



DOI 10.28925/2663-4023.2026.35.1384

УДК 004.77:330.322:620.9

Чугреєв Кирилл Александрович

асистент кафедри технологій цифрового розвитку

Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна

ORCID:0009-0005-6951-344X

k.chugreev@duikt.edu.ua**Гончаренко Дмитро Александрович**

магістр

Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна

ORCID: 0009-0000-8182-7496

goncharenkod@gmail.com**Кузьміч Артур Ігорович**

магістр

Державний університет інформаційно-комунікаційних технологій, м. Київ, Україна

ORCID: 0009-0002-8763-0146

Artur1Kuzmich@gmail.com**Кулініч Олег Миколайович**

кандидат технічних наук, доцент, доцент кафедри комп'ютерних систем, мереж та кібербезпеки

Національний університету біоресурсів та природокористування України, Київ, Україна

Київський столичний університет імені Бориса Грінченка, Київ, Україна

ORCID: 0000-0002-0643-6898

olmaxol@i.ua**ІНТЕРАКТИВНИЙ ВЕБ-СЕРВІС ДЛЯ ПОРІВНЯННЯ ТЕРМІНІВ ОКУПНОСТІ ІНВЕСТИЦІЙ У
ЕНЕРГОЕФЕКТИВНІСТЬ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ TYPESCRIPT, NEXT.JS,
NESTJS ТА PRISMA**

Анотація. У статті висвітлено результати розробки та досліджень інтерактивного веб-сервісу для автоматизованого порівняльного оцінювання термінів окупності інвестицій у енергоефективні технології. Актуальність дослідження зумовлена наслідками повномасштабної військової агресії проти України, що спричинили кризу об'єктів критичної інфраструктури та сформували стійкий попит на децентралізовані рішення енергозабезпечення (СЕС, теплові насоси, термомодернізація). Систематизовано параметри українських державних і міжнародних програм фінансової підтримки 2025–2027 років («Енергодім», «ГрінДІМ», «5-7-9%», ПРООН, ЄІБ) для трьох категорій споживачів: домовласників, ОСББ та суб'єктів МСБ. Метою дослідження є підвищення ефективності прийняття інвестиційних рішень шляхом автоматизації розрахунків економічних показників. Запропоновано розширену математичну модель, яка параметризує частку грантового співфінансування, темп зростання тарифів на енергоносії та ставку дисконтування для одночасного обчислення п'яти показників: простого (PP) та дисконтованого (DPP) термінів окупності, чистої приведеної вартості (NPV), внутрішньої норми прибутковості (IRR) та індексу прибутковості (PI). Описано проектування тривірневої C4-архітектури, реляційної ER-моделі даних із сімома сутностями та декларативної Prisma-схеми. Наведено особливості повностекової реалізації на базі TypeScript, Next.js 14, NestJS 10, PostgreSQL та бібліотеки Recharts. Представлено результати багаторівневого тестування (38 тестів, 87% покриття коду) та розрахункові приклади, що підтверджують скорочення окупності для ОСББ у 1,7 раза при 70% гранті. Визначено обмеження системи та перспективи подальшого розвитку, зокрема інтеграцію з Open Banking API та впровадження AI-модуля на основі LLM.

Ключові слова: енергоефективність; окупність інвестицій; веб-сервіс; чиста приведена вартість; внутрішня норма прибутковості; грантове фінансування; TypeScript; Next.js; NestJS; Prisma.



ВСТУП

Наслідки повномасштабної військової агресії проти України спричинили системну кризу об'єктів критичної енергетичної інфраструктури. Руйнування генеруючих потужностей, регулярні відключення електроенергії та системне зростання тарифів сформували стійкий попит на децентралізовані рішення для забезпечення енергетичної автономності домогосподарств, об'єднань співвласників багатоквартирних будинків (ОСББ) та суб'єктів малого і середнього бізнесу (МСБ). Основними технологічними напрямками децентралізації стали сонячні електростанції (СЕС), теплові насоси, системи накопичення енергії, когенераційні установки та комплексні заходи з термомодернізації будівель [3, 4, 5].

Відповіддю держави та міжнародних партнерів стало розгортання у 2025–2027 роках масштабного комплексу програм фінансової підтримки. Ключовим інструментом для ОСББ є програма «Енергодім» Фонду енергоефективності України, яка за Пакетом Б передбачає грантову компенсацію до 70% вартості робіт із термомодернізації (максимально до 25 млн грн на один об'єкт). Паралельно з лютого 2026 року стартувала програма «ГрінДІМ», спрямована на підтримку розподіленої відновлюваної генерації (компенсація до 70% вартості СЕС та теплових насосів з лімітом 2–4 млн грн). Для приватних домогосподарств розширено дію пільгового кредитування за програмою «5-7-9%» (до 150 тис. грн під 5–9% річних), а для МСБ граничну суму пільгового кредиту було збільшено до 250 млн грн під 1–9% річних. Додатковий обсяг зовнішнього фінансування забезпечується програмами ПРООН (\$230 млн на когенерацію) та Європейського інвестиційного банку (€600 млн у межах Ukraine Investment Framework).

Постановка проблеми

Зростання кількості та специфіки програм фінансової підтримки створило для кінцевих споживачів проблему складності прийняття обґрунтованого інвестиційного рішення. Інвестор не має змоги без спеціалізованих інструментів оцінити фінансову ефективність проекту, яка потребує одночасного врахування п'яти базових показників: простого терміну окупності (PP), чистої приведеної вартості (NPV), дисконтованого терміну окупності (DPP), внутрішньої норми прибутковості (IRR) та індексу прибутковості (PI). Аналіз існуючих аналогів на ринку програмного забезпечення дозволяє виділити чотири групи рішень:

1. Калькулятори комерційних банків (ПриватБанк, Ощадбанк) – розраховують графік погашення кредиту та переплату, але не враховують реальної економії від енергозбереження, темпу зростання тарифів та дисконтованих показників.
2. Калькулятори виробників обладнання (SolarEdge, Danfoss) – оцінюють економію для конкретної лінійки товарів, проте є заангажованими, використовують статичні тарифи та не підтримують порівняння між різними технологіями.
3. Державні інформаційні системи (калькулятор Фонду ЕЕ) – обчислюють лише розмір гранту, але не будують повний інвестиційний профіль проекту.
4. Міжнародні аналітичні платформи (HOMER Energy, RETScreen) – мають потужний методологічний апарат, проте не адаптовані до українських пільгових програм, мають складний інтерфейс і є закритими для інтеграції.

Таблиця 1

Порівняльний аналіз існуючих програмних рішень

Критерій	Банківські калькулятори	Калькулятори виробників	Держ. системи (Фонд ЕЕ)	Міжнародні	Проектована система
Врахування грантів UA	Ні	Ні	Частково	Ні	Так
Мультитехнологічне порівняння	Ні	Ні	Ні	Так	Так
Розрахунок NPV, IRR, PI	Ні	Ні	Ні	Так	Так
Актуальні тарифи UA	Частково	Ні	Частково	Ні	Так



Критерій	Банківські калькулятори	Калькулятори виробників	Держ. системи (Фонд ЕЕ)	Міжнародні	Проектована система
UX для нетехнічних осіб	Так	Так	Так	Ні	Так
Збереження та порівняння проєктів	Ні	Ні	Ні	Так	Так
Відкритий REST API	Ні	Ні	Ні	Ні	Так

Таким чином, виявлено три ключових недоліки існуючих програмних рішень: відсутність інтеграції умов українських грантових програм у фінансову модель, відсутність рішення з повним набором показників у зручному UX та відсутність функціоналу порівняння проєктів із відкритим REST API. Виявлені прогалини зумовлюють актуальність розробки спеціалізованого веб-сервісу.

Аналіз останніх досліджень і публікацій

Питання оцінювання ефективності інвестицій в енергозбереження та використання дисконтованих показників висвітлено у працях класиків фінансового менеджменту та сучасних вітчизняних дослідників. Зорієнтованість на застосування методів NPV, IRR та DPP у проєктах відновлюваної енергетики обґрунтовано у роботах А. Дамодарана [7], Є. Бриггема та І. Бланка [1]. Методичні засади розрахунку економічної ефективності енергозберігаючих заходів у комунальному та промисловому секторах України висвітлені у працях В. Розена та О. Борисенко. Паралельно у сфері програмної інженерії активно досліджуються питання побудови високонавантажених повностекових веб-застосунків на мові TypeScript з використанням фреймворків Next.js, NestJS та типобезпечних ORM (Prisma). Принципи архітектурного документування складних інформаційних систем розроблено Саймоном Брауном [6] у формі C4-моделі. Проте, незважаючи на значну кількість публікацій з окремих напрямів, жодне з опрацьованих джерел не пропонує єдиної математичної моделі, яка поєднувала б обчислення п'яти інвестиційних показників з параметричним урахуванням специфіки українських грантових програм, та її реалізації у вигляді масштабованого веб-сервісу з можливістю мультитехнологічного порівняння проєктів [2].

Метою дослідження є підвищення ефективності прийняття рішень щодо інвестування в енергоефективне обладнання шляхом автоматизованого порівняльного розрахунку термінів окупності інвестицій з урахуванням українських грантових програм за допомогою розробки інтерактивного веб-сервісу

РЕЗУЛЬТАТИ ДОСЛІДЖЕННЯ

У результаті дослідження запропоновано метод інтерактивного багатосценарного порівняння інвестицій в енергоефективні технології. На відміну від одиничного веб-калькулятора, запропонований метод передбачає формування за одним набором користувацьких умов множини однотипних сценаріїв для декількох технологій, їх централізовану валідацію, незалежне опрацювання, агрегування, ранжування та автоматичне повторне формування результатів після зміни параметрів. У такій постановці фінансовий розрахунок розглядається як внутрішній предметний модуль, тоді як об'єктом формалізації є алгоритм функціонування веб-сервісу та перетворення даних між його клієнтським, серверним й інформаційним рівнями.

Стан веб-сервісу подано коротцем

$$S = \langle U, D, T, P, G, C, R \rangle. \quad (1)$$

У виразі (1) U є множиною користувачів системи; D - множиною довідкових даних; T - множиною енергоефективних технологій; P - множиною користувацьких проєктів; G - множиною програм фінансової підтримки; C - множиною сформованих сценаріїв; R - множиною результатів. Таке подання відокремлює довідкові сутності системи від даних конкретного сеансу порівняння та дає можливість формально описати процес переходу від параметрів користувача до структурованого результату.

Основне відображення запропонованого методу визначено як



$$\mathcal{M}: (X, T, G, D) \rightarrow \{R_1, R_2, \dots, R_k\}. \quad (2)$$

Тут X містить уніфікований набір параметрів, введених користувачем; T задає каталог технологій; G описує параметри програм фінансування; D містить довідкові дані; R_i є результатом обробки i -го сценарію. Принциповою особливістю (2) є те, що виходом методу є не одиничний результат, а множина взаємно зіставних результатів, отриманих за однакових зовнішніх умов.

Вхідний DTO подано вектором

$$X = [x_1, x_2, \dots, x_m]^T. \quad (3)$$

Кожний компонент x_j відповідає окремому параметру клієнтської форми та серверного DTO. Для кожного параметра задається область допустимих значень Ω_j , а загальна область коректних вхідних даних визначається перетином

$$\Omega = \bigcap_{j=1}^m \Omega_j. \quad (4)$$

Оператор валідації описується функцією

$$V(X) = \begin{cases} X', & X \in \Omega, \\ E(X), & X \notin \Omega, \end{cases} \quad (5)$$

де X' - нормалізований DTO, допущений до подальшої обробки, а $E(X)$ - структурований набір повідомлень про помилки. Повний вектор є допустимим лише за одночасного виконання всіх локальних предикатів валідації:

$$X \in \Omega \Leftrightarrow \bigwedge_{j=1}^m v_j(x_j) = 1. \quad (6)$$

Формули (4)-(6) формалізують дворівневу перевірку даних: клієнтська частина Next.js запобігає очевидним помилкам введення, тоді як серверний рівень NestJS повторно перевіряє DTO незалежно від стану браузера. Це дозволяє не передавати некоректні дані до модуля формування сценаріїв та предметного розрахунку.

Для порівняння користувач обирає підмножину технологій

$$T^* = \{T_1, T_2, \dots, T_k\} \subseteq T. \quad (7)$$

Для кожної технології автоматично формується окремий сценарій

$$C_i = \langle X, T_i, G, D \rangle, \quad i = 1, \dots, k. \quad (8)$$

Вектор X залишається спільним для всіх альтернатив, тоді як T_i задає параметри конкретної технології. Це забезпечує порівнянність сценаріїв, оскільки альтернативи оцінюються за однакових користувацьких та довідкових умов. Предметний модуль веб-сервісу перетворює кожний сценарій на структурований результат

$$R_i = F(C_i). \quad (9)$$

Після завершення циклу обробки формується множина

$$R^* = \{F(C_1), F(C_2), \dots, F(C_k)\}. \quad (10)$$

Усі об'єкти R_i мають однакову структуру, тому можуть автоматично агрегуватися, порівнюватися та сортуватися. Оператор ранжування визначено як

$$R_q = \text{sort}(R^*, q, d), \quad (11)$$



де q - критерій порівняння, а d - напрям впорядкування. Наприклад, для терміну окупності використовується висхідний порядок. Таким чином, механізм порівняння відокремлюється від конкретної формули економічного показника: предметний модуль формує уніфікований R_i , а модуль порівняння працює з його структурованими полями.

Повний шлях даних у веб-сервісі описується композицією функціональних перетворень

$$F_{system} = F_{view} \circ F_{API} \circ F_{compare} \circ F_{calc} \circ F_{validation}. \quad (12)$$

Функція $F_{validation}$ виконує перевірку і нормалізацію DTO; F_{calc} викликає предметний розрахунковий модуль; $F_{compare}$ агрегує та ранжує множину результатів; F_{API} формує контракт серверної відповіді; F_{view} перетворює отримані дані на таблиці, картки та графіки у клієнтській частині. Композиційне подання (12) математично фіксує розподіл відповідальності між компонентами й створює основу для незалежного модульного тестування.

Інтерактивність веб-сервісу формалізовано як ітераційний контур. Якщо після отримання результатів користувач змінює параметри, новий стан вхідного вектора визначається як

$$X^{(s+1)} = X^{(s)} + \Delta X. \quad (13)$$

Після повторної валідації система перебудовує множину сценаріїв і формує новий результат

$$R^{(s+1)} = \mathcal{M}(X^{(s+1)}, T, G, D). \quad (14)$$

Отже, взаємодія користувача із сервісом утворює послідовність станів $R^{(1)}, R^{(2)}, \dots$, що відображає реакцію множини альтернатив на зміну вхідних умов. Саме замкнений контур «введення - валідація - генерація сценаріїв - розрахунок - агрегування - візуалізація - зміна параметрів» відрізняє запропонований метод від статичного веб-калькулятора.

Програмна реалізація методу побудована за трирівневою клієнт-серверною архітектурою. Next.js і TypeScript відповідають за введення параметрів, керування станом, передачу HTTP-запитів та інтерактивне представлення результатів. NestJS реалізує контролери, DTO, ValidationPipe, Guards, сервіс формування сценаріїв та сервіс предметного розрахунку. Prisma забезпечує типізований доступ до PostgreSQL і зв'язує бізнес-логіку з даними про проекти, технології, програми фінансування та результати. Використані технології при цьому розглядаються як засоби реалізації формальної моделі, а не як самостійний науковий результат [11,12,13].

Для аналізу продуктивності час відповіді веб-сервісу декомпозовано як

$$T_{response} = T_{client} + T_{net} + T_{validation} + T_{business} + T_{db} + T_{serialization} + T_{render} \quad (15)$$

У формулі (15) T_{client} характеризує час формування запиту на клієнті; T_{net} - мережеву затримку; $T_{validation}$ - серверну перевірку DTO; $T_{business}$ - виконання бізнес-логіки; T_{db} - взаємодію з базою даних; $T_{serialization}$ - формування JSON-відповіді; T_{render} - оновлення інтерфейсу. Така декомпозиція задає структуру експериментального профілювання сервісу та дозволяє локалізувати компонент, який формує основну затримку.

За послідовної обробки k сценаріїв модельний час визначається як

$$T_{seq} = T_0 + \sum_{i=1}^k T(C_i), \quad (16)$$

де T_0 - постійні накладні витрати, а $T(C_i)$ - час обробки окремого сценарію. Якщо незалежні сценарії реалізовано конкурентно, теоретична оцінка набуває вигляду

$$T_{conc} \approx T_0 + \max_{1 \leq i \leq k} T(C_i) + T_{agg}, \quad (17)$$

де T_{agg} - витрати на очікування, об'єднання та сортування результатів. Теоретичний коефіцієнт прискорення визначається як

$$K_S = \frac{T_{seq}}{T_{conc}}. \quad (18)$$

Формули (16)-(18) доцільно використовувати як модель для подальшої експериментальної перевірки. Вони не означають автоматичного прискорення лише внаслідок використання NestJS, а задають показники, які можуть бути виміряні під час навантажувального тестування реалізованого сервісу.

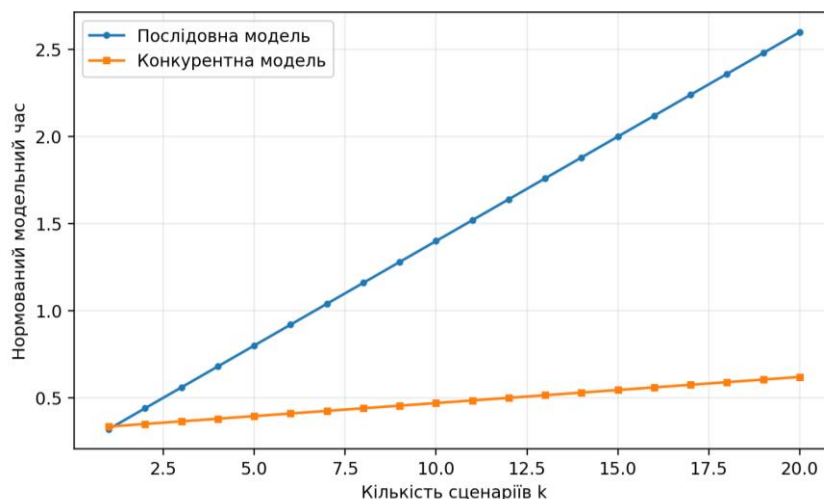


Рис. 1. Теоретична модель часу обробки сценаріїв.

Обчислювальна складність запропонованого алгоритму визначається кількістю технологій k та кількістю часових інтервалів n , які опрацьовує внутрішній предметний модуль. Формування результатів для k сценаріїв має складність $O(kn)$, тоді як сортування множини результатів потребує $O(k \log k)$. Загальна оцінка становить

$$T_{system}(k, n) = O(kn + k \log k). \quad (19)$$

Додаткова пам'ять, необхідна для зберігання множини результатів і робочих даних сценарію, оцінюється як

$$M_{memory} = O(k + n). \quad (20)$$

За типових значень n основною складовою є $O(kn)$, тому при масштабуванні веб-сервісу найбільш доцільними напрямками оптимізації є усунення повторних предметних розрахунків, кешування незмінних довідкових даних та конкурентне опрацювання незалежних сценаріїв за умови, що ефект такої оптимізації підтверджений навантажувальним тестуванням.

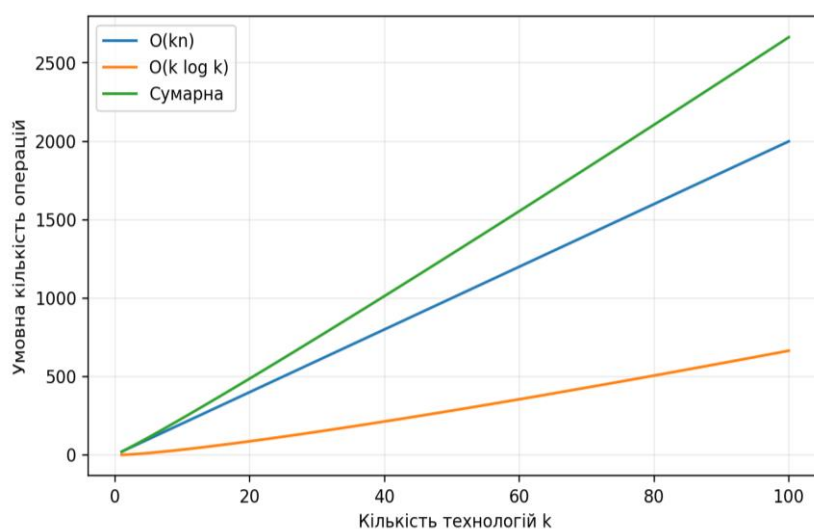


Рис. 2. Теоретична модель зростання обчислювальної складності алгоритму при $n = 20$.

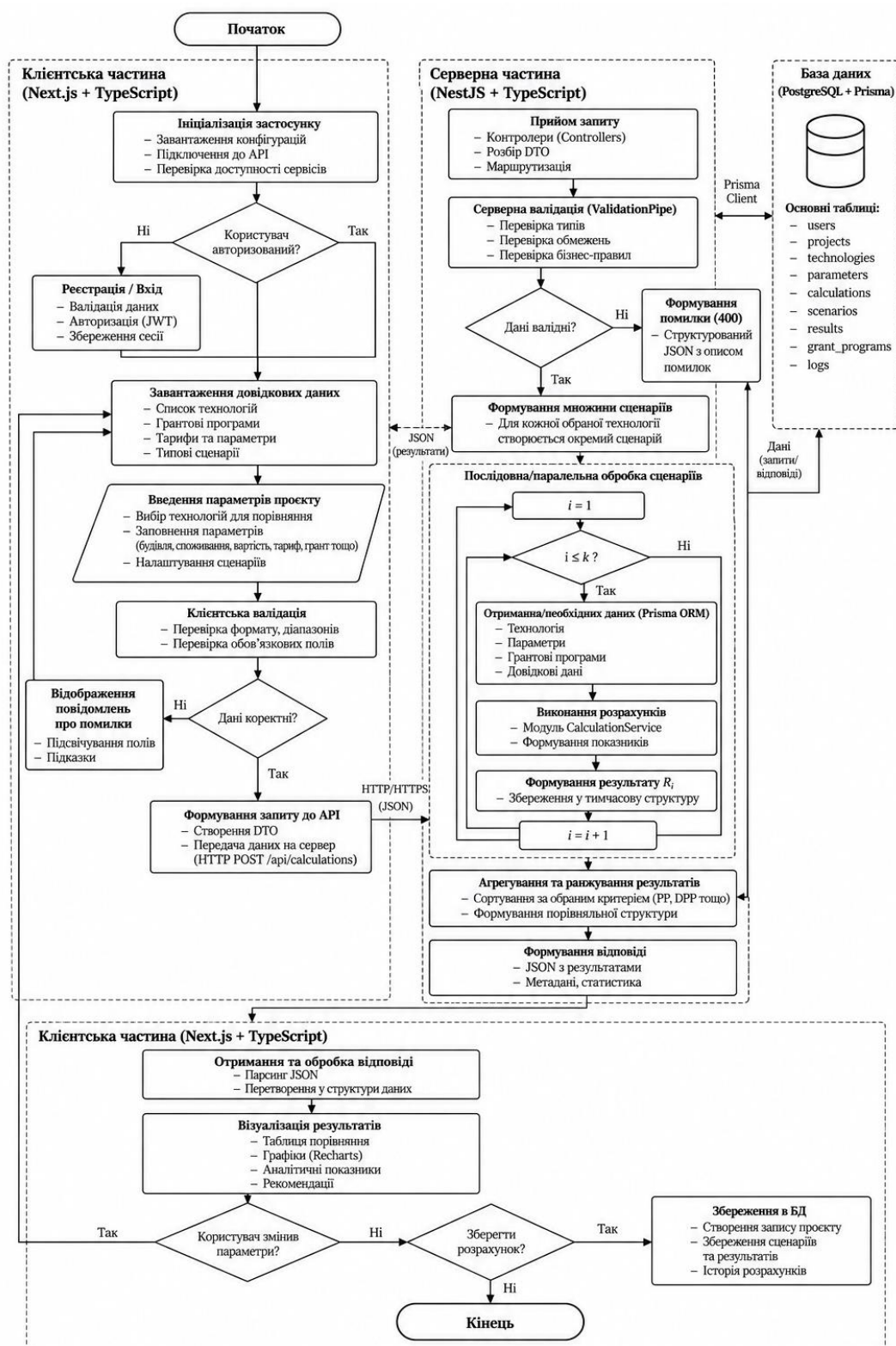


Рис. 3. Use Case діаграма інтерактивного сервісу порівняння термінів окупності

На рисунку 3 представлено блок-схему запропонованого методу інтерактивного багатосценарного порівняння інвестицій в енергоефективні технології, реалізованого у вигляді клієнт-серверного веб-сервісу. На відміну від традиційного підходу, за якого кожна технологія оцінюється окремим незалежним розрахунком, запропонований алгоритм забезпечує формування та обробку множини альтернативних



сценаріїв у межах одного циклу взаємодії користувача із системою. При цьому для всіх альтернатив використовуються спільні початкові умови, що забезпечує їх коректне зіставлення.

Робота алгоритму починається з ініціалізації клієнтської частини веб-сервісу, завантаження конфігурації та перевірки доступності програмного інтерфейсу. Після цього визначається стан авторизації користувача. Для неавторизованого користувача передбачено окрему гілку реєстрації або входу, що включає перевірку введених даних, авторизацію за допомогою JWT та створення користувацької сесії. Після успішної авторизації завантажуються необхідні для подальшого аналізу довідкові дані: перелік доступних енергоефективних технологій, грантових програм, тарифів, параметрів та типових сценаріїв.

На клієнтському рівні користувач обирає декілька технологій для порівняння та задає параметри проєкту. Принципово важливим елементом алгоритму є двоступенева валідація даних. Спочатку перевірка формату, діапазонів та обов'язковості введених значень здійснюється на стороні Next.js. Якщо виявлено помилку, виконання розрахунку припиняється, користувачу відображається відповідне повідомлення, а керування повертається до етапу введення параметрів. Після успішної клієнтської перевірки формується DTO та HTTP-запит до серверної частини. На рівні NestJS виконується повторна серверна валідація за допомогою ValidationPipe, що не дозволяє передати некоректні або неприпустимі дані безпосередньо до бізнес-логіки.

Після підтвердження коректності вхідних даних формується множина сценаріїв порівняння. Для кожної обраної технології створюється окремий сценарій S_i , у якому спільні параметри проєкту поєднуються з характеристиками відповідної технології, параметрами грантової програми та іншими довідковими даними. Саме на цьому етапі одиничний запит користувача перетворюється на задачу багатосценарного аналізу.

Подальша обробка реалізується циклічно для всіх сформованих сценаріїв. Для кожного S_i через Prisma ORM отримуються необхідні дані, після чого модуль CalculationService виконує розрахунок показників, а результат R_i додається до загальної множини результатів. Цикл продовжується доти, доки не буде оброблено всі k обраних технологій. У результаті формується множина $R^* = \{R_1, R_2, \dots, R_k\}$.

Важливо, що сама блок-схема не прив'язує метод до одного конкретного показника окупності. CalculationService є окремим функціональним модулем, тому склад показників може бути розширений без зміни загальної структури алгоритму. Це забезпечує модульність запропонованого рішення та можливість подальшого розширення функціональності веб-сервісу.

Після завершення обробки всіх сценаріїв виконується агрегування отриманих результатів та їх ранжування за обраним критерієм. Формально цей етап може бути представлений оператором $R_q = \text{sort}(R^*, q, d)$, де q визначає критерій порівняння, а d – напрям сортування. Таким чином, результатом роботи серверної частини є не сукупність ізольованих розрахунків, а впорядкована порівняльна структура альтернатив, яка передається клієнтській частині у форматі JSON.

На клієнтському рівні отримані результати перетворюються у форму, зручну для прийняття рішення: порівняльні таблиці, графічні залежності, аналітичні показники та рекомендації.

Найважливішою особливістю запропонованої блок-схеми є наявність замкненого інтерактивного контуру. Після отримання результатів користувач може змінити один або декілька початкових параметрів. У такому разі система не завершує роботу, а повертається до етапу формування параметрів, після чого автоматично повторюються валідація, формування сценаріїв, їх обробка, агрегування та візуалізація результатів.

Завдяки цьому веб-сервіс працює не як звичайний калькулятор, що формує один результат за одним набором параметрів, а як інтерактивний інструмент дослідження альтернатив. Користувач може змінювати умови та одразу спостерігати, як ці зміни впливають на взаємне положення технологій у сформованому рейтингу. Це дає можливість досліджувати декілька альтернатив за однакових початкових умов і оцінювати стійкість отриманого вибору до зміни параметрів проєкту.

Ще однією особливістю є чітке розмежування клієнтського, серверного та інформаційного рівнів. Next.js + TypeScript відповідають за взаємодію з користувачем і візуалізацію; NestJS + TypeScript – за валідацію, формування сценаріїв, бізнес-логіку, обчислення та агрегування; Prisma + PostgreSQL – за отримання і збереження структурованих даних. Тому блок-схема одночасно відображає алгоритмічну логіку запропонованого методу та спосіб її програмної реалізації [8, 9, 10].

У результаті виконання алгоритму формується не лише числове значення терміну окупності. Система отримує множину уніфікованих результатів для всіх вибраних технологій, їх порівняльне впорядкування за заданим критерієм, дані для побудови таблиць і графіків, а також, за необхідності, збережений сценарій разом з історією розрахунків. При зміні початкових умов формується нова множина результатів, що дає можливість аналізувати зміну переваг окремих технологій залежно від параметрів проєкту.

Запропонована блок-схема формалізує повний цикл інтерактивного багатосценарного порівняння



енергоєфективних технологій – від формування та двоступеневої валідації вхідних даних до автоматичного створення множини сценаріїв, їх незалежної обробки, агрегування, ранжування та інтерактивної візуалізації результатів [14, 15]. Особливістю запропонованого підходу є поєднання циклічної багатосценарної обробки з контуром динамічного перерахунку при зміні початкових параметрів, що дає змогу перейти від одиничного розрахунку окупності до інтерактивного дослідження альтернативних варіантів інвестування та їх порівняння за однакових початкових умов. На рисунках 4а та 4б представлено екранні форми Е-сервісу, що реалізує запропонований метод [16].

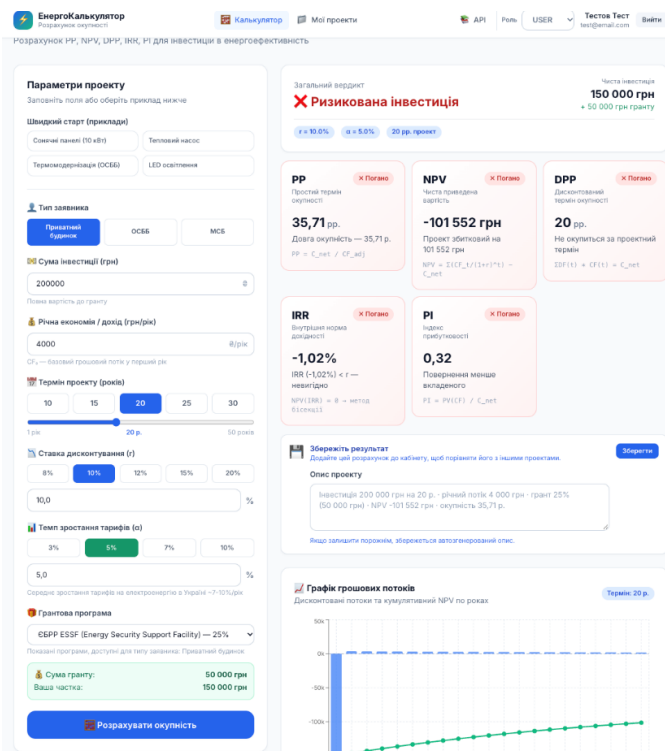


Рис. 4а. Екранна форма розрахунку невігідного проекту

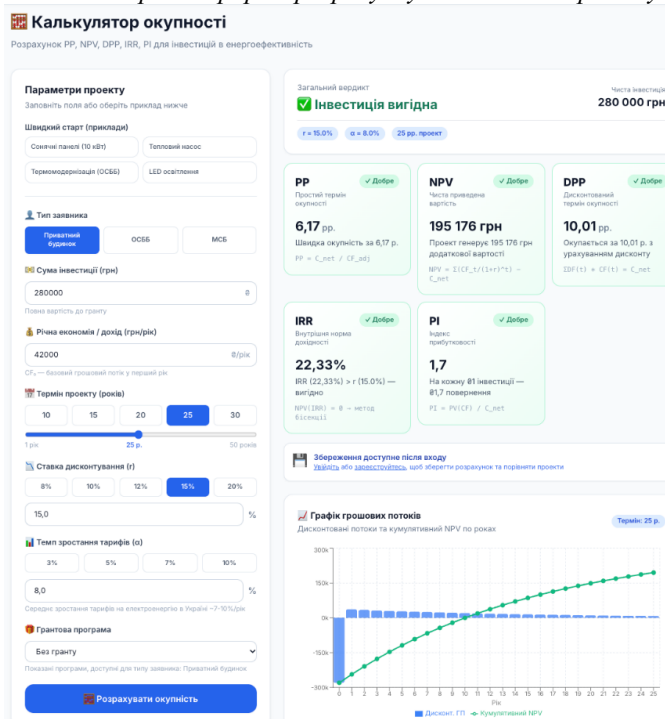


Рис. 4б. Екранна форма розрахунку вигідного проекту



Науковий результат полягає в удосконаленні методу порівняльного оцінювання інвестицій в енергоефективні технології шляхом введення інтерактивного багатосценарного механізму, який на основі уніфікованого вектора вхідних параметрів забезпечує централізовану валідацію, автоматизоване формування множини альтернативних сценаріїв, незалежне отримання результатів для кожної технології, їх агрегування та ранжування, а також динамічне повторне формування результатів при зміні початкових умов. Запропонована математична модель формалізує саме процес функціонування веб-сервісу, тоді як TypeScript, Next.js, NestJS та Prisma забезпечують його програмну реалізацію.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У результаті проведеного дослідження розроблено інтерактивний веб-сервіс для порівняння термінів окупності інвестицій у енергоефективність. Основні наукові та практичні результати полягають у наступному:

1. Систематизовано параметри українських програм фінансування («Енергодім», «ГрінДІМ», «5-7-9%», ПРООН, ЄІБ) та сформовано класифікаційну матрицю для трьох профілів споживачів (домовласники, ОСББ, МСБ).
2. Побудовано розширену математичну модель, яка вперше забезпечує одночасне обчислення п'яти інвестиційних показників (PP, NPV, DPP, IRR, PI) з параметричним урахуванням частки гранту, темпу зростання тарифів та ставки дисконтування у єдиному обчислювальному контурі.
3. Спроектовано трирівневу C4-архітектуру та реляційну схему даних на 7 сутностях із підтримкою каскадного видалення та тип-безпечного доступу через Prisma ORM. Реалізовано повностековий веб-застосунок на TypeScript (Next.js 14 + NestJS 10) з інтерактивними діаграмами Recharts, модулем порівняння до 5 проєктів та адміністративною панеллю керування довідниками.
4. Проведено тестування системи (38 тестів, 87% покриття), яке підтвердило повну відповідність обчислень теоретичним розрахункам ($PP = 2,5\$$ року для ОСББ з грантом 70%; $PP = 4,2\$$ року для СЕС домовласника).
5. Виявлені обмеження системи охоплюють незбіжність методу Ньютона-Рафсона для нетипових грошових потоків (повертає NaN), а також фіксований курс валют при розрахунку МСБ-проєктів.

Перспективи подальших досліджень полягають у таких напрямках:

1. Інтеграція з Open Banking API банків-партнерів для автоматичного передзаповнення кредитних заявок за програмою «5-7-9%».
2. Розробка AI-модуля на основі великих мовних моделей (LLM) для генерації персоналізованих рекомендацій щодо вибору обладнання.
3. Розширення розрахункового модуля на масштабні когенераційні установки понад 1 МВт з підтримкою мультивалютних розрахунків.
4. Реалізація автоматичного парсингу оновлень Фонду ЕЕ та Міністерства відновлення для щоденного оновлення нормативних умов програм.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бланк, І. О., Гуляєва, Н. М., & Вавдійчик, І. М. (2023). *Інвестиційний менеджмент*. Державний торговельно-економічний університет.
2. Гончаренко, Д. О. (2026). *Інтерактивний веб-сервіс для порівняння термінів окупності інвестицій у енергоефективність з використанням технологій TypeScript, NextJS, NestJS та Prisma* (Кваліфікаційна робота бакалавра, Державний університет інформаційно-комунікаційних технологій).
3. Кабінет Міністрів України. (2026, 28 січня). *Про затвердження Порядку надання фінансової державної підтримки суб'єктам господарювання (програма "Доступні кредити 5-7-9%")* (Постанова № 28).
4. Фонд енергоефективності України. (2026a). *Програма «ГрінДІМ»*. <https://eefund.org.ua/novyny/grindim/>
5. Фонд енергоефективності України. (2026b). *Програма «Енергодім»*. <https://eefund.org.ua/energodim/>
6. Brown, S. (2019). *Software architecture for developers: Visualise, document and explore your software architecture* (Vol. 2). Leanpub.
7. Damodaran, A. (2012). *Investment valuation: Tools and techniques for determining the value of any asset*. Wiley.
8. NestJS. (n.d.). *NestJS documentation: A progressive Node.js framework*. <https://docs.nestjs.com>



9. Next.js. (n.d.). *Next.js 14 documentation: App Router & React Server Components*. <https://nextjs.org/docs>
10. OWASP Foundation. (2025). *OWASP Top 10:2025 application security risks*. <https://owasp.org/Top10/2025/>
11. PostgreSQL. (2026). *PostgreSQL 16.0 documentation*. <https://www.postgresql.org/docs/16/index.html>
12. Prisma. (n.d.). *Prisma ORM documentation: Next-generation Node.js and TypeScript ORM*. <https://www.prisma.io/docs>
13. Recharts. (n.d.). *Recharts: A composable charting library built on React components*. <https://recharts.github.io/>
14. Tailwind CSS. (n.d.). *Tailwind CSS documentation*. <https://tailwindcss.com/docs>
15. USAID Energy Security Project. (2025). *Market analysis of energy efficiency technologies for small and medium businesses in Ukraine*. USAID.
16. Поперешняк С., Жебка В., Ананченко О. (2025) Цифрові технології як технічний фундамент розробки масштабованих Е-сервісів та цифрових платформ. *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»*, 3(31), 541–558. <https://doi.org/10.28925/2663-4023.2025.31.1053>

**Kyryll Chuhreiev**

Assistant at the Department of Digital Development Technologies
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID: 0009-0005-6951-344X
k.chugreev@duikt.edu.ua

Dmytro Honcharenko

Master's degree
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID: 0009-0000-8182-7496
goncharenkod@gmail.com

Artur Kuzmich

Master's degree
State University of Information and Communication Technologies, Kyiv, Ukraine
ORCID: 0009-0002-8763-0146
Artur1Kuzmich@gmail.com

Oleh Kulinich

Candidate of Technical Sciences, Associate Professor,
Department of Computer Systems, Networks and Cybersecurity
National University of Life and Environmental Sciences of Ukraine, Kyiv, Ukraine
Borys Grinchenko Kyiv Metropolitan University, Kyiv, Ukraine
ORCID: 0000-0002-0643-6898
olmaxol@i.ua

**INTERACTIVE WEB SERVICE FOR COMPARING ENERGY EFFICIENCY INVESTMENT
PAYBACK PERIODS USING TYPESCRIPT, NEXT.JS, NESTJS AND PRISMA**

Abstract. The article presents the results of the development and research of an interactive web service for automated comparative evaluation of investment payback periods in energy-efficient technologies. The relevance of the research is driven by the consequences of full-scale military aggression against Ukraine, which caused a crisis in critical infrastructure facilities and formed a strong demand for decentralized energy supply solutions (solar power plants, heat pumps, thermal modernization). The parameters of Ukrainian state and international financial support programs for 2025–2027 ("Energodim", "GreenDIM", "5-7-9%", UNDP, EIB) are systematized for three categories of consumers: homeowners, OSBB (condominium associations), and SMEs. The aim of the study is to increase the efficiency of investment decision-making by automating calculations of economic indicators. An extended mathematical model is proposed, which parameterizes the grant co-financing share, energy tariff growth rate, and discount rate for the simultaneous calculation of five key indicators: Payback Period (*PP*), Discounted Payback Period (*DPP*), Net Present Value (*NPV*), Internal Rate of Return (*IRR*), and Profitability Index (*PI*). The design of a three-level C4 architecture, a relational ER data model with seven entities, and a declarative Prisma schema are described. Features of the full-stack implementation based on TypeScript, Next.js 14, NestJS 10, PostgreSQL, and the Recharts library are presented. Results of multi-level testing (38 tests, 87% code coverage) and calculation examples confirming a 1.7-fold reduction in payback period for OSBB with a 70% grant are provided. Limitations of the system and prospects for further development are identified, including integration with Open Banking APIs and the implementation of an LLM-based AI module.

Keywords: energy efficiency; investment payback; web service; net present value; internal rate of return; grant financing; TypeScript; Next.js; NestJS; Prisma.

REFERENCES (TRANSLATED AND TRANSLITERATED)

1. Blank, I. O., Huliaieva, N. M., & Vavdiichyk, I. M. (2023). *Investytsiyni menedzhment* [Investment management]. State University of Trade and Economics.
2. Brown, S. (2019). *Software architecture for developers: Visualise, document and explore your software*



- architecture (Vol. 2). Leanpub.
3. Cabinet of Ministers of Ukraine. (2026, January 28). *Pro zatverdzhennia Poriadku nadannia finansovoi derzhavnoi pidtrymky subiektam hospodariuvannia (prohrama "Dostupni kredyty 5-7-9%")* [On approval of the Procedure for providing state financial support to business entities ("Affordable Loans 5-7-9%" program)] (Resolution No. 28).
 4. Damodaran, A. (2012). *Investment valuation: Tools and techniques for determining the value of any asset* (3rd ed.). Wiley.
 5. Energy Efficiency Fund of Ukraine. (2026a). *Prohrama "HrinDIM"* ["GreenDIM" Program]. <https://eefund.org.ua/novyny/grindim/>
 6. Energy Efficiency Fund of Ukraine. (2026b). *Prohrama "Enerhodim"* ["Energodim" Program]. <https://eefund.org.ua/energodim/>
 7. Honcharenko, D. O. (2026). *Interaktyvnyi veb-servis dlia porivniannia terminiv okupnosti investytsii u enerhoefektyvnist z vykorystanniam tekhnolohii TypeScript, NextJS, NestJS ta Prisma* [Interactive web service for comparing energy efficiency investment payback periods using TypeScript, NextJS, NestJS and Prisma] [Bachelor's thesis, State University of Information and Communication Technologies].
 8. NestJS. (n.d.). *NestJS documentation: A progressive Node.js framework*. Retrieved April 26, 2026, from <https://docs.nestjs.com>
 9. Next.js. (n.d.). *Next.js 14 documentation: App Router & React Server Components*. Retrieved April 26, 2026, from <https://nextjs.org/docs>
 10. OWASP Foundation. (2025). *OWASP Top 10:2025 application security risks*. <https://owasp.org/Top10/2025/>
 11. PostgreSQL. (2026). *PostgreSQL 16.0 documentation*. <https://www.postgresql.org/docs/16/index.html>
 12. Prisma. (n.d.). *Prisma ORM documentation: Next-generation Node.js and TypeScript ORM*. Retrieved April 26, 2026, from <https://www.prisma.io/docs>
 13. Recharts. (n.d.). *Recharts: A composable charting library built on React components*. Retrieved April 26, 2026, from <https://recharts.github.io/>
 14. Tailwind CSS. (n.d.). *Tailwind CSS documentation*. Retrieved April 26, 2026, from <https://tailwindcss.com/docs>
 15. USAID Energy Security Project. (2025). *Market analysis of energy efficiency technologies for small and medium businesses in Ukraine*. USAID.
 16. Popereshniak S., Zhebka V., Ananchenko O. (2025) Digital Technologies as a Technical Foundation for the Development of Scalable E-Services and Digital Platforms. *Electronic Scholarly Journal "Cybersecurity: Education, Science, Technology"*, 3(31), 541–558. <https://doi.org/10.28925/2663-4023.2025.31.1053>

Отримано редакцією журналу / Received: 07.09.26

Прорецензовано / Revised: 19.09.26

Схвалено до друку / Accepted: 26.10.26



This work is licensed under Creative Commons Attribution-noncommercial-sharealike 4.0 International License.