

**Волошин В.С.**

Національний університет водного господарства та природокористування

**Опанасюк М.Ю.**

Національний університет водного господарства та природокористування

## ІННОВАЦІЙНІ ПІДХОДИ ДО РОЗРОБКИ СЕРВЕРНОЇ ЧАСТИНИ WEB-ЗАСТОСУНКУ КОНТРОЛЮ БЮДЖЕТУ ДОМОГОСПОДАРСТВА З ВИКОРИСТАННЯМ EXPRESS

У статті досліджено питання розробки серверної частини Web-застосунку для контролю бюджету домогосподарства, з акцентом на інноваційні підходи та використанням сучасних Web-технологій. Проаналізовано предметну область для визначення основних вимог користувачів і тенденцій у сфері управління фінансами. У статті розглянуто існуючі рішення, такі як *Mint Expense Tracker*, *You Need A Budget* та *Rocket Guard*, що автоматизують процеси відстеження витрат і бюджетування, виступають основою для формування власних ідей. Встановлено, що найкращим інструментом для розробки серверної частини Web-додатку є фреймворк *Express.js*, що працює на базі *Node.js*. В рамках розробки застосовано мікросервісну архітектуру, яка дозволяє масштабувати окремі компоненти системи, не впливаючи на її загальну роботу. Крім того, реалізовано підхід *API-first*, що забезпечує легку інтеграцію із зовнішніми сервісами та мобільними застосунками. З метою підвищення функціональності, було використано графові бази даних для управління даними про фінансові транзакції та взаємозв'язки між категоріями витрат. Застосовано алгоритми машинного навчання для аналізу даних і надання користувачам рекомендацій щодо економії. Оптимізовано інтеграцію з фреймворком *Next.js*, що дозволило забезпечити швидке завантаження сторінок завдяки серверному рендерингу. Встановлено, що важливою перевагою *Express.js* є можливість попередньо генерувати статичні сторінки, що зменшує навантаження на сервер. Виявлено, що для локального зберігання даних про фінанси найкраще використовувати *IndexedDB*, що надає можливість зберігати великі об'єми даних прямо в браузері користувача. Це дозволяє застосунку працювати без постійного підключення до мережі, зберігаючи дані в асинхронному режимі. Застосовано компонентний підхід у процесі розробки інтерфейсу, що спрощує подальшу підтримку та розвиток системи. Додатково впроваджено функцію нічного режиму та автоматичне генерування звітів для поліпшення взаємодії з користувачем.

**Ключові слова:** програмування, бази даних *IndexedDB*, *Express.js*, *Back-end*, *API*, *NoSQL*, домогосподарство.

**Постановка проблеми.** У сучасному світі, де технології швидко розвиваються та бізнес-процеси стають все більш складними, ефективне управління фінансовими ресурсами стає визначальним фактором для досягнення успіху в глобальному економічному середовищі. Спостерігаючи за цим трендом, виникає важлива потреба в створенні інноваційних інструментів для оптимізації управління витратами коштів. Актуальність питання застосування інноваційних підходів до розробки серверної частини WEB-застосунку контролю бюджету домогосподарства, впливає із зростаючої конкуренції та швидкому темпі змін у економічному середовищі, що змушують підприємства шукати нові, ефективні методи управління фінансами.

**Аналіз останніх досліджень і публікацій.** Питаннями управління, аналізу та контролю

бюджетом домогосподарства займаються багато українських та закордонних науковців. Так, наприклад, Ясишена В., Долюк А., Ярова Ю. вважають, що ключовим аспектом управління бюджетом домогосподарства є відповідальність за свої фінанси і прийняття рішень, які відображають потреби та цілі [1, с. 282]. У свою чергу Крилова І. досліджувала питання регулювання водопостачання та водовідведення у різних країнах світу, зокрема у Бельгії кожен оператор послуг має запровадити шестирічний прогноз на основі «Тарифного плану» з конкретними цінами на споживання домогосподарств та тарифи, які їх не стосуються [2, с. 68]. Європейські науковці Ванус Й. (Vanus J.), Кучера П. (Kucera P.), Мартінек Р. (Martinek R.) та Козьорек Й. (Koziorek J.) розробляли прикладне програмне забезпечення

візуалізації, яке використовується для керування операційними та технічними функціями в системі «Розумний дім». Пропонуються ефективні методи перевірки надійності створеного програмного забезпечення [3]. Австрійські дослідники Марушич Л. (Marusic L.), Скоцир П. (Skocić P.), Петрич А. (Petric A.), Єжич Г. (Jezic G.) вважали, що одним із способів зменшення споживання енергії домогосподарством є підвищення обізнаності споживачів, люди часто недбалі, коли йдеться про зниження споживання енергії, навіть якщо це може добре вплинути на їхній сімейний бюджет. Тому авторами презентовано мобільний сервіс Home-in-Palm для дистанційного контролю споживання електроенергії домогосподарствами, який дозволяє дистанційно керувати побутовою технікою, автоматично змінювати стан приладу залежно від місця розташування користувача або певного моменту часу, а також автоматично зменшувати споживання енергії домогосподарством [4].

**Постановка завдання.** Розробити серверну частину WEB-застосунку контролю бюджету домогосподарства з використанням фреймворку Express.js застосувавши інноваційні підходи, для автоматичного отримання фінансових даних з метою їхнього подальшого аналізу та ефективного прийняття управлінських рішень.

**Виклад основного матеріалу.** У контексті проектування та розробки серверної частини Web-застосунку контролю бюджету домогосподарства, обов'язковим етапом є аналіз предметної області. Цей аналіз дозволяє визначити основні вимоги, тенденції та виклики, з якими стикаються люди під час управління фінансами. Нижче розглянуті популярні застосунки або ж Web-платформи для ведення обліку витрат коштів.

Mint Expense Tracker [5] є Web-сервісом та мобільним додатком, які дозволяють користувачам автоматично відстежувати свої витрати, аналізувати бюджет та отримувати підказки щодо економії. Додаток є частиною фінансового платформи Mint, яка належить компанії Intuit. Основна суть даного додатку полягає в тому, щоб надати користувачам можливість зберігати в одному місці всю інформацію про їхні фінанси, витрати та доходи, і надавати аналіз цих даних для кращого розуміння їхнього фінансового стану.

Функції Mint Expense Tracker включають:

- автоматичне відстеження витрат: система автоматично підключається до банківських рахунків та кредитних карт користувача, а потім авто-

матично структурує транзакції за різними видами витрат (їжа, транспорт, розваги тощо);

- створення бюджету: користувачі можуть встановлювати місячні бюджети для різних категорій та слідкувати за їхнім виконанням. Mint надсилає сповіщення, якщо витрати виходять за встановлені рамки;

- аналітика та звіти: сервіс надає графіки та звіти щодо фінансового стану, розподілу витрат, змін в бюджеті та інші аналітичні дані для кращого розуміння фінансової ситуації.

You Need A Budget [6] – це програма для особистого фінансового управління, спрямована на допомогу користувачам ефективно вести бюджет та досягати фінансових цілей. Додаток розвивається компанією YNAB, і його філософія базується на ідеї «Every Dollar Has a Job» («Кожен долар має своє призначення»). Даний застосунок підходить до розробки фінансових навичок та допомагає користувачам бути впевненими у своєму фінансовому майбутньому, забезпечуючи прозорість та контроль над їхніми грошима. Також YNAB пропонує велику кількість освітніх матеріалів, щоб допомогти користувачам краще розуміти принципи бюджетування та фінансового планування.

Pocket Guard [7] – це мобільний додаток та Web-сервіс, призначений для автоматизації ведення бюджету та відстеження витрат. Основна мета – допомогти користувачам контролювати свої фінанси, створювати розумні бюджети та досягати фінансових цілей. Застосунок вирізняється своєю простотою використання та можливістю надавати користувачам широкий огляд їхніх фінансів в одному місці. За допомогою цього застосунку користувачі можуть ефективно керувати своїми фінансами та планувати витрати. Застосунок вирізняється своєю простотою використання та можливістю надавати користувачам широкий огляд їхніх фінансів в одному місці.

Враховуючи дослідження успішних Web-додатків, пропонується власна розробка серверної частини Web-застосунку контролю бюджету домогосподарства (рис. 1) з використанням вільного та відкритого програмного забезпечення Express.js [8], що працює на базі Node.js та Next.js.

У процесі розробки серверної частини даного Web-додатку були використані наступні інноваційні підходи:

- мікросервісна архітектура: розбиття застосунку на незалежні мікросервіси, що дозволяє масштабувати та оновлювати окремі компоненти без впливу на всю систему;

– API-first підхід: розробка API на початку, що дозволяє забезпечити гнучкість у інтеграції з іншими сервісами та мобільними застосунками;

– реалізація графічних баз даних: використання графових баз даних для більш ефективного управління даними про фінансові транзакції та взаємозв'язки між категоріями витрат;

– використання штучного інтелекту: інтеграція алгоритмів машинного навчання для аналізу фінансової інформації та надання рекомендацій щодо економії;

– динамічне управління категоріями: інтерфейс, який дозволяє користувачам легко налаштувати категорії витрат та отримувати рекомендації на основі їхніх фінансових звичок;

– кросплатформенність: розробка серверної частини з урахуванням можливості інтеграції з різними Front-End технологіями, щоб забезпечити доступність на різних пристроях.

Дані підходи можуть суттєво покращити функціональність та ефективність роботи Web-застосунку.

У свою чергу потрібно відмітити, що використання Next.js може вважатися інноваційним підходом для розробки Web-застосунків, у тому числі для сервісів контролю бюджету домогосподарства, оскільки Next.js підтримує серверний рендеринг, що покращує швидкість завантаження сторінок та SEO. Це особливо важливо для застосунків, які потребують швидкого доступу до інформації. Також існує можливість попередньо генерувати статичні сторінки, що зменшує навантаження на сервер і підвищує швидкість роботи Web-додатку. Next.js автоматично оптимізує завантаження коду, завантажуючи лише те, що потрібно для конкретної

сторінки. Це покращує продуктивність застосунку. Досить важливим також є властивість Next.js легко підключатися до серверних API, що робить його зручним для роботи з мікросервісами та зовнішніми даними, а просте управління маршрутами допомагає створювати зрозумілі URL-структури, що важливо для користувацького досвіду.

У процесі розробки серверної частини Web-застосунку контролю бюджету домогосподарства було використано NoSQL бази даних IndexedDB, що є вбудованою архітектурою у браузерах, яка у свою чергу надає можливість Web-додаткам зберігати та отримувати доступ до великої кількості структурованої інформації.

Основні переваги, які надає IndexedDB:

– локальне зберігання: IndexedDB надає можливість Web-застосункам локально зберігати великі об'єми даних без необхідності відправки їх на сервер;

– асинхронний запит до бази даних: робота з даними відбувається асинхронно, що дозволяє уникнути блокування виконання інших завдань під час обробки запитів;

– структуровані дані в форматі ключ-значення: це дозволяє зберігати та вибирати дані за ключами, що полегшує організацію та отримання доступу до інформації;

– вбудована підтримка об'єктів та JSON: можливість зберігати об'єкти та дані у форматі JSON дозволяє працювати з різноманітними типами даних та забезпечує гнучкість в обробці структурованих даних.

Для доступу до бази даних створено сервіс Storage, що допоможе асинхронно отримувати та записувати дані (рис. 2).

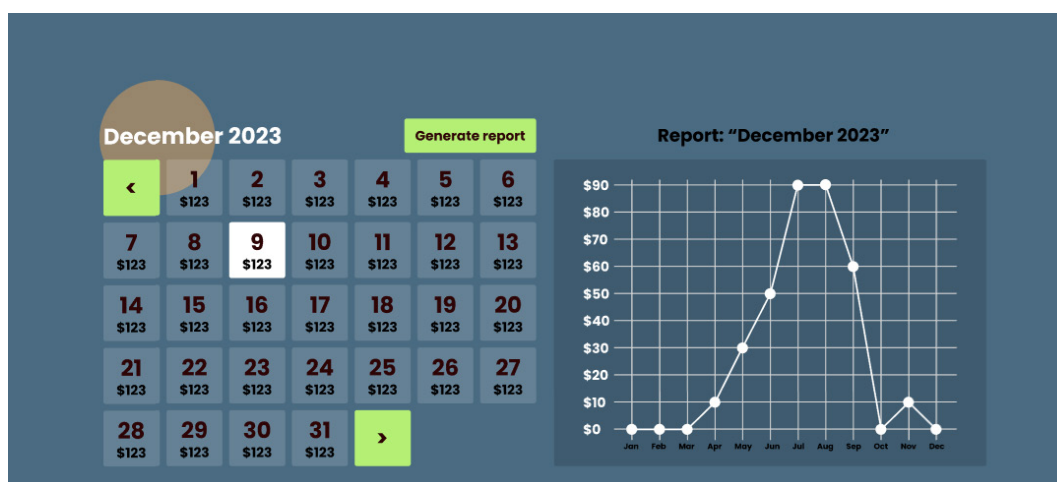


Рис. 1. Приклад побудови графіку витрат Web-застосунку контролю бюджету домогосподарства на базі Express.js

Джерело: власна розробка

```

32 export const storage = {
33   expenses: {
34     get,
35     insert: async (day: number, month: number, year: number, text: string, amount: number) => {
36       const key = `${year}-${month}-${day}`;
37
38       const expenses = await get(day, month, year);
39
40       await store.setItem<ExpenseRecord[]>(key, [...expenses, { text, amount, id: uniqid() }]);
41     },
42     remove: async (day: number, month: number, year: number, id: string) => {
43       const key = `${year}-${month}-${day}`;
44
45       const expenses = await get(day, month, year);
46
47       const index = expenses.findIndex(e => e.id === id);
48
49       if (index !== -1) {
50         await store.setItem(key, [
51           ...expenses.slice(0, index),
52           ...expenses.slice(index + 1),
53         ]);
54       }
55     },

```

Рис. 2. Створення сервісу Storage бази даних IndexedDB серверної частини Web-застосунку контролю бюджету домогосподарства

Джерело: власна розробка

Під час створення інтерфейсу користувача були використані компоненти, що дозволило перевикористати елементи інтерфейсу, спростити подальші зміни та імплементацію застосунку. Вцілому, використання компонентів не лише оптимізує процес створення інтерфейсу, але й забезпечує зручність для подальшого масштабування та розвитку застосунку, що у кінцевому результаті позитивно вплине на взаємодію з користувачами. Основні компоненти, що були використані у розробленому Web-застосунку: Background.tsx (створення фону), Block.tsx (елементи, що відображають порядковий номер дня місяця), BlockButton.tsx (блок кнопок), ExpenseDialog.tsx (відображення модальних вікон), ReportView.tsx (аналіз вихідних даних).

На рисунку 3 наведено програмний код файлу конфігурації TypeScript «tsconfig.json».

Даний файл виконує кілька важливих функцій у процесі компіляції застосунку. Він визначає специфічні параметри, які будуть використовуватись під час компіляції, що забезпечує правильну конфігурацію проекту. Це може включати налаштування для оптимізації продуктивності, вибору версій мов програмування або фреймворків, а також визначення середовища виконання.

Після додавання усіх витрат домогосподарства, користувач має можливість згенерувати звіт, у розробленому Web-застосунку. Для цього необхідно перейти на головну сторінку Web-сайту та натиснути кнопку «Generate report» (рис. 4).

**Висновки.** Використанням Express.js є сучасним інноваційним підходом до розробки сервер-

```

1  {
2    "compilerOptions": {
3      "target": "ESNext",
4      "useDefineForClassFields": true,
5      "lib": [
6        "DOM",
7        "DOM.Iterable",
8        "ESNext"
9      ],
10     "allowJs": false,
11     "skipLibCheck": true,
12     "esModuleInterop": false,
13     "allowSyntheticDefaultImports": true,
14     "strict": true,
15     "forceConsistentCasingInFileNames": true,
16     "module": "ESNext",
17     "moduleResolution": "Node",
18     "resolveJsonModule": true,
19     "isolatedModules": true,
20     "noEmit": true,
21     "jsx": "react-jsx"
22   },
23   "include": [
24     "src"
25   ],
26   "references": [
27     {
28       "path": "./tsconfig.node.json"
29     }
30   ]
31 }

```

Рис. 3. Конфігураційний файл TypeScript Web-застосунку контролю бюджету домогосподарства

Джерело: власна розробка

ної частини WEB-застосунку контролю бюджету домогосподарства. Сьогодні можна розглядати для застосування наступні Web-додатки для



контролю фінансів: Mint Expense Tracker, You Need A Budget та Pocket Guard, які автоматизують процес відстеження витрат, бюджетування та надання фінансових рекомендацій. Власна розробка серверної частини Web-застосунку здійснювалася на базі Express.js, із застосуванням мікросервісної архітектури та підходу API-first, що підвищує гнучкість та масштабованість інформаційної системи. Інновації у розробці Back End включають використання графових баз даних для управління транзакціями, інтеграцію машинного навчання для аналізу даних, динамічне управління категоріями витрат, а також підтримку кро-сплатформенності для зручності користувачів на різних пристроях. NoSQL бази даних IndexedDB досить інтенсивно використовуються для локального зберігання даних, що дозволяє працювати з великими об'ємами структурованої інформації без постійного звернення до сервера. Викорис-

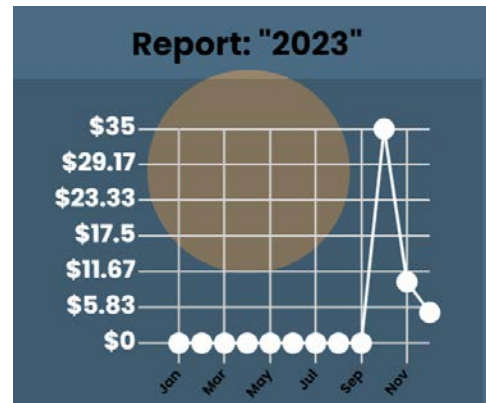


Рис. 4. Автоматично згенерований звіт по витратам за місяцями системи контролю бюджету домогосподарства з використанням Express

Джерело: власна розробка

тання компонентного підходу у розробці інтерфейсу спрощує його масштабування та забезпечує перевикористання елементів.

#### Список літератури:

1. Ясишена В., Долюк А., Ярова Ю. Політика управління бюджетом домогосподарств. *Вісник Хмельницького національного університету*. 2024. Том 326. №1. С. 280–288.
2. Крилова І.І. Регулювання у сфері водопостачання та водовідведення в різних країнах світу. *Вчені записки ТНУ імені В. І. Вернадського. Серія: Державне управління*. 2019. Том 30 (69). № 1. С. 66–73.
3. Vanus J., Kucera P., Martinek R., Koziorek J. Development and testing of a visualization application software, implemented with wireless control system in smart home care. *Human-centric Computing and Information Sciences*. 2014. Volume 4. URL: <https://link.springer.com/article/10.1186/s13673-014-0019-5> (дата звернення: 19.09.2024).
4. Marusic L., Skocir P., Petric A., Jezic G. Home-in-Palm – A mobile service for remote control of household energy consumption. *Proceedings of the 11th International Conference on Telecommunications: conference15-17 June 2011. Austria : Graz*. 2011. URL: <https://ieeexplore.ieee.org/abstract/document/5969917> (дата звернення: 19.09.2024).
5. Budget Tracker & Planner : веб сайт. URL: <https://mint.intuit.com/> (дата звернення: 19.09.2024).
6. YNAB : веб сайт. URL: <https://www.ynab.com/> (дата звернення: 19.09.2024).
7. PocketGuard : Budgeting App & Finance Planner : веб сайт. URL: <https://pocketguard.com/> (дата звернення: 19.09.2024).
8. Express.js – fast, unopinionated, minimalist Web framework for Node.js : веб сайт. URL: <https://expressjs.com/> (дата звернення: 19.09.2024).

#### Voloshyn V.S., Opanasiuk M.Yu. INNOVATIVE APPROACHES TO THE DEVELOPMENT OF THE SERVER PART OF THE HOUSEHOLD BUDGET CONTROL WEB APPLICATION USING EXPRESS

The article explores the development of the backend for a household budget management web application, with a focus on innovative approaches and the use of modern web technologies. The subject area is analyzed to determine the main user requirements and trends in financial management. The article reviews existing solutions, such as Mint Expense Tracker, You Need A Budget, and Pocket Guard, which automate expense tracking and budgeting processes and serve as a foundation for generating original ideas. It has been established that the best tool for backend development of the web application is the Express.js framework, running on Node.js. A microservices architecture has been applied, allowing the scaling of individual system components without affecting overall performance. Additionally, an API-first approach was implemented to facilitate easy integration with external services and mobile applications. To enhance functionality, graph databases were used to manage financial transaction data and relationships between expense categories. Machine learning algorithms were employed to analyze data and provide users with savings recommendations. Integration with the Next.js framework was optimized, ensuring fast page loading through server-side rendering. It was determined that one of Express.js key advantages is the ability to pre-generate static pages, reducing server load. For local financial data storage, IndexedDB was identified as the best option, as it allows large amounts of data to be stored directly in the user's browser, enabling the application to function offline while asynchronously saving data. A component-based approach was used in interface development, simplifying future system maintenance and expansion. Additionally, a dark mode feature and automatic report generation were implemented to improve user experience.

**Key words:** programming, IndexedDB databases, Express.js, Back-end, API, NoSQL, housekeeping.