

3. Звіт Міністерства освіти і науки України з виконання оперативного плану Міністерства освіти і науки України на 2020 рік та основні цілі на 2021 рік. URL: <https://mon.gov.ua/storage/app/media/rizne/2021/03.02.2021/ZVIT%20MINISTERSTVA%20OSVITY%20I%20NAUKY%20UKRAYINY%20Z%20VYKONANNYA%20OPERATYVNOHO%20PLANU.pdf> (дата звернення: 09.10.2023).

4. Про освіту: Закон України від 05.09.2017 р. № 2145-VIII. Дата оновлення: 02.07.2023. URL: <https://zakon.rada.gov.ua/laws/show/2145-19> (дата звернення: 10.10.2023).

5. Ляхощка Л. Л., Ляхощкий В. П. Цифрова освіта і наука – запорука національної безпеки України. Національна безпека України у викликах новітньої історії: кол. монографія. Київ: ДП «Експрес-об'ява», 2019. С. 277–289.

6. Концепція цифрової трансформації освіти і науки. URL: <https://mon.gov.ua/ua/news/konceptsiya-cifrovoi-transformaciyi-osviti-i-nauki-mon-zaproshtuye-do-gromadskogo-obgovorennya> (дата звернення: 10.10.2023).

УДК 004.415.3

## ПРОЄКТУВАННЯ ЗАСТОСУНКУ ІНТЕЛЕКТУАЛЬНОГО АНАЛІЗУ ТА КОНТРОЛЮ ОСОБИСТИХ ФІНАНСІВ

*А. О. Ємельянова, О. В. Зелінська*

*Анотація.* У дослідженні подана інформація про процес проектування вебзастосунку інтелектуального аналізу і контролю власних фінансів. Описуються механізми збору і аналізу даних, зазначаються необхідні обчислення для здійснення аналітики грошових операцій. Наводиться опис основних технологій реалізації, а саме мови програмування JavaScript та її інтерфейсної бібліотеки React.js, їх тенденції розвитку, переваги та особливості використання. Результати проведених досліджень є основою і відіграють важливу роль у подальшій розробці вебзастосунку – програмній реалізації.

*Ключові слова:* вебзастосунок, електронний контроль фінансів, інтелектуальний аналіз даних, JavaScript, React.js.

Гроші відіграють мало не першочергову роль у житті людини, тому відстежування власних фінансів є надзвичайно важливим і актуальним питанням у сучасному світі. Знання того, скільки грошей має людина, які витрати вона повинна зробити та скільки грошей очікувати, допомагає планувати своє життя та забезпечувати фінансову стабільність. Відсутність контролю над особистими фінансами може призвести до низки проблем, як-от накопичення боргів, витрат, що перебільшують дохід, неспроможність забезпечити себе необхідними товарами та послугами, нестабільність фінансів тощо.

Додаток для контролю за фінансами може допомогти зрозуміти, куди йдуть гроші, де можна зекономити та як краще розпланувати свій бюджет. Вебзастосунок для контролю за особистими фінансами повинен надавати користувачу зручні інструменти для відстеження своїх доходів та витрат, аналізу своєї фінансової ситуації та планування витрат на майбутнє.

Метою статті є проектування застосунку, який має полегшити ведення та контроль особистих фінансів із елементами інтелектуального аналізу даних на основі мови програмування JavaScript та її популярної і широко використовуваної інтерфейсної бібліотеки React.js.

Механізм аналізу даних включає кілька кроків і методів, які допомагають отримати інсайти з великого обсягу даних. Загальний опис механізму даних зображений на рис. 1.



Рис. 1. Механізм аналізу даних

Розглянемо більш детально кожен крок процесу аналізу даних [1–3].

Спочатку необхідно зібрати дані з різних джерел, як-от бази даних, файли, API тощо. Цей крок може включати збір структурованих даних (наприклад, числа, таблиці) та/або неструктурованих даних (текст, зображення, відео). Далі дані потрібно очистити та підготувати для подальшого аналізу. Це включає видалення некоректних або відсутніх значень, виправлення помилок, нормалізацію даних тощо. Для кращого розуміння даних важливо візуалізувати їх у вигляді графіків, діаграм або інших візуальних елементів. Це допомагає виявити патерни, тренди та взаємозв'язки між даними.

Для отримання інсайтів з даних застосовуються різні аналітичні методи, як-от статистичний аналіз, класифікація, кластеризація, прогнозування, асоціативний аналіз тощо. Вони допомагають розуміти залежності, знаходити важливі змінні та зробити прогнози.

Після аналізу даних слід проаналізувати результати та зробити висновки. Це може включати інтерпретацію виявлених патернів, пояснення знайдених залежностей та прийняття рішень на основі отриманих інсайтів. Остаточні результати аналізу даних можуть бути використані для прийняття рішень, планування дій, вдосконалення процесів, прогнозування майбутніх подій та впровадження стратегій у сфері фінансів або будь-якій іншій галузі.

Однак варто зазначити, що розглянутий вище механізм аналізу даних є загальною процедурою і може варіюватися залежно від конкретних вимог, методів аналізу та використовуваних інструментів. Фінансові трекери та програми для контролю фінансів можуть використовувати різні формули та показники для аналізу фінансового стану. Розглянемо декілька загальних формул, які можуть застосовуватися в таких програмах:

1. Загальний дохід – сума всіх отриманих доходів за певний період. Це може включати заробітну плату, проценти від інвестицій, дивіденди та інші джерела доходу.

2. Загальні витрати – усі витрати, здійснені за певний період, як-от витрати на їжу, житло, транспорт, розваги, платежі за кредити тощо.

3. Чистий прибуток – розраховується як різниця між загальним доходом та загальними витратами. Він вказує на суму грошей, яка залишається після оплати всіх витрат:

$$\text{Чистий прибуток} = \text{Загальний дохід} - \text{Загальні витрати.} \quad (1)$$

4. Зміна стану рахунку – відображення змін стану рахунку відносно часу. Допомагає відслідковувати збільшення або зменшення фінансового стану:

$$\text{Стан рахунку} = \text{Початковий баланс} + \text{Чистий прибуток.} \quad (2)$$

5. Співвідношення витрат до доходів – визначення відношення витрат до загального доходу і вказування на те, скільки відсотків доходу витрачається:

$$\text{Співвідношення витрат до доходів} = \frac{\text{Витрати}}{\text{Дохід}} \times 100 \%. \quad (3)$$

6. Відсоток збережень – відсоток чистого прибутку, який виділяється на збереження або інвестування:

$$\text{Відсоток збережень} = \frac{\text{Чистий прибуток}}{\text{Загальний дохід}} \times 100 \%. \quad (4)$$

7. Відсоток витрат на категорію (частка витрат) – цей показник визначає відсоток загальних витрат, які спрямовуються на оплату певної категорії (оплата комунальних послуг, продуктів, проїзду тощо) в порівнянні з загальними витратами:

$$\text{Частка витрат} = \frac{\text{Витрати на категорію}}{\text{Загальні витрати}} \times 100 \%. \quad (5)$$

8. Рентабельність – відсоток прибутку відносно загального доходу:

$$\text{Рентабельність} = \frac{\text{Чистий прибуток}}{\text{Загальний дохід}} \times 100 \%. \quad (6)$$

9. Середній чек – визначення середньої вартості кожної транзакції.

$$\text{Середній чек} = \frac{\text{Загальний дохід}}{\text{Кількість транзакцій}}. \quad (7)$$

10. Частка доходу від різних джерел – підсумування надходжень із джерел, з яких отримується дохід, як-от заробітна плата, інвестиції, пасивний дохід тощо.

11. Динаміка доходів у часі – зміна доходу протягом певного періоду.
12. Динаміка витрат у часі – зміна витрат протягом певного періоду.
13. Динаміка збережень – зміни збережень протягом часу.

JavaScript [4] – це багатопарадигмальна, однопоточна, динамічна мова програмування на основі прототипу, яка підтримує об’єктно-орієнтований, імперативний і декларативний стилі (наприклад, функціональне програмування). Хоча ця мова програмування найбільш відома як мова сценаріїв для вебсторінок, багато середовищ без браузерів також використовують її, наприклад, Node.js, Apache CouchDB і Adobe Acrobat. JavaScript існує вже майже три десятиліття, і її універсальність, здатність обслуговувати Front-end та Back-end розробку зробила її основою інструментів більшості розробників [5].

Згідно з опитуванням, проведеним Stack Overflow у 2023 році, JavaScript посідає першу сходинку у рейтингу мов програмування [6] (рис. 2).

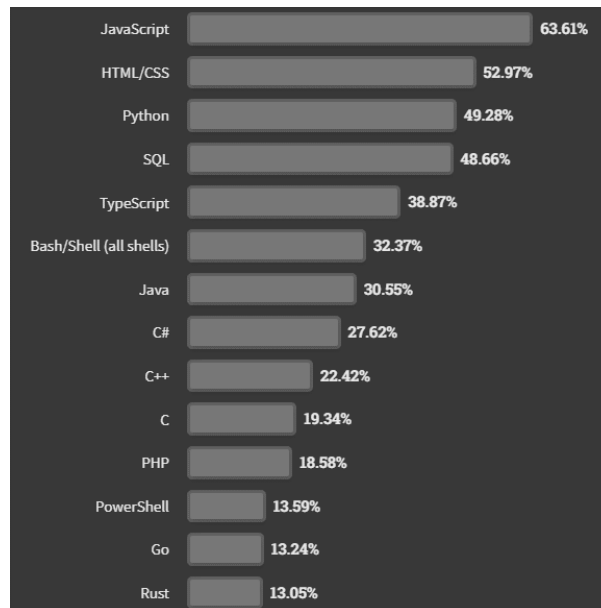


Рис. 2. Найпопулярніші мови програмування у світі за 2023 рік, згідно з опитуванням StackOverflow [6]

Принцип роботи JavaScript коду полягає у такому: він або вбудовується безпосередньо у вебсторінку, або посилається на нього через окремий файл формату \*.js. Коли користувач відвідує вебсторінку, його браузер запускає сценарій разом із кодом HTML і CSS, створюючи функціональну сторінку, що відображається у вкладці браузера. Сценарій завантажується на машини користувачів, де і проходить процес обробки. Це відрізняється від серверної мови, у якій сервер обробляє сценарій перед тим, як надсилати його в браузер. Зустрічаючи блок JS-коду, веббраузер поступово оброблює його зверху донизу. Оскільки він дуже чутливий до порядку, необхідно посилатись на об’єкти чи змінні в блоці перед тим, як змінювати їх. Наявність змінних без значень призведе до невизначеної помилки [7].

Переваги використання мови програмування JavaScript такі [7]:

1. Простота – проста структура спрощує вивчення та впровадження JS. Також вона працює швидше за деякі інші мови. Помилки легко виявляти та виправляти.
2. Швидкість – JavaScript виконує сценарії безпосередньо у веббраузері без попереднього підключення до сервера чи компілятора. До того ж більшість основних браузерів дозволяють JS компілювати код під час виконання програми.
3. Універсальність – JavaScript сумісний з іншими мовами програмування, як-от PHP, Perl і Java. Це також робить Data Science та машинне навчання доступними для розробників.
4. Популярність – у вільному доступі безліч ресурсів і форумів, які допоможуть новачкам із обмеженими технічними навичками та знаннями у JavaScript.

5. Навантаження на сервер – ще одна перевага роботи на стороні клієнта полягає в тому, що JS зменшує кількість запитів, які надсилаються на сервер. Перевірку даних можна виконати через веббраузер, а оновлення стосуються лише певних розділів вебсторінки.

6. Оновлення – команда розробників JavaScript і ECMA International постійно оновлюють і створюють нові фреймворки і бібліотеки, забезпечуючи їх актуальність.

ReactJS (React, React.js) [8] – це JavaScript-бібліотека, розроблена компанією Meta для створення динамічних та інтерактивних додатків і кращого дизайну UI/UX для веб- і мобільних додатків [9]. Це інтерфейсова бібліотека на основі компонентів, яка відповідає лише за рівень перегляду архітектури контролера перегляду моделі (Model View Controller або MVC). React використовується для створення модульних користувацьких інтерфейсів і сприяє розробці повторно використовуваних компонентів інтерфейсу користувача, які відображають динамічні дані [10]. Також React відомий створенням односторінкових додатків (SPA) і мобільних додатків.

Основною метою React є мінімізація помилок, які виникають, коли розробники створюють інтерфейси користувача. Це робиться за допомогою компонентів – самодостатніх, логічних фрагментів коду, які описують частину інтерфейсу користувача. Ці компоненти можна об'єднати разом, щоб створити повноцінний інтерфейс користувача. React абстрагує більшу частину роботи рендерингу, зосереджуючи увагу на дизайні інтерфейсу користувача [11].

Згідно з опитуванням, проведеним Stack Overflow у 2023 році, React посідає друге місце у рейтингу найпопулярніших вебфреймворків та технологій [12] (рис. 3).

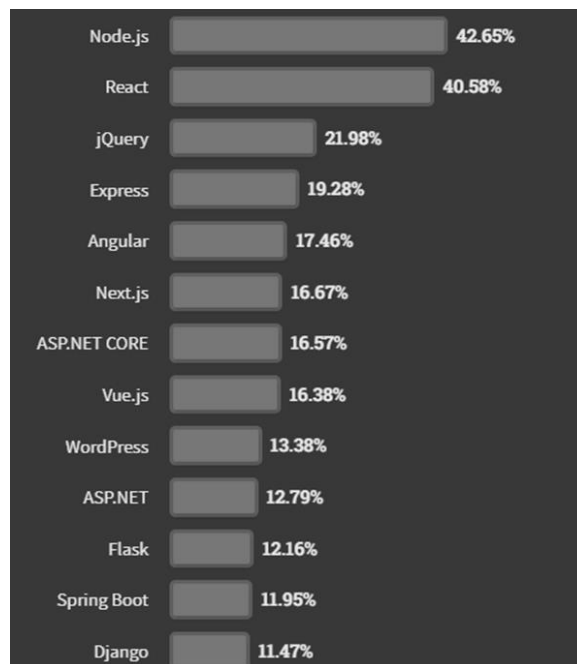


Рис. 3. Найпопулярніші вебфреймворки та технології світу за 2023 рік, згідно з опитуванням StackOverflow [12]

ReactJS використовує декларативну парадигму, яка дає змогу програмам бути ефективними та гнучкими. Він ефективно оновлює та відтворює потрібний компонент у міру зміни даних і створює прості подання для кожного стану в програмі. Декларативне подання полегшує налагодження та передбачуваність коду. Кожен компонент програми React відповідає за рендеринг окремого фрагмента HTML-коду, який можна використовувати декілька разів. Можливість вкладати одні компоненти у інші дає змогу створювати складні програми із простих «будівельних блоків» [10].

До особливостей React можна віднести JSX (розширення синтаксису JavaScript), віртуальний DOM (Virtual DOM), односпрямований потік даних, продуктивність, розширення, умовні твердження, компоненти і простоту [9].

Одними з ключових переваг React є [13–15]:

1. Virtual DOM – підвищення продуктивності завдяки reconciliation.
  2. SEO-friendly – традиційні фреймворки JavaScript мають проблеми з SEO. Пошукові системи зазвичай мають проблеми з читанням програм, які містять багато JavaScript-коду. React долає цю проблему, допомагаючи розробникам легко орієнтуватися в різних пошукових системах. Все через те, що програми React.js можуть запускатись на сервері, а віртуальний DOM відображатиметься та повертатиметься до браузера як звичайна вебсторінка.
  3. Легке створення динамічних додатків – створювати динамічну вебпрограму тільки за допомогою HTML було складно, оскільки це вимагає складного кодування, але React вирішив цю проблему та спростив її. Він забезпечує менше кодування та надає більше функціональних можливостей за допомогою JSX.
  4. Легке тестування коду – React пропонує область, де розробник може тестувати та налагоджувати свій код за допомогою вбудованих інструментів.
  5. Велика спільнота розробників і легкість в освоєнні – React має достатню кількість документації, посібників і навчальних ресурсів. Будь-який розробник із досвідом JavaScript може легко зрозуміти та почати створювати вебдодатки за допомогою React за кілька днів.
- Отже, у цьому дослідженні розглянуто і описано механізми збору та аналізу даних. Визначено необхідні обчислення, що використовуватимуться для аналізу даних та їх візуалізації на підставі внесених фінансових операцій під час програмної реалізації вебзастосунку. Також проведений аналіз і вибір технологій, необхідних для реалізації застосунку шляхом детального огляду тенденцій розвитку, переваг і особливостей використання, що є важливим для подальшого розвитку вебзастосунку.

*Abstract.* This study provides information on the process of designing a web-based application for intelligent analysis and control of personal finances. Data collection and analysis mechanisms are described, and the necessary calculations for analyzing monetary transactions are indicated. The article describes the main implementation technologies, namely the JavaScript programming language and its interface library React.js, their development trends, advantages, and features of use. The results of the conducted research are the basis and play an important role in the further development of the web application – software implementation.

*Keywords:* web application, electronic financial control, intelligent data analysis, JavaScript, React.js.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Аналіз даних. *Вікіпедія*. 2020. URL: [https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D1%96%D0%B7\\_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85](https://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B0%D0%BB%D1%96%D0%B7_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85)
2. Основи статистичного аналізу даних. URL: <https://www.miyklas.com.ua/p/informatica/10-klas/modeli-i-modeliuvannia-analiz-ta-vizualizatsiia-danikh-326184/modeli-i-modeliuvannia-analiz-ta-vizualizatsiia-danikh-325755/re-a43672ea-4fe7-4839-8746-f03a2572fea0>
3. Основи статистики та аналізу даних. URL: <https://socialdata.org.ua/manual/manual4/>
4. JavaScript. *MDN Web Docs*. 2023. URL: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
5. O’Grady B. What is JavaScript? URL: <https://codeinstitute.net/global/blog/what-is-javascript-and-why-should-i-learn-it/>
6. Programming, scripting, and markup languages. *2023 Developer Survey*. 2023. URL: <https://survey.stackoverflow.co/2023/#section-most-popular-technologies-programming-scripting-and-markup-languages>
7. Jordana A. What Is JavaScript? *A Basic Introduction to JS for Beginners*. Jordana. 2023. URL: <https://www.hostinger.com/tutorials/what-is-javascript>
8. React The library for web and native user interfaces. URL: <https://react.dev/>
9. What are the features of ReactJS? 2023. URL: <https://www.geeksforgeeks.org/what-are-the-features-of-reactjs/>
10. ReactJS Introduction. 2023. URL: <https://www.geeksforgeeks.org/reactjs-introduction/>
11. Getting started with React. *MDN Web Docs*. 2023. URL: [https://developer.mozilla.org/en-US/docs/Learn/Tools\\_and\\_testing/Client-side\\_JavaScript\\_frameworks/React\\_getting\\_started](https://developer.mozilla.org/en-US/docs/Learn/Tools_and_testing/Client-side_JavaScript_frameworks/React_getting_started)
12. Web frameworks and technologies. *2023 Developer Survey*. 2023. URL: <https://survey.stackoverflow.co/2023/#section-most-popular-technologies-web-frameworks-and-technologies>
13. Deshpande C. The Best Guide to Know What Is React. 2023. URL: <https://www.simplilearn.com/tutorials/reactjs-tutorial/what-is-reactjs>
14. Pros and Cons of ReactJS. URL: <https://www.javatpoint.com/pros-and-cons-of-react>
15. Goralski C. Pros and Cons of React. 2022. URL: <https://thecodest.co/blog/pros-and-cons-of-react/>