (дата звернення: 10.01.2026).
2. Elkins G. The Web Performance Handbook: A guide to web performance optimization core concepts and techniques. Kindle Edition. 2021. 61 p.
3. Супрун М. В., Маніта І. Ю. Оптимізація сайту як спосіб підвищення його ефективності. *Українські студії в європейському контексті*. 2024. № 8. С. 290–295. URL: <http://elar.tsatu.edu.ua/handle/123456789/18450>
4. Web.dev. Extract Critical CSS. URL: <https://web.dev/articles/extract-critical-css> (дата звернення: 10.01.2026).
5. Google Developers. Optimize CSS Delivery URL: <https://developers.google.com/speed/docs/insights/OptimizeCSSDelivery> (дата звернення: 10.01.2026).
6. Mozilla Developer Network. Cache-Control – HTTP header. URL: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Reference/Headers/Cache-Control> (дата звернення: 10.01.2026).
7. Mozilla Developer Network. Caching HTTP. URL: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Caching> (дата звернення: 10.01.2026).
8. W3C. Web Content Accessibility Guidelines (WCAG) 2.2. URL: <https://www.w3.org/TR/WCAG22> (дата звернення: 10.01.2026).
9. Enge, E., Spencer, S., Stricchiola, J. The Art of SEO: Mastering Search Engine Optimization. 4th ed. Sebastopol: O'Reilly Media, 2023. 773 p.

#### **Kvitka D.I., Muzyka I.O., Kuznetsov D.I. METHODS OF BLOG OPTIMIZATION BASED ON MODERN FRONTEND DEVELOPMENT TECHNOLOGIES**

*The article considers methods for optimizing an information blog based on modern frontend development technologies in order to improve performance, usability, and the overall quality of user experience. The relevance of the study is determined by the rapid evolution of web technologies and the growing requirements for speed, adaptability, and accessibility of web interfaces. The object of the research is the process of developing*

*and optimizing the client-side of web applications, while the subject is the methods and tools for optimizing a blog interface using modern frontend technologies. Authors have analyzed the main stages of frontend development, principles of interface design, and approaches to building the information architecture of blog platforms. Special attention is paid to the use of the Blade templating engine and the Bootstrap 5 framework for creating an adaptive interface, as well as to methods of optimizing styles and resources. Practical aspects of applying CSS caching, minimization and structuring of styles, optimization of static resource loading, and implementation of responsive design are examined. It is shown that the integrated application of these methods makes it possible to reduce page load time, decrease browser workload, and improve the stability of the web interface. The research employs methods of analysis and generalization of scientific sources, comparative analysis of frontend technologies, design and engineering approaches, and practical implementation of a blog interface. The obtained results confirm the feasibility of using modern frontend tools in combination with optimization methods to create efficient information blogs. The practical significance of the study lies in the possibility of applying the proposed approaches in the development and modernization of web resources focused on content publication and distribution under conditions of high requirements for performance and ease of use.*

**Keywords:** *frontend development, web interface, optimization, blog, responsive design, style caching, Blade, Bootstrap 5.*

Дата першого надходження статті до видання: 11.01.2026

Дата прийняття статті до друку після рецензування: 10.02.2026

Дата публікації (оприлюднення) статті: 08.04.2026