

Кошова О. П., кандидат педагогічних наук, доцент кафедри комп'ютерних наук та інформаційних технологій
Полтавського університету економіки і торгівлі
ORCID: 0000-0003-0794-6774

Черненко О. О., кандидат фізико-математичних наук, доцент кафедри комп'ютерних наук та інформаційних технологій
Полтавського університету економіки і торгівлі
ORCID: 0000-0002-9084-0999

Комар І. І., магістр спеціальності «Комп'ютерні науки»
Полтавського університету економіки і торгівлі
ORCID: 0009-0009-3140-268X

РОЗРОБКА ПЛАТФОРМИ ДЛЯ ВЕДЕННЯ БЛОГУ ВИКЛАДАЧА

Метою статті є аналіз архітектури програмного забезпечення для блогу викладача та опис його основних компонентів, які забезпечують ефективне управління контентом та швидку взаємодію з користувачами. Висвітлюються підходи до розробки адміністративного вебдодатку та публічного вебсайту, їх взаємодія через API, а також використання сучасних технологій, таких як Next.js, MongoDB та JWT, для досягнення стабільності, масштабованості та безпеки системи.

Архітектура програмного забезпечення складається з адміністративного вебдодатку (Frontend і Backend) та публічного вебсайту (Frontend). Адміністративний інтерфейс створений на платформі Next.js, що поєднує серверний рендеринг (SSR) та клієнтський рендеринг, забезпечуючи високу швидкість завантаження сторінок. Frontend частина дозволяє викладачам керувати контентом блогу, включаючи створення, редагування та видалення статей, книг та інформації про автора. Backend частина реалізована для обробки запитів, управління базою даних MongoDB, аутентифікації та перевірки прав доступу через JWT. Публічний вебсайт використовує технології статичної генерації сторінок (SSG) та SSR, що забезпечує оптимальну продуктивність та швидке завантаження контенту. Також розглянуто сценарії використання, включаючи авторизацію адміністратора, керування статтями, книгами та налаштуваннями. Для хостингу вибрано платформу Vercel, яка забезпечує автоматичне розгортання через GitHub та підтримку CI/CD.

Результати дослідження підтверджують ефективність вибраної архітектури для забезпечення швидкої та безпечної роботи блогу викладача. Використання Next.js, MongoDB та JWT дозволяє створити гнучку та масштабовану систему з можливістю розширення функціоналу. Публічний вебсайт надає користувачам доступ до статей і книг без аутентифікації, а адміністративний вебдодаток забезпечує зручне управління контентом. Обрані технології сприяють високій продуктивності та зручності використання системи.

Ключові слова: платформа, блог, викладач, освітній процес.

Koshova O. P., Chernenko O. O., Komar I. I. Developing a platform for teacher blogging

The purpose of the article is to analyze the software architecture for a teacher's blog and describe its main components that ensure effective content management and quick interaction with users. It covers approaches to developing an administrative web application and a public website, their interaction via API, as well as the use of modern technologies such as Next.js, MongoDB, and JWT to achieve system stability, scalability, and security.

The software architecture consists of an administrative web application (Frontend and Backend) and a public website (Frontend). The administrative interface is built on the Next.js platform, which combines server-side rendering (SSR) and client-side rendering, ensuring high page loading speed. The frontend part allows teachers to manage blog content, including creating, editing, and deleting articles, books, and author information. The backend part is implemented for query processing, MongoDB database management, authentication, and access control via JWT. The public website uses static page generation (SSG) and SSR technologies, which ensure optimal performance and fast loading of content. Also considered are use cases, including administrator authorization, article, book and settings management. The Vercel platform was chosen for hosting, which provides automatic deployment via GitHub and CI/CD support.

The results of the study confirm the effectiveness of the chosen architecture for ensuring fast and secure operation of the teacher's blog. The use of Next.js, MongoDB, and JWT allows you to create a flexible and scalable system with the ability to expand functionality. The public website provides users with access to articles and books without authentication, and the administrative web application provides convenient content management. The selected technologies contribute to high productivity and ease of use of the system.

Key words: platform, blog, teacher, educational process.

Постановка проблеми. У сучасній системі освіти зростає значення цифрових технологій, що сприяють кращій передачі знань і налагодженню контакту між викладачем та студентом. Важливу роль у цьому процесі відіграють персональні блоги викладачів. Вони дають змогу публікувати навчальні матеріали, ділитися досвідом, поширювати наукові доробки й комунікувати зі студентами. Але створення та підтримка такого блогу вимагає розробки ефективної архітектури ПЗ, яка забезпечить зручність керування, захист доступу, високу швидкість та масштабованість. Брак комплексних рішень із швидким завантаженням, простим редагуванням і захистом даних підкреслює необхідність створення спеціалізованої системи для потреб викладачів.

Аналіз останніх досліджень та публікацій. Сучасні дослідження [1–11] вказують на популяризацію персоналізованих блогів викладачів як засобу дистанційного навчання та обміну знаннями. Наукові праці наголошують на потребі в безпеці, швидкодії та зручному адмініструванні. Наприклад, у [3] розглядається використання соціальних медіа для постійної комунікації у вищій школі, а в [4] – їх вплив на освітній процес.

Метою роботи є розробка платформи для ведення блогу викладача.

Виклад основного матеріалу. Архітектура програмного забезпечення, яка лежить в основі функціонування блогу викладача, складається з двох ключових складових: адміністративного вебдодатку та публічного вебсайту. Адміністративний вебдодаток, у свою чергу, поділяється на дві частини – клієнтську (Frontend) та серверну (Backend). Публічний вебсайт є вітриною для відображення навчального та наукового контенту. Всі ці компоненти інтегруються між собою через чітко структуровані інтерфейси прикладного програмування (API), що забезпечує надійність, узгодженість у роботі та легкість масштабування платформи.

Адміністративна частина (інтерфейс) реалізована з використанням сучасної фреймворк-платформи Next.js, яка поєднує у собі переваги серверного рендерингу (SSR) та клієнтського рендерингу. Такий підхід забезпечує високу швидкість завантаження сторінок навіть при великій кількості контенту. Інтерфейс адміністратора розділений на:

Frontend (клієнтська частина) – дозволяє викладачу керувати вмістом блогу. До функціональності належать: створення нових статей, оновлення та редагування вже опублікованих, їхнє видалення, а також перегляд статистики відвідування, дій користувачів тощо. Окрім того, реалізовано можливість керування списком книг: додавання нових видань, редагування інформації про них та демонстрація цих даних на публічному сайті.

Завдяки можливостям Next.js усі зміни контенту оперативнo оновлюються і миттєво відображаються користувачу завдяки серверному рендерингу.

Backend (серверна частина) – відповідає за обробку всіх запитів, що надходять як з адміністративного інтерфейсу, так і з публічного сайту. Тут реалізовано систему аутентифікації користувачів, контроль доступу до окремих функцій, взаємодію з базою даних. Усі операції з додавання, редагування або видалення контенту, таких як статті, книги чи інформація про автора, передаються через API до серверної частини, яка в свою чергу зберігає дані у базі MongoDB.

Публічний вебсайт, що також базується на Next.js, слугує зовнішньою частиною блогу, доступною для всіх користувачів. Він надає змогу переглядати всі опубліковані статті, книги, а також біографічну інформацію про викладача. Платформа підтримує статичну генерацію сторінок (SSG) поряд із серверним рендерингом, що дозволяє заздалегідь формувати сторінки та зберігати їх у кеші. Такий підхід суттєво пришвидшує їх завантаження, навіть при великій кількості одночасних відвідувачів.

MongoDB використовується як основне сховище для всіх даних блогу. Це документно-орієнтована база даних, яка дозволяє гнучко зберігати інформацію у вигляді документів різного формату. Вона чудово підходить для роботи з динамічним контентом, що змінюється з часом, і легко масштабується, дозволяючи збільшити обсяг даних без втрати продуктивності.

Система автентифікації побудована на основі технології JWT (JSON Web Token). Вона забезпечує захист адміністративної частини платформи, дозволяючи доступ до редагування контенту лише авторизованим користувачам. Такий підхід гарантує безпеку облікових даних і зменшує ризики несанкціонованого доступу (рис. 1).

Для опису роботи платформи для ведення блогу побудовано діаграму прецедентів (рис. 2).

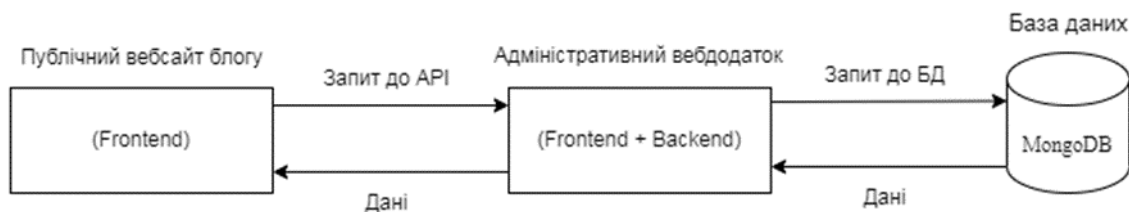


Рис. 1. Архітектура програмного забезпечення



Рис. 2. Діаграма прецедентів

Прецедент: Авторизація

Учасник: адміністратор блогу.

Початковий стан: адміністратор ще не увійшов у систему.

Кінцевий стан: адміністратор успішно авторизований.

Опис дій:

Адміністратор вводить свій логін (email) і пароль.

Натискає кнопку «Увійти».

Система перевіряє правильність введених даних.

Після успішної перевірки адміністратор отримує доступ до системи.

Прецедент: Управління статтями

Учасник: адміністратор блогу.

Початковий стан: адміністратор перебуває в системі.

Кінцевий стан: дані про статті оновлено – додано, змінено або видалено.

Опис дій:

Адміністратор відкриває розділ «Статті».

На екрані з'являється перелік доступних публікацій.

Вибирає потрібну статтю та виконує необхідну дію: додає нову, редагує або видаляє.
Дії обробляються відповідним функціоналом системи.

Прецедент: Управління книгами

Учасник: адміністратор блогу.

Початковий стан: адміністратор авторизований.

Кінцевий стан: відомості про книги оновлено.

Опис дій:

Адміністратор переходить до розділу «Книги».

Виводиться список наявних публікацій.

Обирає конкретну книгу для редагування, додає нову або видаляє наявну.

Система фіксує внесені зміни.

Прецедент: Налаштування профілю

Учасник: адміністратор блогу.

Початковий стан: авторизований адміністратор.

Кінцевий стан: особисті дані викладача оновлено.

Опис дій:

Переходить до розділу «Налаштування».

Переглядає доступні параметри зміни профілю.

Редагує власні персональні дані або додає додаткову інформацію про себе.

Система зберігає зміни до профілю.

Прецедент: Перегляд статті

Учасник: користувач.

Початковий стан: користувач знаходиться на сайті блогу.

Кінцевий стан: стаття переглянута.

Опис дій:

Користувач переходить до розділу «Статті».

Відображається перелік усіх публікацій.

Обирає потрібну статтю.

Ознайомлюється з її змістом.

Прецедент: Перегляд книги

Учасник: користувач.

Початковий стан: користувач відвідує сайт блогу.

Кінцевий стан: переглянуто інформацію про вибрану книгу.

Опис дій:

Користувач відкриває розділ «Книги».

Перед ним з'являється список доступних видань.

Обирає необхідну книгу.

Переглядає її зміст або опис.

Прецедент: Перегляд інформації про автора

Учасник: користувач.

Початковий стан: відкрито сторінку блогу.

Кінцевий стан: користувач ознайомився з інформацією про автора.

Опис дій:

Користувач натискає на вкладку «Про автора».

На екрані відображається біографічна інформація.

Користувач переглядає вміст сторінки.

Для розгортання проекту обрано платформу Vercel, оскільки вона забезпечує зручний та швидкий процес розгортання проекту. Її перевагами є автоматизація, підтримка CI/CD, простота інтеграції з репозиторіями та можливість налаштування змінних середовища.

Спочатку підготовлено проєкт до розгортання. Для цього встановлено всі залежності за допомогою «npm install», а також перевірено локальну роботу програми. Особливу увагу потрібно приділили файлу package.json, у якому потрібно вказати скрипт для збірки «build»: «next build».

Наступним кроком було створення акаунту на платформі Vercel та інтеграція з репозиторієм проєкту через GitHub. Процес розгортання виконується автоматично після вибору репозиторію та натискання кнопки «Deploy». Платформа виконала збірку проєкту, а після успішного завершення надано посилання на готовий сайт. У разі виявлення помилок у роботі сайту їх виправлення у вихідному коді та збереження змін у репозиторію спричиняє автоматичний запуск повторного розгортання.

Щоб вказати свій домен на Vercel, необхідно перейти до налаштувань проєкту в панелі керування Vercel. У розділі «Domains» потрібно додати домен: «https://teacher-blog.vercel.app/». Проєкт готовий до використання.

Висновки з дослідження і перспективи подальших розвідок у цьому напрямі. Розроблена платформа базується на передових підходах до проектування програмного забезпечення, включаючи використання компонентної архітектури та модульного підходу до побудови інтерфейсу. Це забезпечило високу продуктивність, масштабованість та можливість подальшого розширення функціоналу.

Подальші дослідження можуть бути спрямовані на інтеграцію штучного інтелекту для персоналізації контенту, удосконалення механізмів безпеки та впровадження аналітичних інструментів для покращення взаємодії користувачів із платформою. Отримані результати можуть бути корисними для розробників освітніх вебрішень, а також для викладачів, які прагнуть підвищити якість цифрового навчального середовища.

Список використаних джерел:

1. Гуз К. М. Проектування і розробка персонального блогу викладача як засобу активізації пізнавальної діяльності учнів. Волинський національний університет імені Лесі Українки. 2023. Режим доступу: <https://evnuir.vnu.edu.ua/handle/123456789/26961>.
2. Мартинюк О. В. Освітній блог як засіб професійної підготовки майбутніх вихователів закладів дошкільної освіти. Інформаційні технології і засоби навчання. 2022. Т. 87, № 3. С. 54-67. Режим доступу: <https://journal.iitta.gov.ua/index.php/itlt/article/view/2425/1443>.
3. Kumar, S., Nanda, P. Social Media in Higher Education: A Framework for Continuous Engagement. *Journal of Educational Technology & Society*, 2020, 23(1), pp. 33-44.
4. Manca, S., Ranieri, M. Implications of social network sites for teaching and learning. Where we are and where we want to go. *Education and Information Technologies*, 2020, 22(2), pp. 605-622.
5. Василенко М., Горбаченко С., Слатвінська В., Чепурна О. Сценарне планування як інструмент безперервного бізнесу: WEB-технології (комплексний підхід) Інформаційні технології та суспільство. 2023. № 3. Режим доступу: <https://journals.maup.com.ua/index.php/it/article/view/2098>.
6. Слабінога М. О., Чабан С. В. Розробка веб-додатків в контексті оптимізації їх швидкодії. Таврійський науковий вісник. Серія: Технічні науки. 2022. № 3. С. 45–52. DOI: 10.32851/tnv-tech.2022.3.7. Режим доступу: <https://journals.ksauniv.ks.ua/index.php/tech/article/view/219>.
7. Антоненко А. В., Балвак А. А., Цвик О. С., Ємелін Д. М., Гришковець Є. П. Інноваційні методи відображення інформації в веб-браузерах. Таврійський науковий вісник. Серія: Технічні науки. 2024. № 1. С. 81–88. DOI: 10.32782/tnv-tech.2024.1.9. Режим доступу: <https://journals.ksauniv.ks.ua/index.php/tech/article/view/542>.
8. Д. М. Ольховський, С.В. Гаркуша, О. О. Черненко, В.О. Лазаренко Розробка сайту кафедри навчально-наукового інституту денної освіти. Збірник наукових праць національного університету кораблебудування імені адмірала Макарова, № 1 (494), 2024, С. 119-125.
9. Олексійчук Ю. Ф., Ольховський Д.М., Ольховська О.В., Андрушків О. М., Проектування, розробка та тестування web-сервісу для вибору тем дипломних робіт. Системи та технології, № 1, № 1, с. 43-50. <https://doi.org/10.32782/2521-6643-2024-1-67.7>
10. О.П. Кошова, О.О. Черненко, Т.В. Чілікіна, І.І. Комар Особливості розробки web-застосунків для системи дистанційного навчання з допомогою бібліотеки React. Системи та технології, 2023, 65(1), С. 20-31. <https://doi.org/10.32782/2521-6643-2023.1-65.3>
11. Кошова О. П., Ольховська О.В., Тацій Д. С., Олексійчук Ю.Ф., Черненко О.О. РОЗРОБКА ВЕБ-ДОДАТКІВ ТА СЕРВІСІВ НА ПЛАТФОРМІ NODE.JS Таврійський науковий вісник. Серія: Технічні науки, 2023. Вип. 2. С. 78-89. <https://doi.org/10.32782/tnv-tech.2023.2.9>

References:

1. Huz K. M. (2023) Proektuvannia i rozrobka personalnoho blohu vykladacha yak zasobu aktyvizatsii piznavalnoi diialnosti uchniv [Design and development of a teacher's personal blog as a means of activating students' cognitive activity]. *Volynskyi natsionalnyi universytet imeni Lesi Ukrainky. Rezhym dostupu: https://evnuir.vnu.edu.ua/handle/123456789/26961* [in Ukrainian]
2. Martyniuk O. V. (2022) Osvitnii bloh yak zasib profesiinoi pidhotovky maibutnikh vykhovateliv zakladiv doshkilnoi osvity [Educational blog as a means of professional training of future educators of preschool educational institutions]. *Informatsiini tekhnolohii i zasoby navchannia* [Information technologies and teaching aids] T. 87, № 3. S. 54-67. Rezhym dostupu: <https://journal.iitta.gov.ua/index.php/itlt/article/view/2425/1443> [in Ukrainian]
3. Kumar, S., Nanda, P. (2020) Social Media in Higher Education: A Framework for Continuous Engagement. *Journal of Educational Technology & Society*, 23(1), pp. 33-44.
4. Manca, S., Ranieri, M. (2020) Implications of social network sites for teaching and learning. Where we are and where we want to go. *Education and Information Technologies*, 22(2), pp. 605-622.
5. Vasylenko M., Horbachenko S., Slatvinska V., Chepurna O. (2023) Stsenarne planuvannia yak instrument bezperervnoho biznesu: WEB-tekhnologii (kompleksnyi pidkhid) [Scenario planning as a tool for continuous business: WEB technologies (comprehensive approach)] *Informatsiini tekhnolohii ta suspilstvo*. [Information

Technologies and Society]. № 3. Rezhym dostupu: <https://journals.maup.com.ua/index.php/it/article/view/2098> [in Ukrainian]

6. Slabinoha M. O., Chaban S. V. (2022) Rozrobka veb-dodatkov v konteksti optymizatsii yikh shvydkodii [Development of web applications in the context of optimizing their performance]. Tavriyskyi naukovi visnyk. Serii: Tekhnichni nauky. [Tavria Scientific Bulletin. Series: Technical Sciences.] № 3. S. 45–52. DOI: 10.32851/tnv-tech.2022.3.7. Rezhym dostupu: <https://journals.ksauniv.ks.ua/index.php/tech/article/view/219> [in Ukrainian]

7. Antonenko A. V., Balvak A. A., Tsvyk O. S., Yemelin D. M., Hryshkovets Ye. P. (2024) Innovatsiini metody vidobrazhennia informatsii v veb-brauzerakh [Innovative methods of displaying information in web browsers]. Tavriyskyi naukovi visnyk. Serii: Tekhnichni nauky [Tavria Scientific Bulletin. Series: Technical Sciences]. № 1. S. 81–88. DOI: 10.32782/tnv-tech.2024.1.9. Rezhym dostupu: <https://journals.ksauniv.ks.ua/index.php/tech/article/view/542> [in Ukrainian]

8. D. M. Olkhovskiy, S.V. Harkusha, O. O. Chernenko, V.O. (2024) Lazarenko Rozrobka сайту кафедри навчально-наукового інституту денної освіти [Development of the website of the Department of the Educational and Scientific Institute of Full-time Education]. Zbirnyk naukovykh prats natsionalnoho universytetu korablebuduvannia imeni admiral Makarova [Collection of Scientific Papers of the Admiral Makarov National Shipbuilding University], № 1 (494), S. 119-125. [in Ukrainian]

9. Oleksiichuk Yu. F., Olkhovskiy D.M., Olkhovska O.V., Andrushkiv O. M. (2024). Proiektuvannia, rozrobka ta testuvannia web-servisu dlia vyboru tem dyplomnykh robit [Design, development and testing of a web-service for selecting thesis topics]. Systemy ta tekhnolohii [Systems and Technologies], № 1, № 1, s. 43-50. <https://doi.org/10.32782/2521-6643-2024-1-67.7> [in Ukrainian]

10. O.P. Koshova, O.O. Chernenko, T.V. Chilikina, I.I. Komar (2023) Osoblyvosti rozrobky web-zastosunkiv dlia systemy dystantsiinoho navchannia z dopomohoiu biblioteki React [Features of developing web applications for a distance learning system using the React library]. Systemy ta tekhnolohii [Systems and Technologies], 65(1), S.20-31. <https://doi.org/10.32782/2521-6643-2023.1-65.3> [in Ukrainian]

11. Koshova O. P., Olkhovska O.V., Tatsii D. S., Oleksiichuk Yu.F., Chernenko O.O. (2023) ROZROBKA VEB-DODATKIV TA SERVISIV NA PLATFORMI NODE.JS [DEVELOPMENT OF WEB APPLICATIONS AND SERVICES ON THE NODE.JS PLATFORM] Tavriyskyi naukovi visnyk. Serii: Tekhnichni nauky [Tavria Scientific Bulletin. Series: Technical Sciences]. Vyp. 2. S. 78-89. <https://doi.org/10.32782/tnv-tech.2023.2.9> [in Ukrainian]