

ПРОГРАМНА СИСТЕМА ДЕЦЕНТРАЛІЗОВАНОГО ОБМІНУ ТОКЕНІВ З МЕХАНІЗМАМИ КЕРУВАННЯ ЛІКВІДНІСТЮ

Вінницький національний технічний університет

Анотація

Розглянуто проблематику забезпечення доступності та зручності використання децентралізованих фінансових інструментів. Запропоновано архітектурне рішення вебзастосунку на базі сучасного технологічного стека (Next.js, Wagmi, viem) з інтеграцією протоколу Uniswap V3 для виконання обміну токенів та агрегатора DefiLlama для моніторингу пулів ліквідності. Проаналізовано переваги такого підходу порівняно з існуючими DeFi-інтерфейсами.

Ключові слова: децентралізовані фінанси; обмін токенів; Uniswap V3; смарт-контракти; ліквідність; Ethereum.

Abstract

The problem of ensuring accessibility and usability of decentralized financial instruments is considered. An architectural solution for a web application based on a modern technology stack (Next.js, Wagmi, viem) with Uniswap V3 protocol integration for token swapping and DefiLlama aggregator for liquidity pool monitoring is proposed. The advantages of this approach compared to existing DeFi interfaces are analyzed.

Keywords: decentralized finance; token exchange; Uniswap V3; smart contracts; liquidity; Ethereum.

Вступ

Децентралізовані фінанси (DeFi) є одним із найбільш динамічно зростаючих секторів сучасної індустрії програмного забезпечення, а загальний обсяг заблокованих коштів у DeFi-протоколах перевищив 100 млрд. дол. США [1]. Проте більшість існуючих інтерфейсів є монофункціональними: вони або надають доступ до одного протоколу обміну, або зосереджуються виключно на агрегуванні доходності [2]. Це змушує користувача використовувати кілька окремих застосунків. Тому актуальною є розробка уніфікованої системи, що поєднає обмін токенів, моніторинг ліквідності та відстеження портфеля в єдиному інтерфейсі.

Основна частина

Ключовою технічною проблемою при розробці DeFi-застосунку є взаємодія клієнтського застосунку зі смарт-контрактами в блокчейні. Традиційні підходи до побудови вебзастосунків на базі REST API не підходять для цієї мети, оскільки блокчейн є децентралізованою системою без централізованих серверних ендпоінтів [3]. Натомість клієнтський застосунок має безпосередньо взаємодіяти з вузлами мережі Ethereum через провайдери JSON-RPC.

Для реалізації системи обрано стек технологій, що включає фреймворк Next.js 16 (App Router) з React 19 для побудови клієнтського інтерфейсу та бібліотеки Wagmi і viem для взаємодії з блокчейном Ethereum [4]. Wagmi забезпечує набір React-хуків для підключення гаманців, читання стану контрактів та відправки транзакцій, тоді як viem виступає низькорівневим типобезпечним клієнтом для роботи з Ethereum JSON-RPC. Керування станом асинхронних запитів реалізовано за допомогою TanStack Query. Підключення гаманців (MetaMask, WalletConnect та ін.) відбувається через Reown AppKit – уніфікований UI-компонент з підтримкою соціальної автентифікації.

Функцію обміну токенів реалізовано через інтеграцію з протоколом Uniswap V3 [5] у тестовій мережі Sepolia. Окремий програмний модуль інкапсулює логіку отримання цінової котировки від смарт-контракту QuoterV2 та виконання обміну через SwapRouter02 з огортанням у мультівиклик для атомарності транзакції. Нативний ефір (ETH) автоматично мапнується на адресу обгорнутого ефіру (WETH) для внутрішніх операцій протоколу. Перед виконанням обміну застосунок перевіряє та за необхідності запитує дозвіл на використання ERC-20 токенів від імені користувача. Захист від проковзування ціни реалізовано через встановлення мінімально допустимої суми отримання з відхиленням 0,5%. Функцію заробітку реалізовано шляхом агрегації даних про пули ліквідності з семи провідних протоколів (Aave V3, Compound V3, Spark, Morpho Blue, Fluid, Euler, Venus) на п'яти

мережах (Ethereum, Base, Arbitrum, Optimism, Polygon) через API сервісу DefiLlama [6]. Фільтрація результатів відбувається за критеріями мінімальної ліквідності (TVL > 1 млн. дол. США) та реалістичної прибутковості (APY від 1% до 100%), що дозволяє виключити нерелевантні або підозрілі пули з вибірки.

Для обґрунтування переваг обраного підходу було проведено порівняльний аналіз архітектурних особливостей популярних існуючих платформ та запропонованого програмного рішення (таблиця 1).

Таблиця 1 – Порівняльна характеристика DeFi - інтерфейсів

Критерії порівняння	Uniswap	Zapper	1inch	Запропоноване рішення
Обмін токенів	Так	Ні	Так	Так
Агрегація доходності	Ні	Так	Ні	Так
Відстеження портфеля	Ні	Так	Ні	Так
Аналітика ринку	Так	Частково	Частково	Так

Детальний аналіз даних, наведених у таблиці 1, дозволяє виявити специфіку кожного з рішень. Інтерфейс Uniswap є еталонним у сфері обміну токенів, проте не надає можливостей для моніторингу пасивного доходу. Zapper орієнтований на відстеження активів та управління портфелем, проте не дозволяє безпосередньо виконувати обмін. Агрегатор 1inch фокусується на пошуку оптимального маршруту обміну, але не інтегрує функціонал заробітку. Запропоноване рішення об'єднує усі перелічені можливості в єдиному застосунку, усуваючи необхідність переходу між кількома сервісами при управлінні DeFi-портфелем.

Висновки

Розроблено програмну систему децентралізованого обміну токенів, що поєднує swap-обмін через Uniswap V3, агрегацію пропозицій пасивного доходу та відстеження портфеля. Використання стека Next.js + Wagmi + viem забезпечує типобезпечну взаємодію зі смарт-контрактами без власної серверної інфраструктури. Підтримка п'яти блокчейн-мереж та семи протоколів забезпечує широке охоплення ринку та практичну цінність для кінцевих користувачів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Buterin V. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform [Електронний ресурс]. – Режим доступу: <https://ethereum.org/en/whitepaper/>. – Дата звернення: 21.03.2026 р.
2. Xu J., Paruch K., Cousaert S., Feng Y. SoK: Decentralized Exchanges (DEX) with Automated Market Maker (AMM) Protocols // ACM Computing Surveys. 2023. Vol. 55, No. 11. P. 1–50.
3. Wood G. Ethereum: A Secure Decentralised Generalised Transaction Ledger. Ethereum Yellow Paper, 2014. 39 p.
4. Офіційна документація бібліотеки Wagmi [Електронний ресурс]. – Режим доступу: <https://wagmi.sh/>. – Дата звернення: 21.03.2026 р.
5. Adams H., Zinsmeister N., Salem M., Keefer R., Robinson D. Uniswap v3 Core [Електронний ресурс]. – Режим доступу: <https://uniswap.org/whitepaper-v3.pdf>. – Дата звернення: 21.03.2026 р.
6. Офіційна документація сервісу DefiLlama Yields API [Електронний ресурс]. – Режим доступу: <https://defillama.com/docs/api>. – Дата звернення: 21.03.2026 р.
7. Офіційна документація фреймворку Next.js [Електронний ресурс]. – Режим доступу: <https://nextjs.org/docs>. – Дата звернення: 21.03.2026 р.

Майданюк Володимир Павлович – кандидат технічних наук, доцент кафедри програмного забезпечення, Факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: maidaniuk2000@gmail.com.

Херсон Богдан Юрійович – студент групи 6ПІ-22б, Факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: bohdan.kherson4308@gmail.com.

Maidaniuk Volodymyr – Candidate of Technical Sciences, Associate Professor of the Department of Software Engineering, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia.

Kherson Bohdan – student of group 6PI-22b, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia.