

Міністерство освіти і науки України



NURE

Харківський національний університет
радіоелектроніки

ЗБІРНИК

студентських наукових статей

«Автоматизація та приладобудування»

«Automation and Development of Electronic Devices»

ADED-2026

(Випуск 1)

[електронне видання]



<https://tapr.nure.ua/>



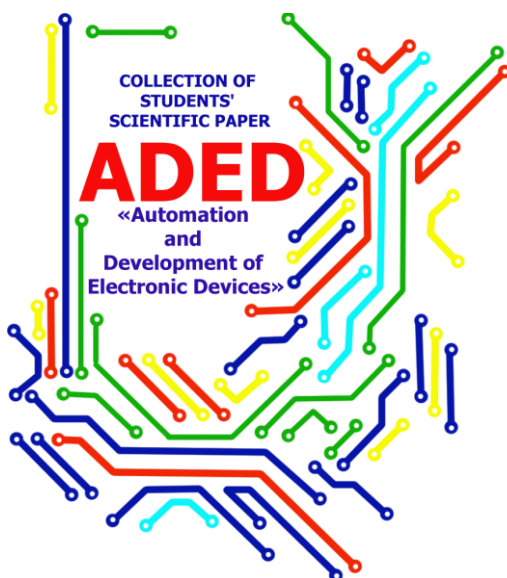
<http://itez.zntu.edu.ua/>



<http://kafea.kdu.edu.ua>

Харків 2026

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
Кафедра комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та
безпекової інженерії (КІТАРБІ)



ЗБІРНИК

студентських наукових статей

«Автоматизація та приладобудування»

«Automation and Development of Electronic Devices»

ADED-2026

(Випуск 1)

[електронне видання]

Харків 2026

Головий редактор **Невлюдов Ігор Шакирович**, доктор технічних наук, професор, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії, Харківського національного університету радіоелектроніки.

Редакційна колегія: **Филипенко Олександр Іванович**, доктор технічних наук, професор, декан факультету автоматизації, комп'ютерно-інтегрованих технологій та систем, Харківського національного університету радіоелектроніки.

Цимбал Олександр Михайлович, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії, Харківського національного університету радіоелектроніки.

Андрусевич Анатолій Олександрович, доктор технічних наук, професор, начальник Криворізького коледжу національного авіаційного університету

Косенко Віктор Васильович, доктор технічних наук, професор, зам. директора Державного підприємства «Південний державний проектно-конструкторський та науково-дослідний інститут авіаційної промисловості».

Замірець Микола Васильович, доктор технічних наук, професор, директор Державного підприємства Науково-дослідного технологічного інституту приладобудування.

Свищ Володимир Митрофанович, доктор технічних наук, професор, радник директора Державне науково-виробниче підприємство «Об'єднання Комунар».

Фомовська Олена Владиславівна, кандидат технічних наук, доцент завідувач кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського.

Кухаренко Дмитро Володимирович, кандидат технічних наук, доцент кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського

Демська Наталія Павлівна, кандидат технічних наук, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії, Харківського національного університету радіоелектроніки.

Фурманова Наталія Іванівна, кандидат технічних наук, доцент, декан факультету інформаційної безпеки та електронних комунікацій, Національного університету «Запорізька політехніка».

Відповідальний редактор: **Євсєєв Владислав В'ячеславович**, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії, Харківського національного університету радіоелектроніки.

Автоматизація та Приладобудування («Automation and Development of Electronic Devices» ADED-2026) [Електронний ресурс] : збірник студентських наукових статей / Харківський національний університет радіоелектроніки ; [редкол.: І.Ш. Невлюдов та ін.]. – Харків : ХНУРЕ, 2026. – Вип. 1. – 237с.

Collection of Students' Scientific Paper «Automation and Development Of Electronic Devices» ADED-2026 Part 1 (Key infrastructure 2026) - Kharkiv/ The Editorial.: Nevlyudov I.Sh. (head), that all. Kharkiv: Kind of Kharkiv National University of Radio Electronics [electronic edition], 2026. – 237p with.

Рекомендовано рішенням
Науково-технічної ради
Харківського національного
університету радіоелектроніки
протокол №6 від 29.11.2018

Рекомендовано рішенням Вченої ради
факультету Автоматизації, комп'ютерно-інтегрованих
технологій та систем
Харківського національного
університету радіоелектроніки
протокол № 3 від 20 травня 2026

Збірник містить наукові статті здобувачів першого (бакалаврського), другого (магістерського) рівнів вищої освіти кафедри комп'ютерно-інтегрованих технологій, автоматизації, робототехніки та безпекової інженерії (КІТАРБІ) Харківського національного університету радіоелектроніки, кафедри Інформаційних технологій електронних засобів (ІТЕД) Запорізького національного технічного університету та кафедри Електронних апаратів (ЕА) Кременчуцького національного університету ім. М. Остроградського які навчаються за спеціальностями: G7 (151, 174) Автоматизація, комп'ютерно-інтегровані технології та робототехніка; 172 Телекомунікації та радіотехніка, 171 Електроніка та 163 Біомедична інженерія. Статті надані в авторській редакції.

©ХНУРЕ, 2026 рік

ЗМІСТ

<i>В.О. Александров</i>	
Безпілотні морські апарати. Розрахунок та технічні особливості	8
<i>І.В. Ганькін</i>	
Адаптивне управління вентиляційною системою фарбувального цеху	14
<i>П.Д. Дон</i>	
Розробка модуля 5-ї осі для 3D-принтера	20
<i>С.В. Єрмоменко</i>	
Датчики для автоматизованого управління технологічними процесами в агресивних середовищах, шкідливих та небезпечних умовах праці	26
<i>С.О. Запорожченко</i>	
Підвищення ефективності системи автоматизованого обліку електроенергії за рахунок введення функції прогнозування	32
<i>О.В. Корнієнко</i>	
Аналіз сучасних моделей нейромережей для розпізнавання рукописного тексту	37
<i>О.В. Корнієнко</i>	
Аналіз сучасних систем контролю подачі електроенергії для підприємства	40
<i>Volodymyr Kotenko</i>	
Analysis of the Current State, Problems and Development Trends of Computer Vision in Unmanned Surface Vehicle Systems	43
<i>О.В. Кремзюков</i>	
Підвищення стійкості АСУТП до кіберзагроз засобами автоматизованого резервного копіювання	50
<i>С.В. Мельніков</i>	
Проблеми визначення відповідальності в системах автономної робототехніки	57
<i>С.В. Мельніков</i>	
Архітектура Веб-додатків: фронтенд та бекенд як складові інформаційної системи	63
<i>В.О. Ольховський</i>	
Хмарні технології як інструмент ресурсозбереження в автоматизованих системах	70
<i>Н.В. Панченко</i>	
Огляд технології reflective DTF-друку та її застосування	75
<i>С.С. Скриннік</i>	
Розроблення демонстраційного навчального стенду «Промисловий конвеєр»	81
<i>В.В. Столяренко</i>	
Програмний засіб для створення електропневматичних схем	87
<i>Д.С. Хабаров</i>	
Аналіз сучасних ШІ: ЧАТ GPT, DEEPSEEK, GEMINI	94
<i>М.С. Чмига</i>	
Роль штучного інтелекту в оптимізації інформаційно-пошукових систем	101
<i>О.В. Шевченко</i>	
Вентиляція захисних споруд цивільного захисту: Технологічні аспекти та вимоги безпеки	106
<i>О.В. Шевченко</i>	
Порівняльний аналіз типів вентиляційних систем: Від природної до інтелектуальної (DCV)	113

АРХІТЕКТУРА ВЕБ-ДОДАТКІВ: ФРОНТЕНД ТА БЕКЕНД ЯК СКЛАДОВІ ІНФОРМАЦІЙНОЇ СИСТЕМИ

С.В. Мельніков

Харківський національний університет радіоелектроніки

Україна, 61166, Харків, пр. Науки 14

E-mail: stanislav.melnikov@nure.ua

Анотація: У роботі розглянуто фундаментальні поняття сучасної веб-розробки: фронтенд (клієнтська сторона) та бекенд (серверна сторона). За допомогою практичних аналогій пояснено принципи взаємодії цих складових та роль API (інтерфейсу програмування застосунків) у передачі даних. Охарактеризовано базовий стек технологій фронтенду (HTML, CSS, JavaScript) та сучасні фреймворки (React, Vue.js, Angular). Визначено основні компоненти архітектури бекенду (сервери, бази даних, мови програмування), а також висвітлено питання безпеки, продуктивності та типів архітектур (монолітна, мікросервісна, serverless). Робота спрямована на формування цілісного розуміння технічної будови веб-ресурсів та їх функціонування як єдиної інформаційної системи.

Ключові слова: веб-розробка, фронтенд, бекенд, api, клієнт-серверна архітектура.

WEB APPLICATION ARCHITECTURE: FRONTEND AND BACKEND AS COMPONENTS OF AN INFORMATION SYSTEM

S. V. Melnikov

Kharkiv Kharkiv National University of Radio Electronics

Ukraine, 61166, Kharkiv, Nauky av, 14

E-mail: stanislav.melnikov@nure.ua

Abstract: The paper examines the fundamental concepts of modern web development: frontend (client-side) and backend (server-side). Using practical analogies, the principles of interaction between these components and the role of API (Application Programming Interface) in data transmission are explained. The basic frontend technology stack (HTML, CSS, JavaScript) and modern frameworks (React, Vue.js, Angular) are characterized. The main components of backend architecture (servers, databases, programming languages) are defined, as well as issues of security, performance, and types of architectures (monolithic, microservices, serverless). The work aims to form a holistic understanding of the technical structure of web resources and their functioning as a unified information system.

Key words: web development, frontend, backend, api, client-server architecture.

Сучасна епоха – це епоха цифрових технологій [1-7]. Щодня мільярди людей відкривають браузер, щоб перевірити пошту, подивитися відео на YouTube або замовити товари в інтернет-магазинах. Для звичайного користувача сайт – це просто картинка на екрані, яка реагує на натискання. Але з технічної точки зору будь-який веб-ресурс – це складна система, що нагадує айсберг. Те, що видно на екрані, – це лише вершина айсберга, яка називається Frontend (Фронтенд). А те, що приховано під водою і забезпечує роботу всієї системи, – це Backend (Бекенд). Розуміння різниці між цими поняттями є першим і найважливішим кроком для будь-кого, хто хоче зрозуміти, як працює сучасний інтернет. Мета цього реферату – доступно пояснити сутність фронтенду та бекенду, розглянути їхні інструменти та принципи взаємодії.

Сьогодні веб стає всюдисущим [8-11]. Смартфони, розумні колонки, автівки, холодильники та навіть міські світлофори під'єднані до інтернету. За кожною такою взаємодією стоїть та сама архітектура: фронтенд (те, що бачить користувач) і бекенд (те, що обчислює дані). Розуміння того, як вони пов'язані, дозволяє не просто користуватися технологіями, а бачити їхню логіку – від першого кліку до отримання результату.

Знання принципів роботи фронтенду та бекенду стає базовою цифровою грамотністю. Це допомагає ставити правильні завдання розробникам, оцінювати реалістичність функцій сайту або застосунку та розуміти, чому одні сервіси працюють миттєво, а інші – зависають. Тому тема є актуальною не лише для технічних фахівців, а для всіх, хто взаємодіє з цифровим світом.

Frontend – це сукупність програмних та візуальних компонентів веб-ресурсу, які виконуються на стороні пристрою користувача (клієнта) у браузері. Іншими словами, це все те, що користувач безпосередньо бачить і з чим взаємодіє: кнопки, форми, зображення, анімація, меню, тексти тощо.

Frontend вирішує три основні завдання:

- візуальне представлення даних (як саме виглядає інформація);
- забезпечення інтерактивності (реакція на дії користувача: кліки, прокручування, введення тексту);
- комунікація з Backend (відправлення запитів та отримання відповідей) (рис. 1).

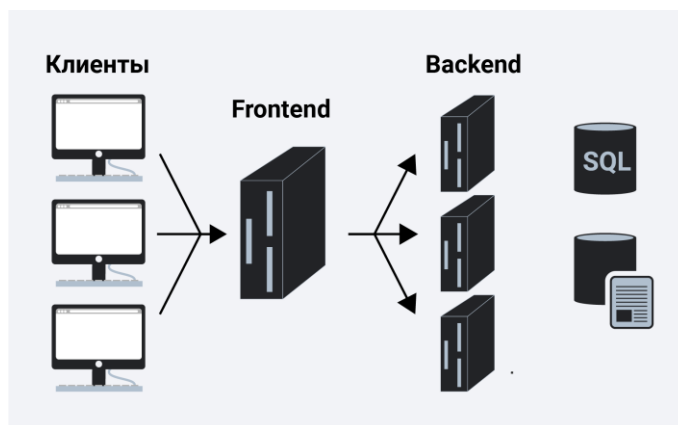


Рисунок 1 – Графічне представлення взаємодії бекенду та фронтенду

Весь видимий веб-простір ґрунтується на трьох фундаментальних технологіях, кожна з яких виконує строго визначену функцію.

HTML (HyperText Markup Language) – мова гіпертекстової розмітки. Дана технологія відповідає за структурну організацію веб-сторінки, але не за її візуальне оформлення чи логіку. HTML інтерпретується браузером як набір елементів, що позначають заголовки, абзаци, зображення, гіперпосилання, кнопки та інші складові інтерфейсу. Аналогія: скелет людини або несучий каркас будівлі (цегляні стіни без оздоблення).

CSS (Cascading Style Sheets) – каскадні таблиці стилів. Ця технологія забезпечує візуальне представлення HTML-елементів. До її функцій належать: визначення кольорової гами, гарнітур шрифтів, розмірів, відступів, позиціонування об'єктів на сторінці, а також адаптивне відображення контенту під різні роздільна здатність екранів (зокрема мобільних пристроїв). Завдяки CSS статична структурна основа (HTML) набуває естетично привабливого вигляду. Аналогія: одяг людини або фарба, шпалери, декор та інші оздоблювальні матеріали в будинку.

JavaScript (JS) – мова програмування, що реалізує динамічну поведінку веб-сторінки. Вона забезпечує інтерактивність: реакцію на дії користувача (кліки, наведення курсора, введення

тексту), оновлення вмісту без перезавантаження сторінки, анімацію, валідацію даних у формах, взаємодію з картами тощо. На відміну від HTML та CSS, які є мовами розмітки та стилів, JavaScript є повноцінною мовою програмування з розгалуженою логікою. Аналогія: нервова система та м'язова система людини, які забезпечують рухливість та реакцію на зовнішні подразники.

Незважаючи на те, що HTML, CSS та JavaScript утворюють фундамент веб-розробки, створення сучасних веб-додатків виключно за допомогою цих «чистих» технологій стикається з низкою обмежень [12-16]. Зростання складності інтерфейсів, необхідність ефективного управління великими обсягами даних та забезпечення швидкодії призвели до появи більш потужних засобів розробки. У зв'язку з цим виникли фреймворки – готові програмні набори інструментів, що стандартизують та прискорюють процес створення клієнтської частини веб-ресурсів.

Сьогодні сайти настільки складні, що писати їх на «чистому» коді довго і важко. Тому розробники використовують фреймворки (готові набори інструментів). Найпопулярніші з них:

- React – бібліотека JavaScript з відкритим кодом, розроблена компанією Facebook у 2013 році. Технічно React є бібліотекою, а не повноцінним фреймворком, однак через розширену екосистему додаткових інструментів її часто відносять до фреймворків. Основна концепція React – компонентний підхід, при якому інтерфейс будується з ізольованих, багаторазово використовуваних блоків (компонентів). Ключовою особливістю є використання віртуального DOM (Document Object Model), що оптимізує оновлення сторінки та підвищує продуктивність. React широко застосовується у таких проєктах, як Facebook, Instagram, Netflix та Airbnb.

- Vue.js – прогресивний JavaScript-фреймворк із відкритим кодом, створений колишнім співробітником Google Еваном Ю 2014 року. Vue.js позиціонується як легкий, гнучкий і простий для освоєння інструмент. На відміну від React та Angular, Vue пропонує поступове впровадження: розробник може використовувати лише окремі модулі фреймворку, не змінюючи всю архітектуру проєкту. Vue.js поєднує переваги React (компонентність, віртуальний DOM) та Angular (двонаправлене зв'язування даних). Фреймворк особливо популярний серед невеликих команд, стартапів та в азіатському регіоні (зокрема, в Китаї, де його використовують Alibaba та Baidu).

- Angular – платформа для розробки веб-додатків, створена компанією Google. Слід розрізняти AngularJS (перша версія, випущена 2010 року) та Angular (повністю переписана версія з 2016 року, яка використовує мову TypeScript). Angular є найбільш важким і комплексним інструментом серед перелічених: він надає розробнику не просто бібліотеку для інтерфейсу, а цілісне рішення «з коробки», що включає роботу з HTTP-запитами, маршрутизацію, управління станом, форми, тестування. Завдяки своїй потужності та структурі Angular часто обирають для великих корпоративних проєктів, внутрішніх систем та додатків із складною логікою. Серед відомих продуктів, створених на Angular, – Gmail (нова версія), Google Cloud Console та PayPal.

Backend (Server-side) – це програмно-апаратна частина сервісу. Це код, який працює на віддаленому комп'ютері (сервері). Бекенд відповідає за роботу з даними, безпеку та бізнес-логіку. Backend як серверна частина веб-додатку складається з трьох основних компонентів, кожен із яких виконує специфічну функцію в загальній архітектурі системи. Сервер – це обчислювальний пристрій (фізичний комп'ютер або віртуальна машина), який функціонує безперервно (режим 24/7) та має постійне підключення до мережі Інтернет. Сервер виконує роль приймача вхідних запитів від клієнтських додатків (браузерів, мобільних застосунків) та генерує відповіді на ці запити. У сучасній практиці сервери можуть бути як фізичними (апаратне забезпечення), так і віртуальними, розгорнутими в хмарних середовищах (Amazon AWS, Google Cloud Platform, Microsoft Azure).

База даних (БД) – це структуроване сховище інформації, організоване відповідно до певних правил зберігання, пошуку та обробки даних. База даних забезпечує постійне (персистентне) зберігання інформації, що не втрачається після вимкнення сервера.

Залежно від способу організації даних, бази даних поділяються на два основних типи:

1. Реляційні (SQL) бази даних – дані зберігаються у вигляді суворо структурованих таблиць, між якими встановлено логічні зв'язки. Для управління такими базами використовується мова структурованих запитів SQL (Structured Query Language). Типові представники: MySQL, PostgreSQL, SQLite, Microsoft SQL Server. Реляційні бази даних забезпечують цілісність даних, підтримку транзакцій та атомарність операцій.

2. Нереляційні (NoSQL) бази даних – дані зберігаються в гнучких форматах без жорсткої схеми таблиць. Інформація може бути організована у вигляді документів (наприклад, JSON-об'єктів), пар «ключ – значення», графів або стовпцевих сімейств. Цей тип баз даних забезпечує високу швидкість при роботі з великими обсягами неструктурованих даних. Типовий представник: MongoDB (документо-орієнтована база даних). Бази даних зберігають усю критичну для функціонування веб-додатку інформацію, зокрема: профілі користувачів (логіни, хеші паролів, персональні дані), каталоги товарів, тексти коментарів, історію замовлень, налаштування системи, тимчасові дані сесій тощо.

Ключовим елементом, що забезпечує взаємодію між фронтендом і бекендом, є API (Application Programming Interface) – інтерфейс програмування застосунків. API виступає посередником, який дозволяє різним частинам системи обмінюватися даними за визначеними правилами. Принцип роботи веб-додатку можна описати як послідовність кроків на рис. 2.

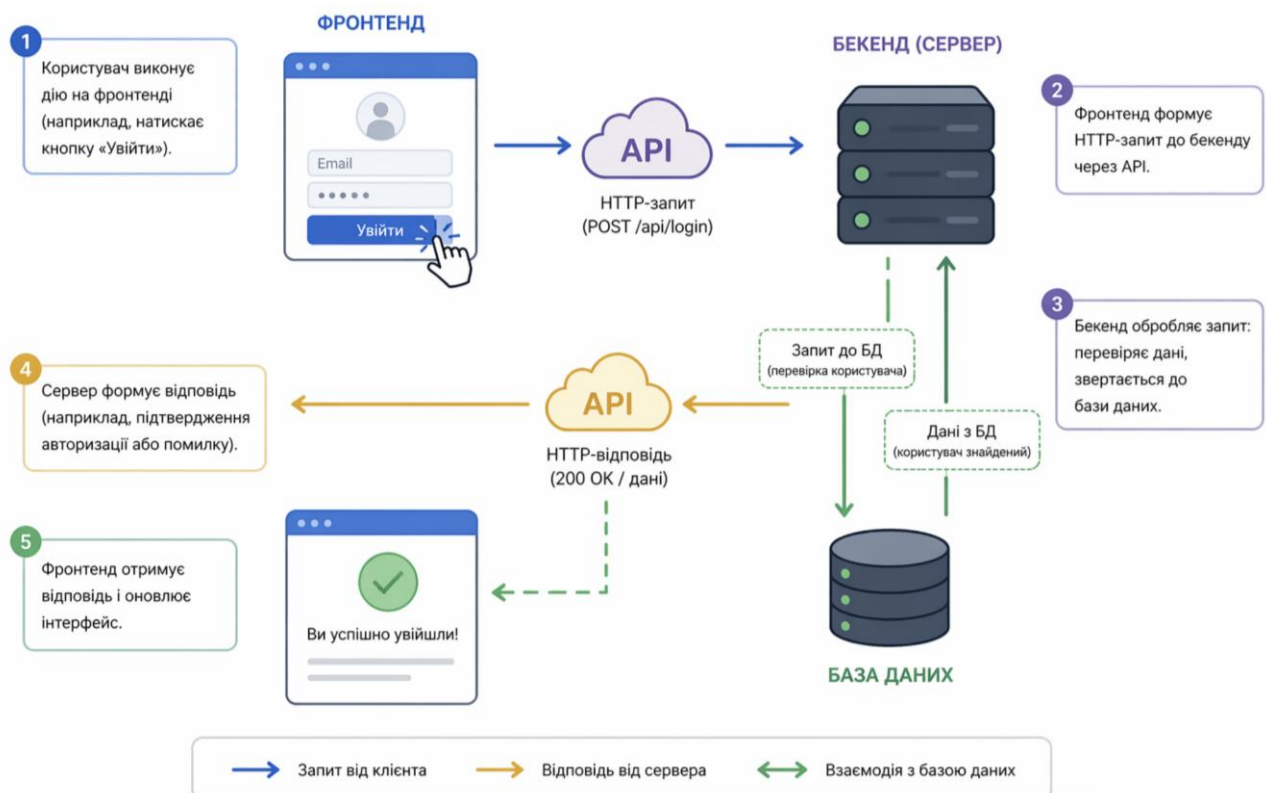


Рисунок 2 – Схема обміну даними між фронтендом, бекендом і базою даних

Такий підхід дозволяє чітко розділити відповідальність між компонентами системи, що є основою сучасної архітектури веб-додатків.

Однією з найпоширеніших моделей побудови веб-систем є клієнт-серверна архітектура, у якій фронтенд виступає клієнтом, а бекенд – сервером. У межах цієї моделі можливі різні підходи до організації взаємодії. Зокрема, виділяють:

- монолітну архітектуру, де всі компоненти (інтерфейс, логіка, база даних) об'єднані в один програмний продукт. Такий підхід простіший у розробці на початковому етапі, але складніший у масштабуванні;
- мікросервісну архітектуру, у якій система складається з незалежних сервісів, кожен із яких відповідає за окрему функцію (наприклад, авторизація, обробка платежів, робота з файлами). Це підвищує гнучкість і масштабованість, але ускладнює управління системою;
- serverless-архітектуру, де розробник не керує серверами напряму, а використовує хмарні функції, які виконуються за запитом. Це дозволяє оптимізувати витрати та автоматично масштабувати систему.

Окрім архітектури, важливим аспектом є спосіб обміну даними між клієнтом і сервером. Найпоширенішими форматами є JSON та XML, причому JSON використовується значно частіше завдяки своїй простоті та зручності для JavaScript.

Ще одним критично важливим компонентом сучасних веб-додатків є **безпека**. Бекенд відповідає за:

- автентифікацію та авторизацію користувачів;
- захист даних (шифрування, HTTPS);
- запобігання атакам (SQL-ін'єкції, XSS, CSRF).

Фронтенд також відіграє роль у безпеці, наприклад, шляхом первинної валідації введених даних, але основний контроль завжди здійснюється на сервері.

Не менш важливим є питання продуктивності веб-додатків. Для її підвищення застосовуються такі підходи:

- кешування даних (зменшення кількості звернень до сервера);
- використання CDN (Content Delivery Network) для швидкої доставки статичних ресурсів;
- оптимізація запитів до бази даних;
- асинхронна обробка даних (наприклад, через AJAX або Fetch API).

У сучасних умовах веб-додатки дедалі частіше розглядаються як повноцінні інформаційні системи, що інтегруються з іншими сервісами: платіжними системами, соціальними мережами, аналітичними платформами. Це підкреслює важливість правильної архітектури, яка забезпечує масштабованість, надійність та зручність підтримки.

У результаті проведеного аналізу було розглянуто архітектуру сучасних веб-додатків крізь призму двох її фундаментальних складових – фронтенду та бекенду. Встановлено, що фронтенд, який базується на трьох базових технологіях (HTML, CSS, JavaScript), відповідає за візуальне представлення даних, інтерактивність та комунікацію з серверною частиною. З'ясовано, що ускладнення сучасних веб-проектів зумовило появу фреймворків (React, Vue.js, Angular), які стандартизують розробку та підвищують продуктивність клієнтської частини. Визначено, що бекенд є програмно-апаратним комплексом, який функціонує на віддалених серверах і включає три ключові компоненти: власне сервер, базу даних та серверний код. Показано, що реляційні (SQL) та нереляційні (NoSQL) бази даних виконують функцію постійного зберігання інформації, тоді як сервер забезпечує безперервну обробку вхідних запитів та реалізацію бізнес-логіки. Доведено, що ефективна взаємодія між фронтендом та бекендом досягається завдяки API (інтерфейсу програмування застосунків), який виступає посередником і регламентує правила обміну даними за протоколом HTTP. Розглянуто різні типи архітектур (монолітну, мікросервісну, serverless), кожна з яких має власні переваги та недоліки щодо масштабованості, складності розробки та супроводу системи. Окрему увагу

приділено питанням безпеки та продуктивності. Встановлено, що, хоча фронтенд забезпечує первинну валідацію даних, основний контроль автентифікації, авторизації та захисту від зловмисних атак покладається на бекенд. Підвищення продуктивності досягається шляхом кешування, використання мереж доставки контенту (CDN) та асинхронної обробки запитів.

Таким чином, фронтенд та бекенд не є ізольованими компонентами, а формують єдину взаємопов'язану інформаційну систему. Їхня узгоджена робота визначає ключові характеристики сучасного веб-додатку: швидкодію, надійність, безпеку та зручність використання. Розуміння розподілу функцій між цими складовими є необхідною умовою для грамотного проектування, розробки та експлуатації веб-ресурсів у сучасну цифрову епоху.

ЛІТЕРАТУРА

1. Polikanov, K., et al. Smart home with house module: overview of automation technologies // International Conference «DIGITAL INNOVATION & SUSTAINABLE DEVELOPMENT 2024», 2024 - pp. 20-21
2. Lvov, A. A., et al. Overview of digital locks with artificial intelligence / A. A. Lvov, S. V. Sotnik // Комп'ютерно-інтегровані технології автоматизації технологічних процесів на транспорті та у виробництві : матеріали всеукр. наук.-практ. конф. здобувачів вищ. освіти і молодих учених, 20 листоп. 2024 р. / Харків. нац. автомоб.-дор. ун-т. – Харків, 2024. – С. 112-116
3. Sotnik, S. V. Development of automated control system and registration of metal in continuous casting // Radio Electronics, Computer Science, Control, 2024. – 3. – pp. 197-211
4. Vasylychenko, Y., et al. Development of Security and Fire Alarm Integrated Automation System at Enterprise // WSEAS Transactions on Systems. – 2025. – № 24. – pp. 642-664
5. Konieva, A., et al. Main trends in the development of automated image processing systems // «Computer-integrated technologies, automation and robotics» CITAR-2025. – 2025. – pp. 68-72
6. Yechevskyi, A. D., et al. Research of orientation methods of autonomous mobile robots in industrial conditions // «Computer-integrated technologies, automation and robotics» CITAR-2025. – 2025. – pp. 115-119
7. Sotnik, S. Development of a range measurement module on an ultrasonic sensor with a GSM module // Radio Electronics, Computer Science, Control. – 2. – 2025. – pp. 32-44
8. Andreiev, A., et al. “Web application security: protection against modern cyber threats” Overview of key vulnerabilities (XSS, CSRF, SQL injections), protection methods, use of HTTPS, authentication, and authorization // Manufacturing & Mechatronic Systems 2025: Proceedings of IX st International Conference, Kharkiv, October 25-26, 2025: Theses of Reports. – 2025. – pp. 66-70
9. Vorobyov, M., et al. Jamstack Architecture as a Synthesis of Serverless Back-End and Dynamic Front-End // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 25-29
10. Dvoynikova, I., et al. Analysis of the Effectiveness and Cybersecurity Risks of the Github Copilot Tool // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 34-38
11. Taran, A., et al. WEB3 and Decentralized Applications. A Practical Look at Blockchain Development // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 81-85
12. Taran, A., et al. Low-Code/No-Code Web Platforms: Opportunities and Limitations // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 96-100
13. Fesenko, A., et al. Comparative Analysis of Programming Languages for Developing System User Interfaces // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic

Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 119-123

14. Taran, A., et al. Impact of 5G/6G Networks on the Development of IOT, Robotics, and Autonomous Systems. Low Latency and Mass Connection of Devices // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 114-118

15. Dvoynikova, I., et al. 6G Networks – A Technological Foundation for Autonomous Systems and the Internet of Everything / I. Dvoynikova, S. Sotnik // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 39-43

16. Andriukhin, I., et al. The Concept of a Digital Twin as a Virtual Copy of Physical Objects, Processes, and Systems // All-Ukrainian Conference “Intelligent Civil Safety Technologies and Robotic Systems for Emergency and Rescue Operations” (ICSTRO-2026) February 12-13, 2026. – pp. 17-21