

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра інформаційних технологій

«До захисту допущено»

В.о. завідувача кафедри

_____ Світлана ВАЩЕНКО

_____ 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня бакалавр

зі спеціальності 122 «Комп'ютерні науки»,

освітньо-професійної програми «Інформаційні технології проектування»

на тему: Вебдодаток підтримки функціонування децентралізованої біржі

Здобувача групи ІТ-01 Прокопчука Андрія Володимировича

Кваліфікаційна робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

_____ (підпис)

Андрій ПРОКОПЧУК

Керівник _____ доцент, к.т.н., доцент Анна НЕНЯ _____ (підпис)

Суми – 2024

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра інформаційних технологій
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

В. о. зав. кафедри ІТ

_____ Світлана ВАЩЕНКО

«__» _____ 2024 р.

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ

Прокончука Андрія Володимировича

1 Тема роботи Вебдодаток підтримки функціонування децентралізованої біржі
керівник роботи Неня Анна Вікторівна, к.т.н., доцент

затверджені наказом по університету від « 07 » 05 2024 р. №0482 VI

2 Строк подання студентом роботи « 26 » 05 2024 р.

3 Вхідні дані до роботи технічне завдання на розробку вебдодатку підтримки функціонування децентралізованої біржі

4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) аналіз предметної області, моделювання та проектування вебдодатку, розробка та тестування вебдодатку

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
актуальність, постановка задачі, аналіз додатків аналогів, порівняльна таблиця, функціональні вимоги, функціональне моделювання, архітектура додатку, демонстрація роботи вебдодатку, висновки

6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Планування робіт проекту	01.04.24 – 05.04.24	
2	Оформлення технічного завдання	06.04.24 – 12.04.24	
3	Проведення аналізу предметної області	13.04.24 – 19.04.24	
4	Проектування вебдодатку	20.04.24 – 26.04.24	
5	Структурно-функціональний аналіз	27.04.24 – 03.05.24	
6	Розробка вебдодатку	04.05.24 – 15.05.24	
7	Тестування вебдодатку	16.05.24 – 22.05.24	
8	Оформлення звіту	23.05.24 – 26.05.24	

Студент

(підпис)

Андрій ПРОКОПЧУК

Керівник роботи

(підпис)

к.т.н., доц. Анна НЕНЯ

АНОТАЦІЯ

Тема кваліфікаційної роботи бакалавра «Вебдодаток підтримки функціонування децентралізованої біржі».

Пояснювальна записка складається зі вступу, 3 розділів, висновків, списку використаних джерел із 23 найменувань, додатків. Загальний обсяг роботи – 66 сторінок, у тому числі 40 сторінок основного тексту, 3 сторінки списку використаних джерел, 23 сторінки додатків.

Актуальність роботи: існує необхідність у платформах, які дозволяють безпечно, прозоро та ефективно здійснювати операції з криптовалютами. Децентралізовані біржі виступають важливим інструментом у цьому процесі, оскільки вони забезпечують можливість торгівлі без посередників, що знижує ризик шахрайства та підвищує рівень приватності.

Мета роботи: створити інноваційний та неповторний продукт, орієнтований на користувача. Він повинен базуватися на технології блокчейн, для забезпечення функціональності та захисту користувачів.

Ключові слова: вебдодаток, блокчейн, криптовалюти, децентралізація, React, Node.js, Ethereum.

ЗМІСТ

ВСТУП	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	8
1.1 Огляд останніх досліджень	8
1.2 Аналіз існуючих продуктів – аналогів.....	10
1.3 Мета та задачі досліджень	16
1.4 Вибір засобів реалізації	17
2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ ВЕБДОДАТКУ.....	20
2.1 Моделювання.....	20
2.2 Проектування інформаційної системи.....	22
2.3 Схема зберігання даних.....	23
2.4 Архітектура вебдодатку	24
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕБДОДАТКУ	26
3.1 Програмна реалізація.....	26
3.2 Використання розробленого продукту	34
3.3 Тестування розробки	37
ВИСНОВКИ.....	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	41
ДОДАТОК А.....	44
ДОДАТОК Б.	54

ВСТУП

У сучасному світі фінансових технологій децентралізація відіграє ключову роль у розробці інноваційних рішень. Зокрема, децентралізовані біржі (DEX) стають все більш популярними в просторі криптовалют як альтернатива традиційним централізованим біржам. Вони дозволяють користувачам безпосередньо обмінюватися активами та забезпечують більшу прозорість, безпеку та анонімність.

Підтримка функціонування децентралізованої біржі включає в себе декілька ключових аспектів. По-перше, це створення інфраструктури, що забезпечить стабільну роботу біржі. По-друге, інтеграція з блокчейном, що дозволить безпечно зберігати дані про транзакції. По-третє, це розробка інтуїтивно зрозумілого інтерфейсу та надання освітніх матеріалів, що значно знизить поріг входження у світ криптовалют.

Основною проблемою, яку вирішує наш додаток, є забезпечення безпеки та анонімності користувачів. Основна мета – створити додаток децентралізованої біржі, який гарантуватиме безпеку та прозорість, використовуючи сучасні та ефективні інструменти для реалізації. Користувач повинен бути впевнений, що його дані не будуть зібрані та розсекречені, в цілях маркетингу або його особистого переслідування.

У цьому проєкті ми зосереджуємося на розробці децентралізованої web3 біржі з використанням сучасних інструментів і технологій, таких як Moralis Web3 API, 1Inch Aggregator, Wagmi, React і NodeJS.

Задачі проєкту включають:

- Аналіз стану досліджень в галузі розвитку децентралізованих бірж
- Огляд аналогів децентралізованих бірж
- Моделювання роботи вебдодатку підтримки функціонування децентралізованої біржі

- Розробка надійної і масштабованої структури, яка забезпечить спрощену обробку транзакцій та максимальну безпеку
- Створення інтуїтивно зрозумілого та зручного інтерфейсу для користувачів, що включає всі необхідні функції.
- Забезпечення зручної інтеграції з різними блокчейн-мережами для оптимізації швидкості транзакцій.
- Інтеграція смарт-контрактів для автоматизації та пришвидшення процесів.

Об'єкт дослідження – децентралізовані криптовалютні біржі

Предмет дослідження – методи та технології розробки та підтримки функціонування децентралізованих бірж

Практична значимість дослідження полягає у розробці вебдодатку підтримки функціонування децентралізованої біржі, який забезпечить високу безпеку, прозорість та анонімність для користувачів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень

Децентралізовані біржі поступово стають важливим етапом у розвитку фінансових технологій. Оскільки зараз блокчейн технології стрімко розвиваються, а інтерес до криптовалют збільшується, то ці біржі стають важливими інструментами. У цьому розділі ми розглянемо значення та особливості розвитку децентралізованих бірж, зосередимось на технічній основі, на якій вони базуються [1].

Потреба у безпечному, прозорому та ефективному механізмі обміну цифровими валютами є причиною важливості децентралізованих бірж. Традиційні фінансові установи мають певні проблеми пов'язані з централізацією. Ними є надто високі комісії, обмежений доступ для певних верст населення та потенційні ризики для безпеки користувачів та їх коштів. Децентралізовані біржі з легкістю вирішують всі ці та інші проблеми, саме тому вони є дуже перспективними [2].

Іншим важливим фактором, що пояснює швидкий розвиток попиту на децентралізовані біржі, є їх здатність забезпечувати користувачам високий рівень безпеки та повну анонімність. Поточні дослідження підтверджують, що біржі можуть забезпечити надійний захист даних і прозорість торгівлі. Це стало можливим завдяки блокчейн технологіям, таким як Ethereum, які забезпечують надійну та швидку обробку транзакцій [3].

Інноваційні блокчейн технології, такі як смарт-контракти, відіграють ключову роль у розвитку децентралізованих бірж. Смарт контракти можна використовувати для автоматизації та підвищення ефективності фінансових процесів. Різні блокчейни забезпечують розподілене зберігання даних і обробку транзакцій без централізованого посередника, забезпечуючи високий рівень надійності та незалежності [4].

Найпопулярнішою технологією, на якій базується більшість децентралізованих бірж, є Ethereum. Ethereum це лідируюча блокчейн-мережа, на якій вперше було запроваджено смарт-контракти, завдяки яким була утворена можливість розробляти dApp (децентралізовані застосунки). Основною концепцією цієї мережі є те, що центральних органів влади для забезпечення функціонування мережі, немає. Отже, користувачі самі контролюють прийняття рішень в інтересах всієї спільноти. В основі Ethereum лежить блокчейн, в який записуються всі дані про транзакції, там же вони й зберігаються. Функціонування блокчейну забезпечують ноди (вузли). Нодами є розподілена система ПК або серверів, які є основним джерелом обчислювальної потужності для Ethereum. Вони виконують багато важливих завдань, завдяки ПО встановленому на них. Такими завданнями є збереження та запис даних про транзакції та смарт-контракти. Запустити власну ноду (вузол) може будь яка людина, з достатнім рівнем знань та кількості ресурсів. Чим більше буде різноманітних вузлів, тим складніше буде отримати контроль над більшою частиною мережі. Що унеможливує атаку на блокчейн задля отримання власної вигоди (блокування певних транзакцій, викрадення коштів, маніпулювання ціною). Задля написання смарт-контрактів для Ethereum, використовується мова програмування Solidity. Нативним токеном мережі є ЕТН. Нативний позначає те, що саме ця валюта використовується для виконання всіх операцій блокчейні. Для будь якої транзакції або іншої дії потрібно заплатити комісію (газ) в ЕТН. Кількість газу варується від складності операції та навантаження на мережу, тобто чим більше газу треба використати, тим дорожче буде транзакція. За допомогою Ethereum можна створювати інші сумісні токени та децентралізовані застосунки, такі як децентралізовані біржі [5].

У цілому, децентралізовані біржі Web3 відображають сучасні тенденції у фінансовій сфері та стають важливим кроком у розвитку фінансової індустрії. Вони базуються на перспективних технологіях, які демонструють потенціал для створення більш доступних, ефективних та прозорих фінансових послуг. Завдяки цьому кількість людей пов'язаних з криптовалютами з часом буде збільшуватись, що рано чи

пізно призведе до тотальної фінансової революції і назавжди змінить наше бачення економіки [6].

1.2 Аналіз існуючих продуктів – аналогів

Була проаналізована певна кількість додатків, для того щоб сформувати представлення про сучасні тренди та побачити функціонал конкурентів. Цими сайтами є Uniswap, SushiSwap та PancakeSwap. Треба звернути увагу на дизайн цих додатків, їх комісії та безпечність.

Сайт UniSwap [7] має мінімалістичний та інтуїтивно зрозумілий дизайн, який акцентує увагу на основних функціях обміну та додавання ліквідності, з простим доступом до важливих інформаційних розділів (рис. 1.1-1.2).

- Один з найпопулярніших DEX на Ethereum (Ethereum — це універсальна розподілена система смарт-контрактів на базі блокчейн-технології.) [8].
- Простота використання та велика кількість ліквідності роблять його привабливим для широкого кола користувачів.

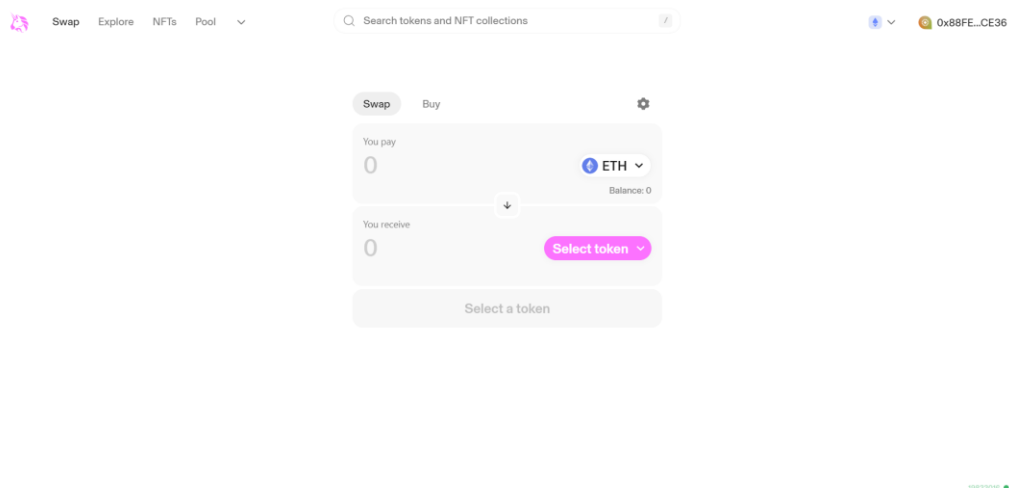


Рисунок 1.1 – Сторінка обміну tokenів UniSwap

Джерело: [7]

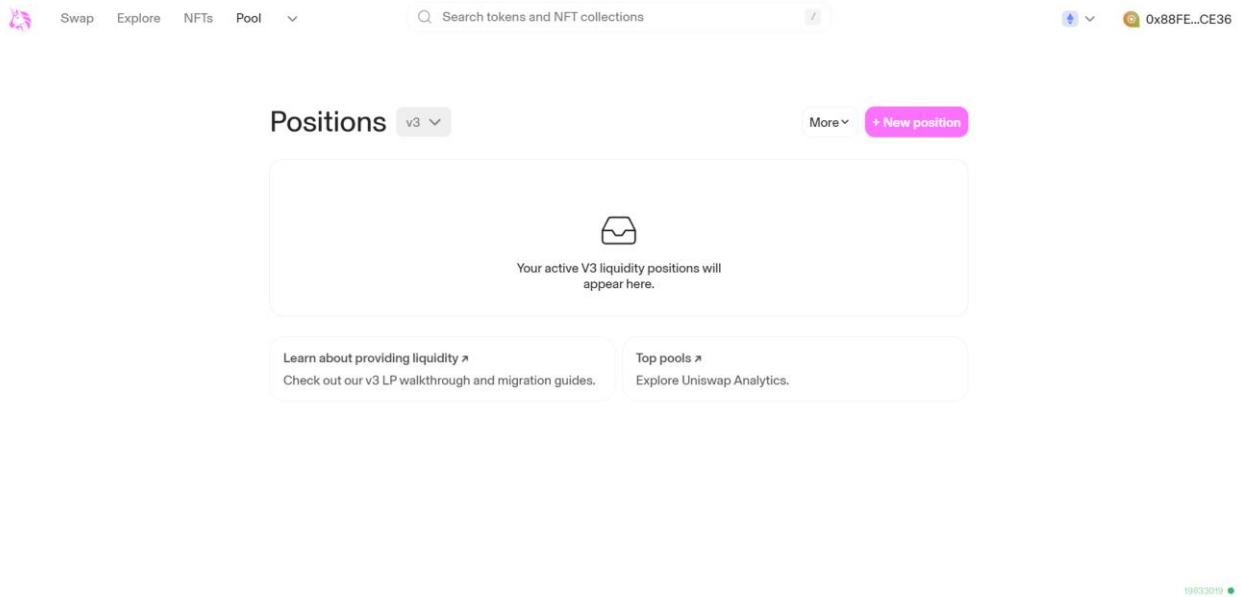


Рисунок 1.2 – Сторінка додавання ліквідності UniSwap

Джерело: [7]

SushiSwap [9] використовує більш насичений візуальний стиль з акцентом на японську тематику. Пропонує деталізоване меню з різноманітними опціями торгівлі, ліквідності та фармінгу (рис. 1.3-1.5).

- Початково був форком (Форк — це подія, коли одна частина блокчейну розгалужується на іншу, а вихідний код копіюється та змінюється для створення окремого блокчейну [10].) UniSwap, але з часом розвинув унікальні функції.
- Особливість полягає у можливості фармінгу ліквідності та отриманні SUSHI токенів.

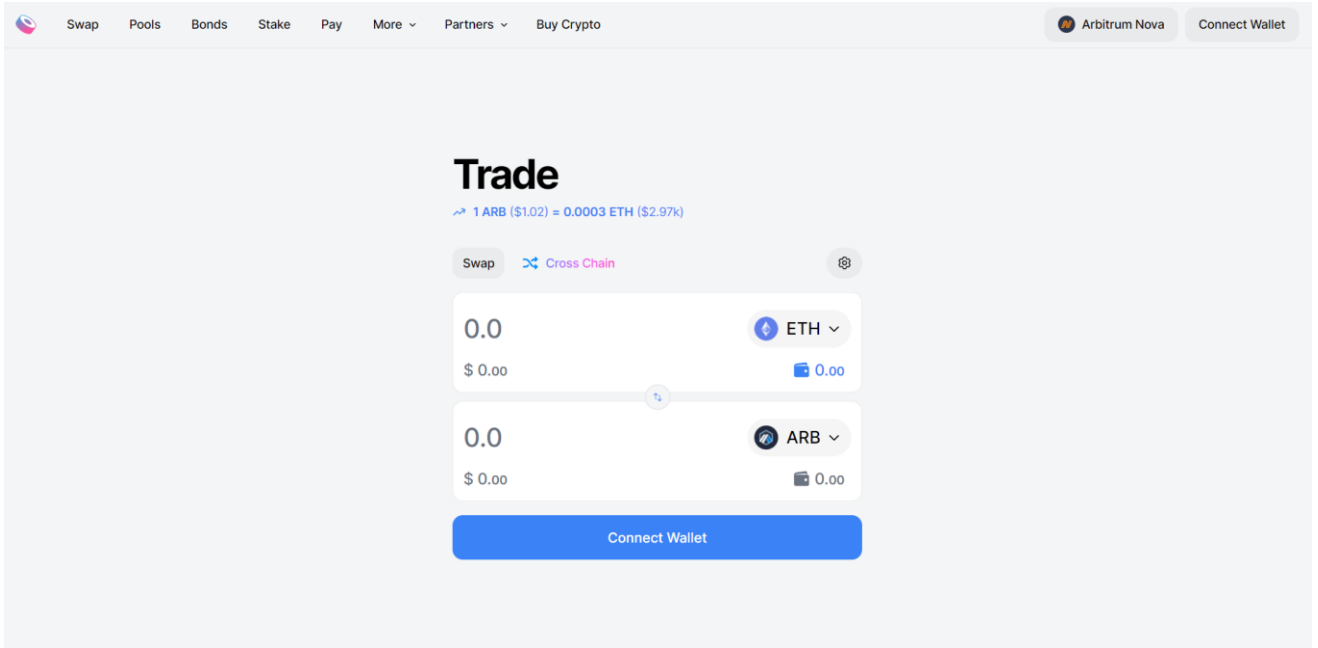


Рисунок 1.3 – Сторінка обміну токенів SushiSwap

Джерело: [9]

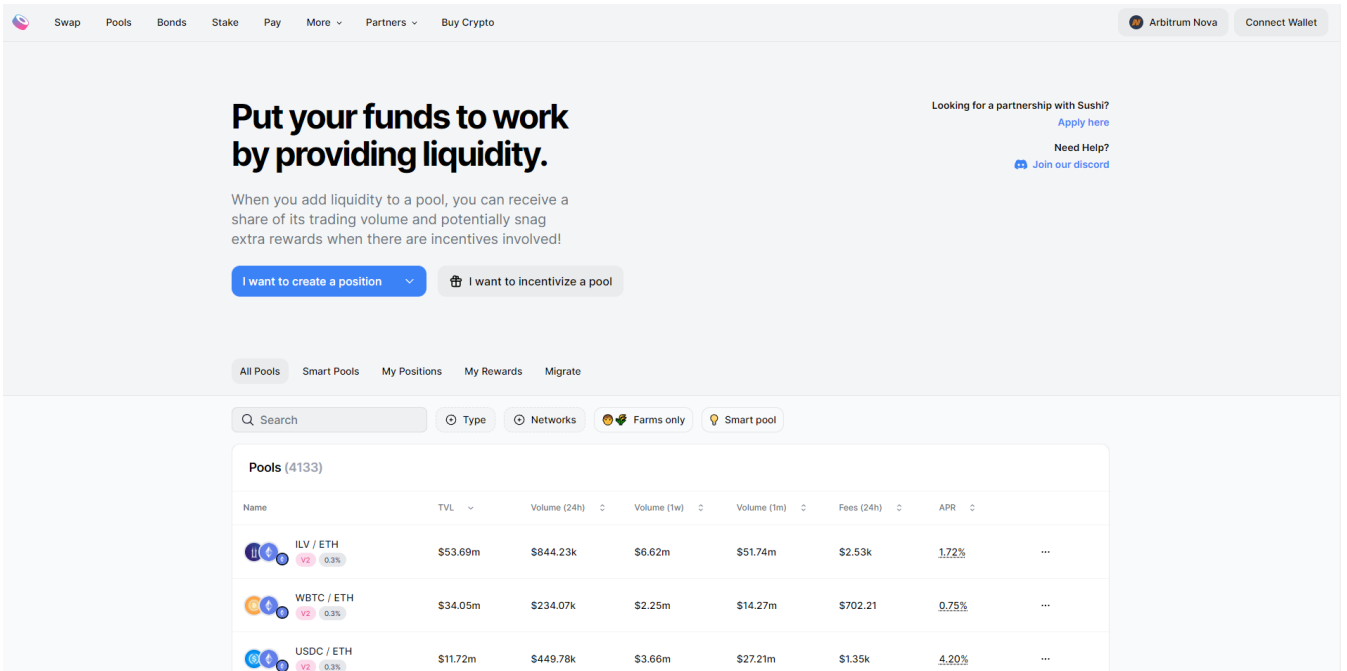


Рисунок 1.4 – Сторінка додавання ліквідності SushiSwap

Джерело: [9]

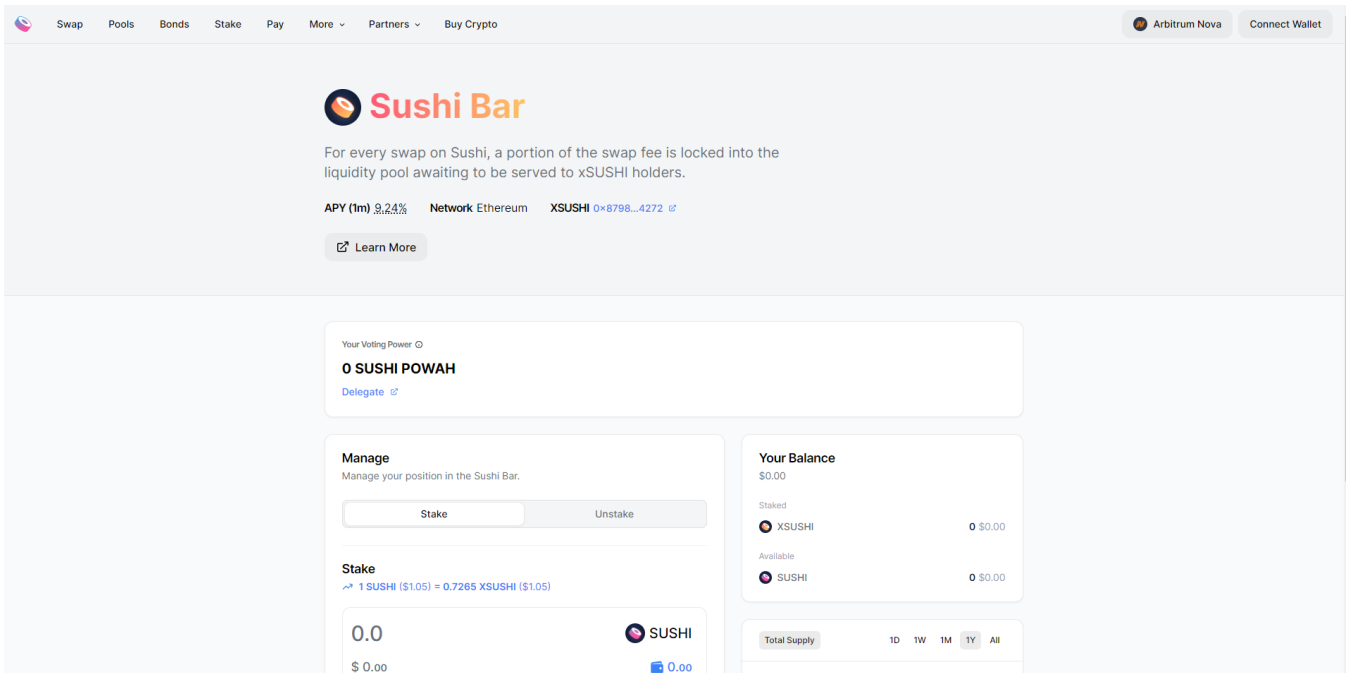


Рисунок 1.5 – Сторінка фармінгу tokenів SushiSwap

Джерело: [9]

PancakeSwap [11] має веселий та мультяшний дизайн функцій обміну, фармінгу tokenів та ігрових елементів (рис. 1.3).

- Популярний на Binance Smart Chain, пропонує нижчі комісії порівняно з Ethereum-базованими DEX.
- Особливістю є висока швидкість транзакцій і фармінг CAKE tokenів.

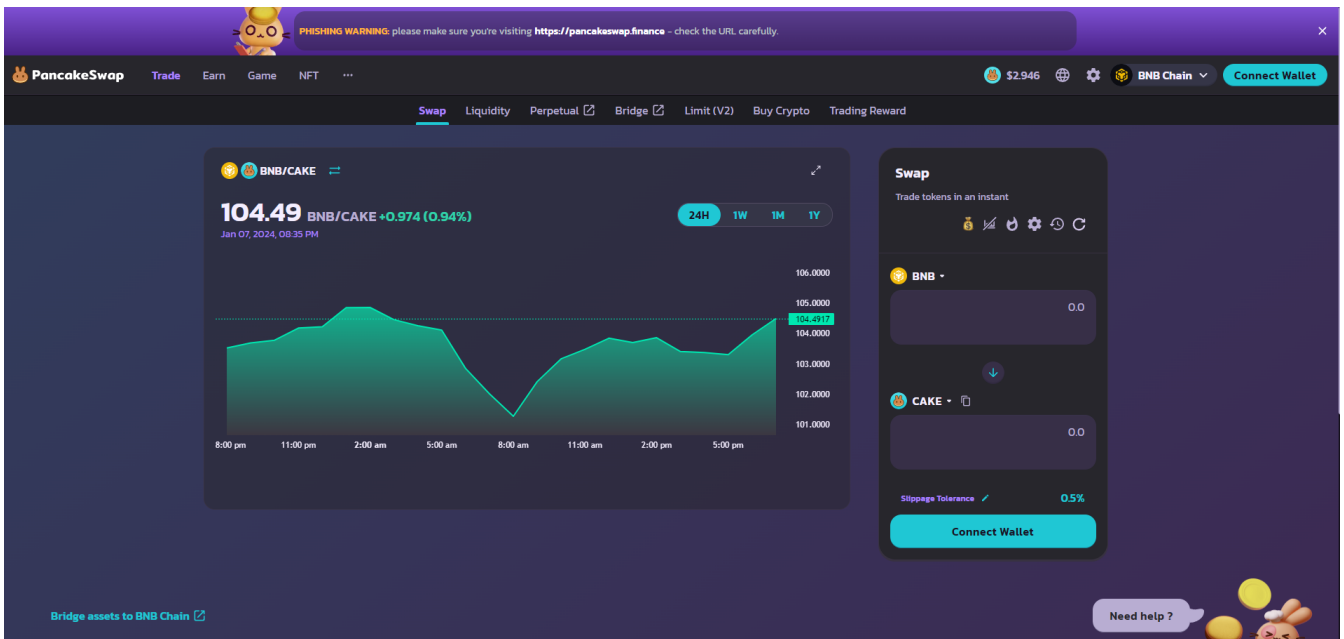


Рисунок 1.6 – Сторінка обміну tokenів PancakeSwap

Джерело: [11]

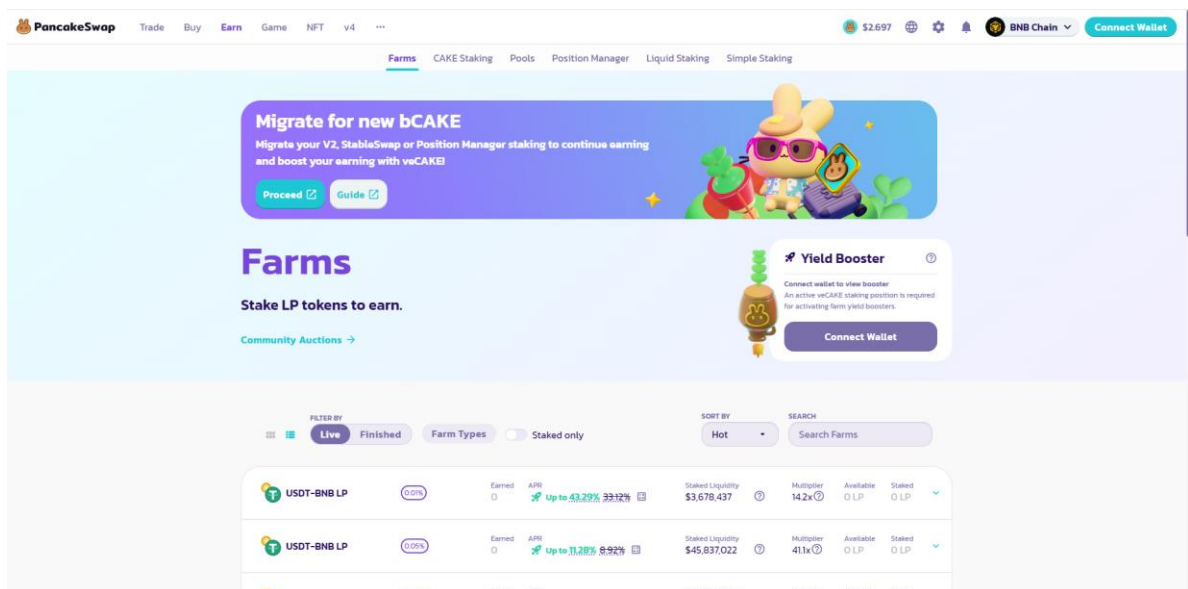


Рисунок 1.7 – Сторінка фармінгу tokenів CAKE PancakeSwap

Джерело: [11]

Кожна з цих DEX має унікальний підхід до дизайну та структури свого сайту, що відображає їхню індивідуальність та цільову аудиторію.

Таблиця 1.1 – Порівняльна таблиця характеристик DEX

Характеристика	UniSwap	SushiSwap	PancakeSwap	LightSwap
Тип біржі	Автоматизований маркет-мейкер (АММ)	Автоматизований маркет-мейкер (АММ)	Автоматизований маркет-мейкер (АММ)	Автоматизований маркет-мейкер (АММ)
Блокчейн	Ethereum	Ethereum та інші	Binance Smart Chain	Ethereum
Токени	ERC-20 та інші сумісні	ERC-20, BEP-20 та інші	BEP-20 та інші	ERC-20 та інші сумісні
Особливості	Висока ліквідність, простий інтерфейс	Фармінг ліквідності, SUSHI токени як нагорода	Низькі комісії, фармінг CAKE tokenів	Низькі комісії, повна анонімність та відсутність обмежень
Комісії	Змінні, залежать від обраної пари tokenів	Змінні, залежать від обраної пари tokenів	Змінні, але зазвичай нижчі за інші DEX	Змінні, залежать від обраної пари tokenів
Безпека	Висока, але залежить від смарт-контрактів	Висока, але залежить від смарт-контрактів	Висока, з додатковими функціями безпеки	Висока, гарантується повна анонімність

Джерело: розроблено автором

Аналізуючи існуючі аналоги децентралізованих бірж (DEX), як-от UniSwap, SushiSwap, і PancakeSwap, можна виділити кілька ключових аспектів, на які потрібно звернути увагу при розробці власної DEX.

Для приваблення та утримання користувачів, особливо тих, хто використовує DEX вперше, дизайн інтерфейсу повинен бути простим та зрозумілим.

Для отримання довіри серед користувачів, необхідно забезпечити високий рівень безпеки. Для цього вебдодаток потребує захист від шахрайства, хакерських атак та зломів смарт-контрактів.

Можливість торгувати різними типами токенів може значно розширити аудиторію біржі. Слід розглянути підтримку різних стандартів токенів, таких як ERC-20, BEP-20 тощо.

При розробці DEX важливо знайти баланс між заробітком коштів з платформи та конкурентоспроможними комісіями.

1.3 Мета та задачі досліджень

Метою цього дослідження є розробка децентралізованої web3 біржі, яка забезпечує безпечну, прозору та ефективну торгівлю криптовалютами. Основна мета – створити надійний та затребуваний продукт, орієнтований на користувача. Він повинен базуватися на технології блокчейн, для забезпечення функціональності та захисту користувачів.

Основні вимоги до створюваного програмного продукту і технологій є наступними:

- Реалізувати інтеграцію з Moralis Web3 API для забезпечення взаємодії з блокчейн мережами задля обробки транзакцій.
- Використати 1Inch Aggregator для оптимізації маршрутів обміну та забезпечення кращих комісій.
- Інтегрувати Wagmi для авторизації користувачів та забезпечення безпеки
- Створити зручний та інтуїтивно зрозумілий фронтенд за допомогою React.
- Розробити надійний та масштабований бекенд на NodeJS.

Для досягнення мети проекту необхідно виконати наступні задачі:

- Визначити актуальність дослідження, цільову аудиторію та дослідити предметну область.
- Провести аналіз існуючих аналогів децентралізованих бірж і виділити їх переваги та недоліки.
- Розробити фронтенд додатку за допомогою HTML, CSS, Javascript та бібліотеки React
- Розробити серверну частину вебдодатку за допомогою Node.js
- Інтегрувати авторизацію користувача завдяки бібліотеці Wagmi
- Задля отримання актуальної інформації про курси валют, реалізувати отримання даних через Moralis Web3 API
- Реалізувати інтеграцію з блокчейном Ethereum та відправку транзакції через 1inch API

1.4 Вибір засобів реалізації

Реалізація вебдодатку підтримки функціональної web3 біржі передбачає використання певних технологій. Нижче наведений детальний огляд кожної з них та обґрунтування вибору.

Frontend технології:

- HTML, CSS, JavaScript: Фронтенд технології спрямовані на створення інтерфейсу, який в подальшому буде представлений користувачу. HTML, CSS та JavaScript є основними технологіями для розробки веб-інтерфейсу [12].
- React JS – це бібліотека JavaScript, яку використовують для розробки інтерфейсу. Цей фреймворк надає доступ до швидкого, високопродуктивного створення застосунків, з можливістю масштабування. Його використання дозволить створити динамічні та ефективні компоненти, задля зручної роботи з вебдодатком [13].

Backend технології:

- Node.js – це середовище виконання JavaScript. Платформа була створена для легкого масштабування швидких мережеских застосунків. Для створення динамічних вебсторінок використовують серверні сценарії, які виконують запити на стороні сервера, чим зменшують навантаження на ресурси ПК та пришвидшують завантаження. Завдяки парадигмі “JavaScript everywhere” не треба постійно перемикатись між серверними мовами програмування та зовнішньою розробкою, що значно оптимізує код. Так як Node.js може обробляти кілька одночасних підключень, це дозволяє легше масштабувати продукт [14].

Blockchain технології:

- Moralis Web3 API: Moralis Web3 API - це інструмент для розробки децентралізованих додатків, який забезпечує швидку та просту інтеграцію з різноманітними блокчейн-мережами [15].
- 1Inch API Aggregator: 1Inch Aggregator - це передовий сервіс, який автоматизує обмін криптовалютами між різними блокчейн-мережами. Його використання дозволить забезпечити оптимальні комісії та найкращу швидкість для обміну криптовалютами для користувачів біржі [16].
- Wagmi – це web3 JavaScript бібліотека, яка дозволить під’єднати гаманець до вебдодатку нашої біржі. Wagmi забезпечує відмінний досвід розробки завдяки модульним та комбінованим API, автоматичній безпеці типів, а також детальній документації [17].

Безпека у вебдодатку підтримки функціонування децентралізованої біржі забезпечується завдяки основним принципам технології блокчейн. У блокчейні дані не зберігаються на одному центральному сервері, а розподілені між нодами (вузлами). Завдяки цьому, навіть якщо певна кількість вузлів вийдуть з ладу, система продовжить свою роботу. Цей принцип називається децентралізацією.

Коли транзакція опиняється в блокчейні, то її вже не можна видалити чи ввести до неї зміни, саме це унеможливорює її підробку шахраями. Також блокчейн

використовує хешування (алгоритм створення певного виводу фіксованого розміру із вхідних даних за допомогою складних математичних формул, під назвою хеш-функції)[18]. Кожна транзакція підписується цифровим підписом, що гарантує власнику однобічний контроль над нею.

2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ ВЕБДОДАТКУ

2.1 Моделювання

IDEF0 (Integration Definition for Function Modeling) – метод моделювання, який використовується для аналізу, розробки та документування функцій та процесів у системах [19]. Основними компонентами цих діаграм є функціональні блоки та стрілки. Функціональні блоки представляють окремі функції або процеси, стрілки в свою чергу, вказують на потоки інформації [20].

Функціональне моделювання процесу підтримки діяльності вебдодатку децентралізованої біржі зображено на рисунку 2.1. В цьому випадку вхідними даними є дані авторизації користувача та введені ним дані для обміну. Правилами та обмеженнями згідно до яких виконується процес є функціональні можливості, дані з Moralis Web3 API та дані з 1inch API. Незмінними ресурсами, які необхідні для підтримки процесу є технічне забезпечення, вебдодаток LightSwap та користувач. На виході ми отримуємо виконану транзакцію обміну.

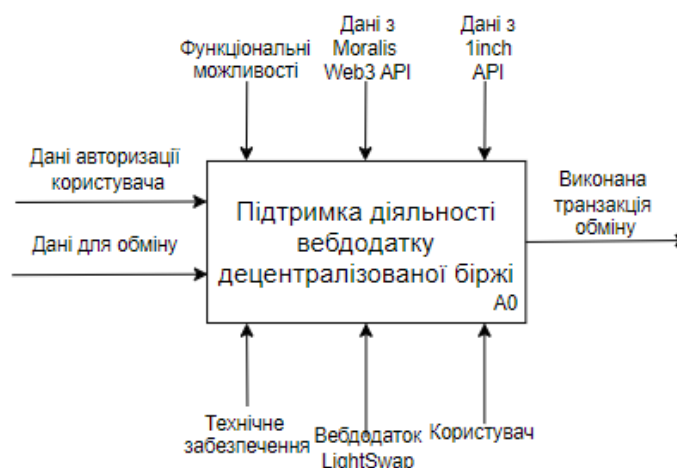


Рисунок 2.1 – IDEF0 діаграма підтримки діяльності вебдодатку децентралізованої біржі

Джерело: розроблено автором

Діаграма декомпозиції, яка детальніше описує попередній процес підтримки діяльності вебдодатку децентралізованої біржі, зображена на рисунку 2.2.

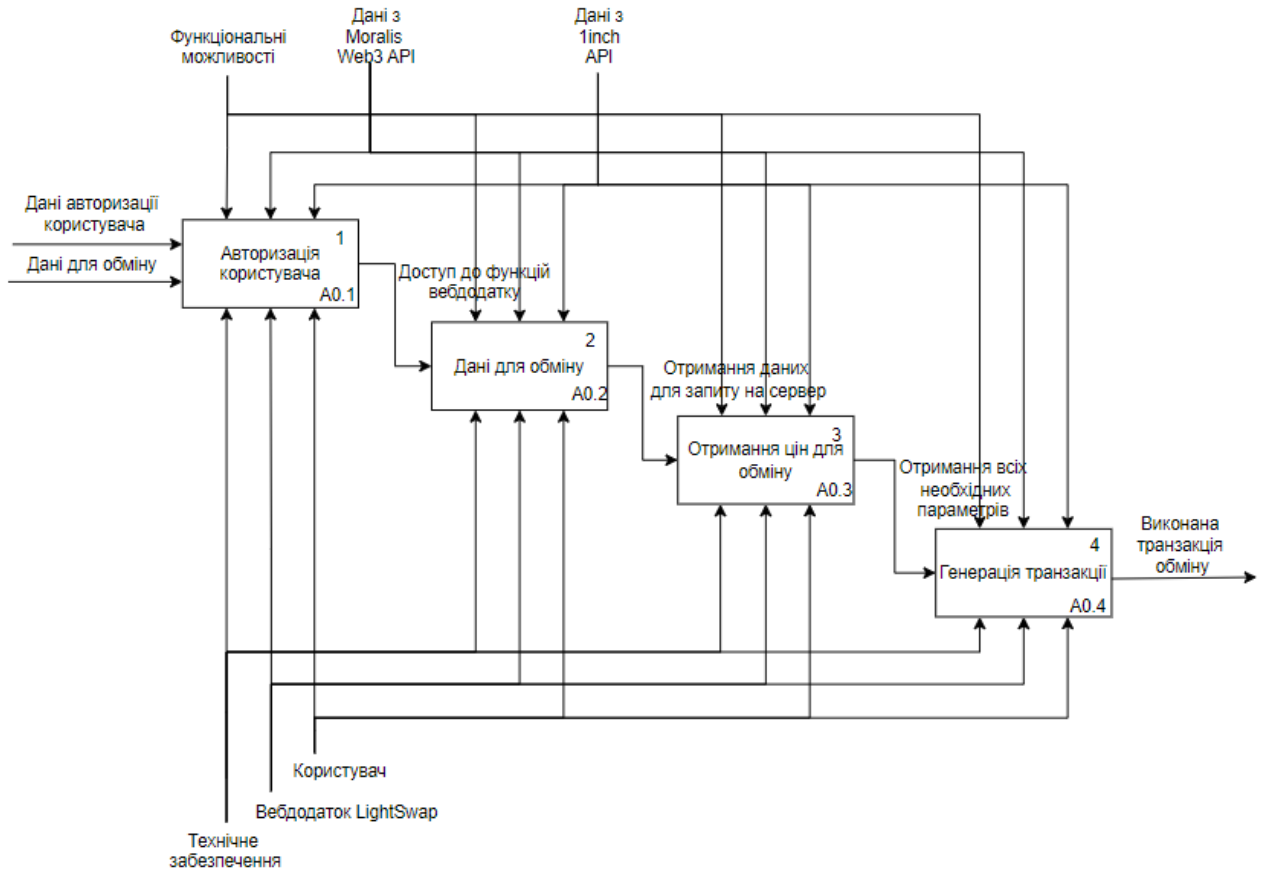


Рисунок 2.2 – Декомпозиція IDEF0

Джерело: розроблено автором

Діаграма декомпозиції процесу підтримки діяльності вебдодатку децентралізованої біржі складається з чотирьох підпроцесів: авторизація користувача, введення параметрів обміну, отримання цін для обміну та генерації транзакції.

2.2 Проектування інформаційної системи

Діаграма варіантів використання (use-case діаграма) — це тип UML (мова уніфікованого моделювання) діаграми, який зображує взаємодію між акторами та інформаційною системою задля виконання певних задач. Вона надає високорівневий огляд функціональності системи, показуючи різні способи взаємодії користувачів з нею [21]. Діаграма варіантів використання для вебдодатку підтримки діяльності децентралізованої біржі зображена на рисунку 2.3.

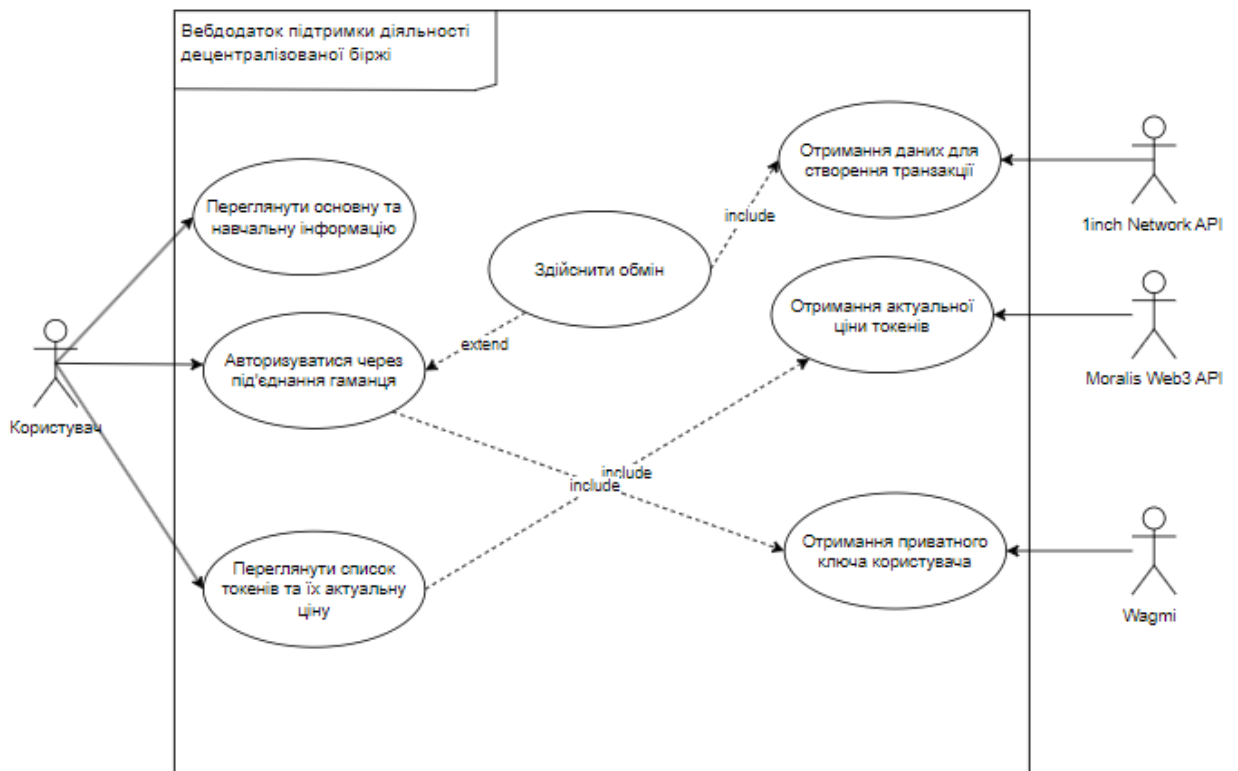


Рисунок 2.3 – Use-case діаграма

Джерело: розроблено автором

2.3 Схема зберігання даних

Блокчейн складається з послідовних блоків, кожен з яких містить хеш-код, обчислений на основі попереднього блоку, та додаткову інформацію про транзакції, операції, підписи та інші дані. Поняття блокчейну можна пояснити як реєстр записів, в який можна лише додавати інформацію, без можливості зміни чи видалення даних з попередніх блоків. Це відбувається завдяки розподіленню блокчейну по вузлах (нодах)[22].

Приклад структури та організації блоків зображений на рисунку 2.4.

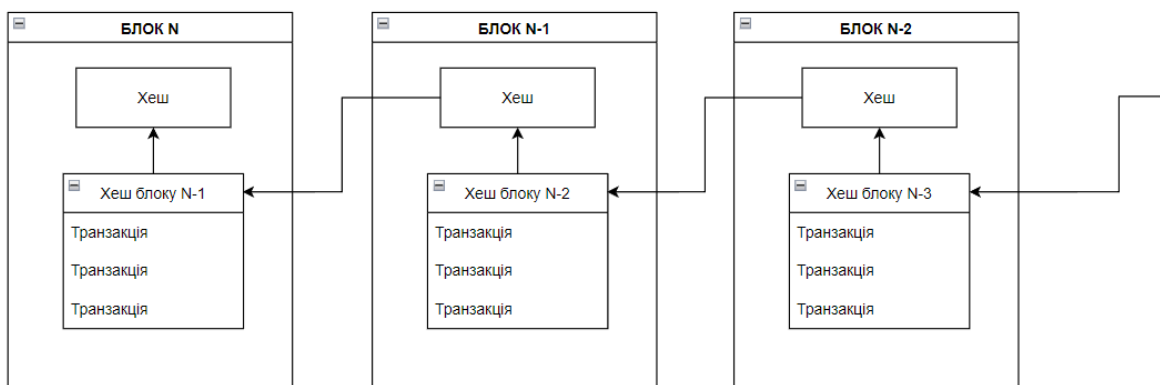


Рисунок 2.4 – Структура та організація блоків

Джерело: розроблено автором

Саме так вся інформація з вебдодатку підтримки діяльності децентралізованої біржі зберігається в блокчейні Ethereum. Маючи хеш нашої вихідної транзакції, можна побачити всі її параметри завдяки сайтам-провідникам. Такими параметрами виступатимуть: дата та час транзакції, адреса відправника, контракти та кількість обмінаних токенів.

2.4 Архітектура вебдодатку

Архітектура вебдодатку підтримки діяльності децентралізованої біржі LightSwap зображено на рисунку 2.5.

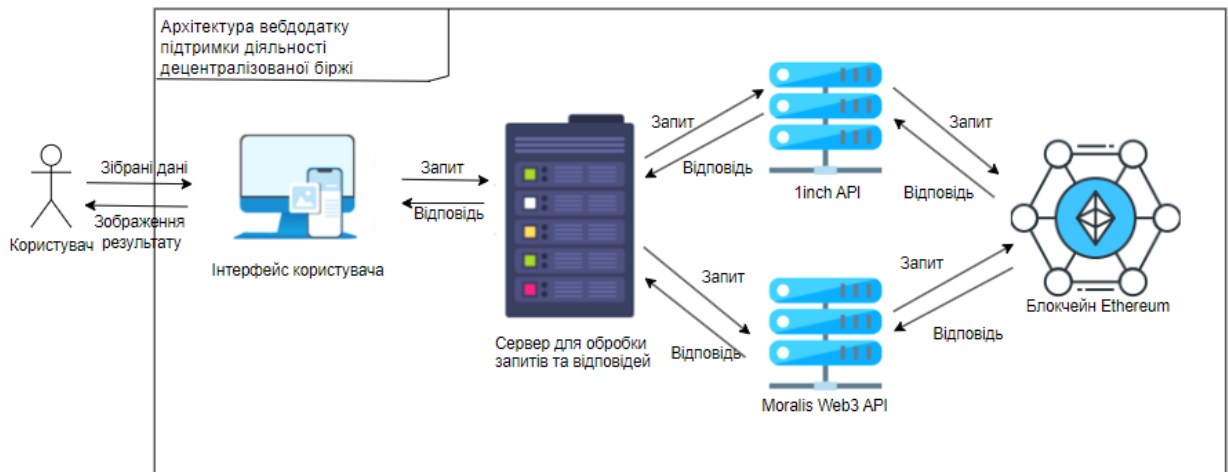


Рисунок 2.5 – Діаграма архітектури вебдодатку

Джерело: розроблено автором

Схема роботи вебдодатку (рис 2.6) складається з декількох етапів, які описують послідовність подій. Нижче наведено опис кожного з цих етапів:

- Актуальна інформація про ціни токенів надсилається до вебдодатку LightSwap за допомогою 1inch API.
- Після авторизації через розширення гаманця в вебдодатку LightSwap користувач обирає токени для обміну та вводить їх кількість.
- Після введення параметрів транзакція відправляється до вузлів за допомогою 1inch API.
- Вузли перевіряють правильність транзакції, а також статус користувача. Сюди входить перевірка підписів, балансу та інших важливих аспектів.
- Після успішної перевірки транзакція верифікується і всі необхідні параметри передаються у смарт-контракт.

- Перевірена транзакція додається до нового блоку, який назавжди включається до існуючої послідовності блоків у блокчейні. Можливість внесення змін до попередніх блоків відсутня, що забезпечує незмінність даних.
- На цьому етапі транзакція вважається виконаною, всі зміни відбулися і можуть бути відслідковані завдяки пошуку хешу на спеціальних сайтах-провідниках.

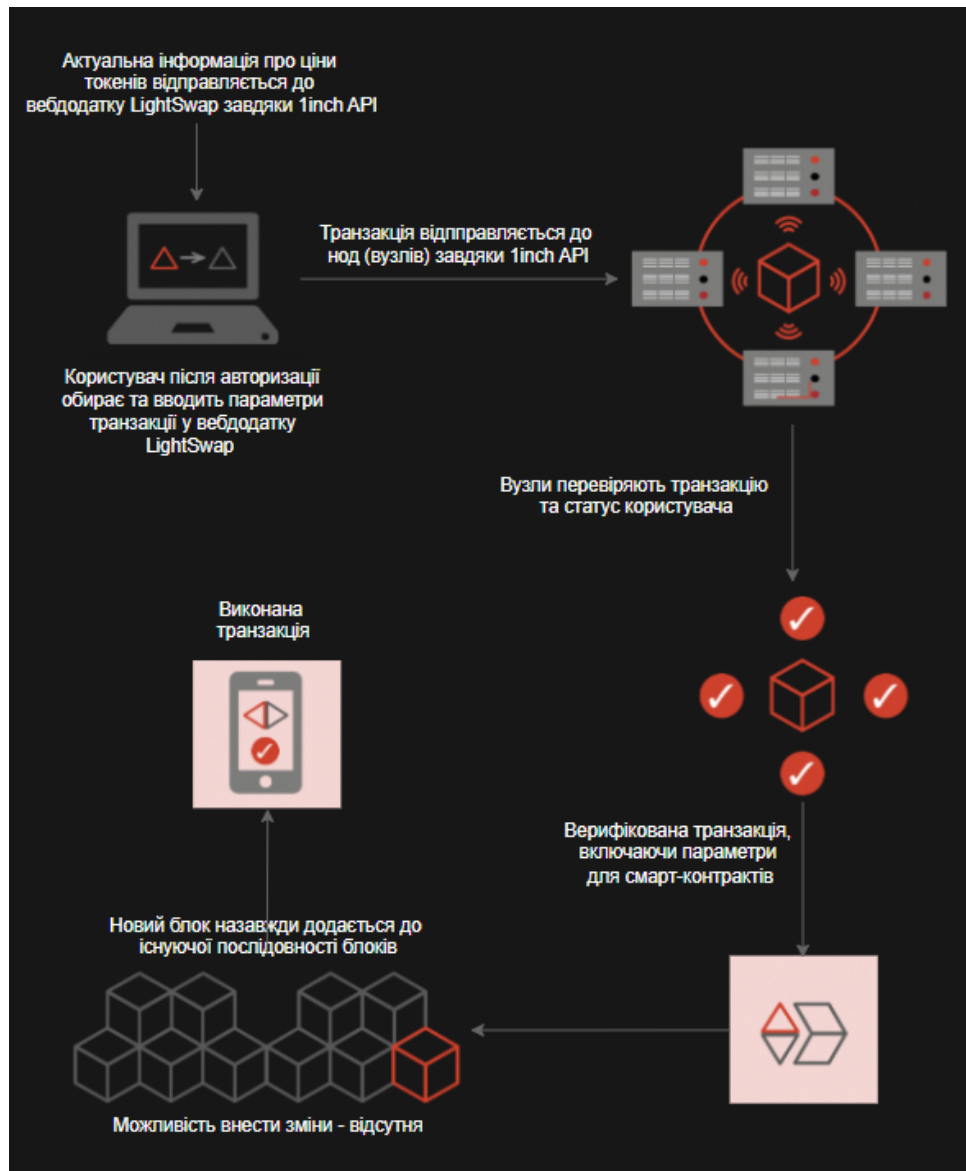


Рисунок 2.6 – Схема роботи вебдодатку

Джерело: розроблено автором

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ВЕБДОДАТКУ

3.1 Програмна реалізація

3.1.1 Розробка фронтенду

Фронтенд вебдодатку підтримки діяльності децентралізованої біржі був розроблений за допомогою HTML, CSS, JavaScript та React. Компонент App.js – це типовий файл конфігурації та запуску вебдодатків у середовищі Node.js. Він є основним файлом цього проекту, в якому міститься імпорт бібліотек та модулів. Маршрутизація реалізована завдяки API React Router (рис. 3.1), який використовує динамічну маршрутизацію (здійснюється під час рендерингу програми). Це дозволяє створити односторінковий вебдодаток з навігацією, без оновлення сторінки при переміщенні по сайту [23].

```
function App() {
  const {address, isConnected} = useAccount();
  const {connect} = useConnect({
    connector: new MetaMaskConnector(),
  });

  return (
    <div className="App">
      <Header connect={connect} isConnected={isConnected} address={address} />
      <div className="mainWindow">
        <Routes>
          <Route path="/main" element={<Main />} />
          <Route path="/" element={<Swap isConnected={isConnected} address={address} />} />
          <Route path="/tokens" element={<Tokens />} />
          <Route path="/info" element={<Info />} />
        </Routes>
      </div>
    </div>
  )
}
```

Рисунок 3.1 – Реалізація маршрутизації

Джерело: розроблено автором

Функція «Header» - це компонент, який є заголовком сторінки з навігаційними елементами. Кнопка під'єднання гаманця змінюється після авторизації, показуючи перші та останні чотири символи адреси користувача. Код реалізації функції зображено на рисунку 3.2.

```
function Header(props) {
  const {address, isConnected, connect} = props;

  return (
    <header>
      <div className='leftH'>
        <Link to='/main' className='link'>
          <img src={Logo} alt="logo" className='logo' />
        </Link>

        <Link to='/' className='link'>
          <div className='hItem'>Swap</div>
        </Link>

        <Link to='/tokens' className='link'>
          <div className='hItem'>Tokens</div>
        </Link>

        <Link to='/info' className='link'>
          <div className='hItem'>Info</div>
        </Link>
      </div>
      <div className='rightH'>
        <div className='hItem'><img src={Eth} alt='eth' className='eth' />Ethereum</div>
        <div className='connectButton' onClick={connect} >
          {isConnected ? (address.slice(0, 4)+ "..."+ address.slice(38)) : "Connect"}
        </div>
      </div>
    </header>
  )
}
```

Рисунок 3.2 – Функція «Header»

Джерело: розроблено автором

Обмін реалізовано в файлі Swar.js, в якому міститься компонент Swar, та інші вкладені допоміжні функції. Він забезпечує функціональність для обміну токенів, налаштування параметрів транзакції, отримання цін та відображення інтерфейсу. Нижче будуть описані ключові аспекти цього файлу. Функція Swar (рис.3.3)

використовує хуки `useState` для керування різними станами, наприклад, кількість токенів, обрана пара токенів, ковзання ціни та інші.

```
const {address, isConnected} = props;
const [slippage, setSlippage] = useState(1.5);
const [tokenOneAmount, setTokenOneAmount] = useState(null);
const [tokenTwoAmount, setTokenTwoAmount] = useState(null);
const [tokenOne, setTokenOne] = useState(tokenList[0]);
const [tokenTwo, setTokenTwo] = useState(tokenList[1]);
const [isOpen, setIsOpen] = useState(false);
const [changeToken, setChangeToken] = useState(1);
const [prices, setPrices] = useState(null);
const [messageApi, contextHolder] = message.useMessage();
const [txDetails, setTxDetails] = useState({
  to: null,
  data: null,
  value: null,
});
```

Рисунок 3.3 – Функція `Swap`

Джерело: розроблено автором

Функції для обробки подій зображені на рисунках 3.4 – 3.8.

```
function handleSlippageChange(e){
  setSlippage(e.target.value);
}
```

Рисунок 3.4 – Функція `handleSlippageChange` – змінює значення ковзання ціни

Джерело: розроблено автором

```
function changeAmount(e){
  setTokenOneAmount(e.target.value);
  if(e.target.value && prices){
    setTokenTwoAmount((e.target.value*prices.ratio).toFixed(3))
  }else{
    setTokenTwoAmount(null);
  }
}
```

Рисунок 3.5 – Функція `changeAmount`: змінює кількість першого токена та обчислює кількість другого токена.

Джерело: розроблено автором

```
function switchTokens(){
  setTokenOneAmount(null);
  setTokenTwoAmount(null);
  setPrices(null);
  const one = tokenOne;
  const two = tokenTwo;
  setTokenOne(two);
  setTokenTwo(one);
  fetchPrices(two.address, one.address);
}
```

Рисунок 3.6 – Функція switchTokens – змінює місцями токени

Джерело: розроблено автором

```
function openModal(asset){
  setChangeToken(asset);
  setIsOpen(true);
}
```

Рисунок 3.7 – Функція openModal – відкриває модальне вікно зі списком
токенів для вибору

Джерело: розроблено автором

```
function modifyToken(i){
  setTokenOneAmount(null);
  setTokenTwoAmount(null);
  setPrices(null);
  if (changeToken === 1){
    setTokenOne(tokenList[i]);
    fetchPrices(tokenList[i].address, tokenTwo.address);
  } else{
    setTokenTwo(tokenList[i]);
    fetchPrices(tokenOne.address, tokenList[i].address);
  }
  setIsOpen(false);
}
```

Рисунок 3.8 – Функція modifyToken – змінює існуючий токен на обраний зі
списку

Джерело: розроблено автором

Функція `fetchPrices` (рис. 3.9) отримує поточні ціни для обраних токенів через взаємодію з сервером.

```

async function getPrices(one, two){
    const res = await axios.get(`http://localhost:3001/tokenPrice`, {
        params: {addressOne: one, addressTwo: two}
    })
    setPrices(res.data);
}

```

Рисунок 3.9 - Функція `getPrices`

Джерело: розроблено автором

Функція `fSwap` (рис. 3.10) передає параметри авторизації розробника до `1inch` API. Потім, за допомогою API перевіряє дозволи на витрати коштів, якщо дозволу немає – генерує транзакцію збільшення ліміту, в іншому випадку – передає параметри для смарт-контракту для обміну.

```

async function makeSwap(){
    const allowance = await axios.get(`http://localhost:3001?url=https://api.1inch.dev/swap/v6.0/1/approve/allowance`, {
        headers: {
            Authorization: "Bearer 7QhMGYk8aE7ZXLeitTio94W8G1I0ix6",
            accept: "application/json"
        },
        params: {
            tokenAddress: tokenOne.address,
            walletAddress: address
        }
    });

    if(allowance.data.allowance === "0"){
        const approve = await axios.get(`http://localhost:3001?url=https://api.1inch.dev/swap/v6.0/1/approve/transaction?tokenAddress=${tokenOne.address}`)
        setTxDetails(approve.data);
        console.log("Tokens are not approved")
        return
    }

    const tx = await axios.get(
        `http://localhost:3001?url=https://api.1inch.dev/swap/v6.0/1/swap?fromTokenAddress=${tokenOne.address}&toTokenAddress=${tokenTwo.address}&amount=${tokenOneAmount.padEnd(tokenOne.decimals, '0')}`
    )

    let decimals = Number(`1E${tokenTwo.decimals}`)
    setTokenTwoAmount((Number(tx.data.toTokenAmount)/decimals).toFixed(2));
    setTxDetails(tx.data.tx);
}

```

Рисунок 3.10 - Функція `makeSwap`

Джерело: розроблено автором

Для створення деяких елементів була використана React бібліотека компонентів Ant Design.

```
import {Input, Popover, Radio, Modal, message} from "antd";
import {
  DownOutlined,
  ArrowDownOutlined,
  SettingOutlined,
} from "@ant-design/icons"
```

Рисунок 3.11 – Імпорт бібліотеки та компонентів

Джерело: розроблено автором

```
<Modal
  open={isOpen}
  footer={null}
  onCancel={() => setIsOpen(false)}
  title='Token select'
>
  <div className='modalContent'>
    {tokenList?.map((e,i)=>{
      return(
        <div
          className='tokenChoice'
          key={i}
          onClick={() => modifyToken(i)}
        >
          <img src={e.img} alt={e.ticker} className='tokenLogo' />
          <div className='tokenChoiceNames'>
            <div className='tokenName'>{e.name}</div>
            <div className='tokenTicker'>{e.ticker}</div>
          </div>
        </div>
      )
    })}
  </div>
</Modal>
```

Рисунок 3.12 – Приклад створення об'єкту модального вікна за допомогою Ant Design.

Джерело: розроблено автором

3.1.1 Розробка бекенду

Реалізація бекенду починається з імпорту необхідних модулів, налаштування порту та програмного забезпечення для обробки CORS та JSON. Налаштовуються заголовки для авторизації, за значенням з файлу .env. Ці процеси зображені на рисунку 3.13.

```
const express = require("express");
const Moralis = require("moralis").default;
const app = express();
const cors = require("cors");
require("dotenv").config();
const port = 3001;

app.use(cors());
app.use(express.json());
const headers = {
  "Authorization": process.env.AUTHORIZATION,
  "Content-Type": "application/json"
};
```

Рисунок 3.13 – Імпорт модулів

Джерело: розроблено автором

Обробка запиту для отримання ціни токенів реалізовується створенням маршруту /tokenPrice (рис. 3.14). Він обробляє GET запити, використовуючи API Moralis, функція отримує ціни двох токенів і розраховує їх співвідношення. Отриманий результат відправляється у відповідь.


```

app.get("/tokenPrice", async (req, res) => {

  const {query} = req;

  const resOne = await Moralis.EvmApi.token.getTokenPrice({
    address: query.addressOne
  })

  const resTwo = await Moralis.EvmApi.token.getTokenPrice({
    address: query.addressTwo
  })

  const usdPrices = {
    tokenOne: resOne.raw.usdPrice,
    tokenTwo: resTwo.raw.usdPrice,
    ratio: resOne.raw.usdPrice/resTwo.raw.usdPrice
  }

  return res.status(200).json(usdPrices);
});

```

Рисунок 3.14 – Маршрут /tokenPrice

Джерело: розроблено автором

Запуск сервера (рис 3.15) відбувається після ініціалізації Moralis через API ключ з файлу .env.

```

Moralis.start({
  apiKey: process.env.MORALIS_KEY,
}).then(() => {
  app.listen(port, () => {
    console.log(`Waiting for API Calls`);
  });
});

```

Рисунок 3.15 – Запуск сервера

Джерело: розроблено автором

3.2 Використання розробленого продукту

Після завантаження вебдодатку, користувач опиняється на головній сторінці, де він може побачити інтерактивні банери, які допомагають слідкувати за життєвим циклом проекту, останніми новинами та спеціальними пропозиціями. Внизу сторінки знаходиться таблиця з оглядом ринку, де наведені ціни на популярні криптовалюти та їх зміни за останні 24 години.

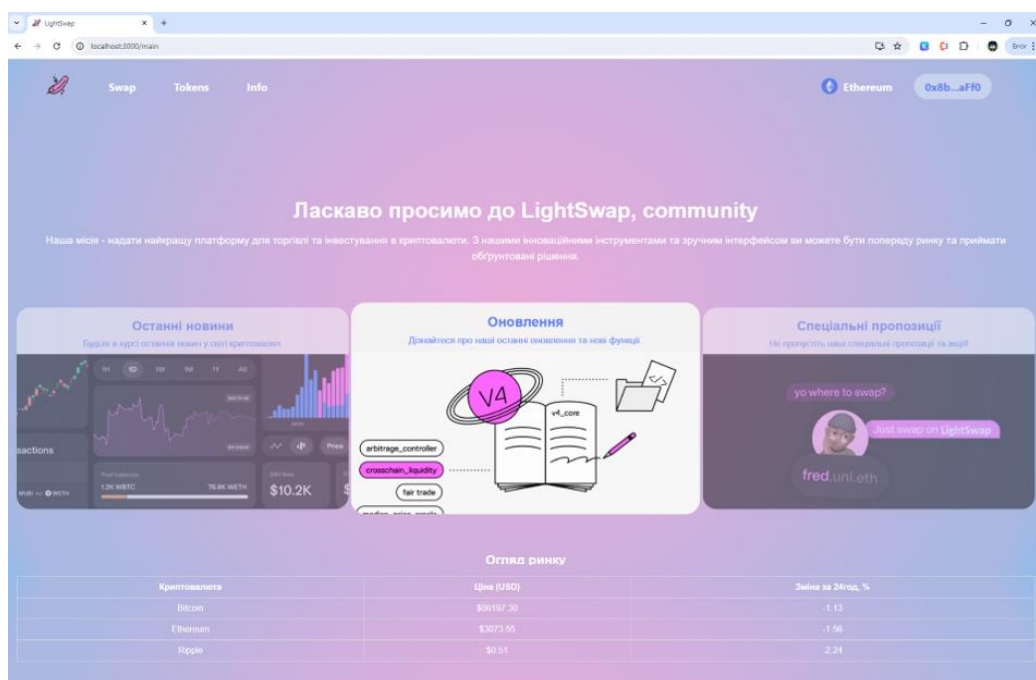


Рисунок 3.16 – Головна сторінка вебдодатку

Джерело: розроблено автором

Для здійснення обміну, потрібно перейти на сторінку Swap (рис.3.17). Авторизація користувача відбувається після натискання кнопки "Connect" у правому верхньому куті.

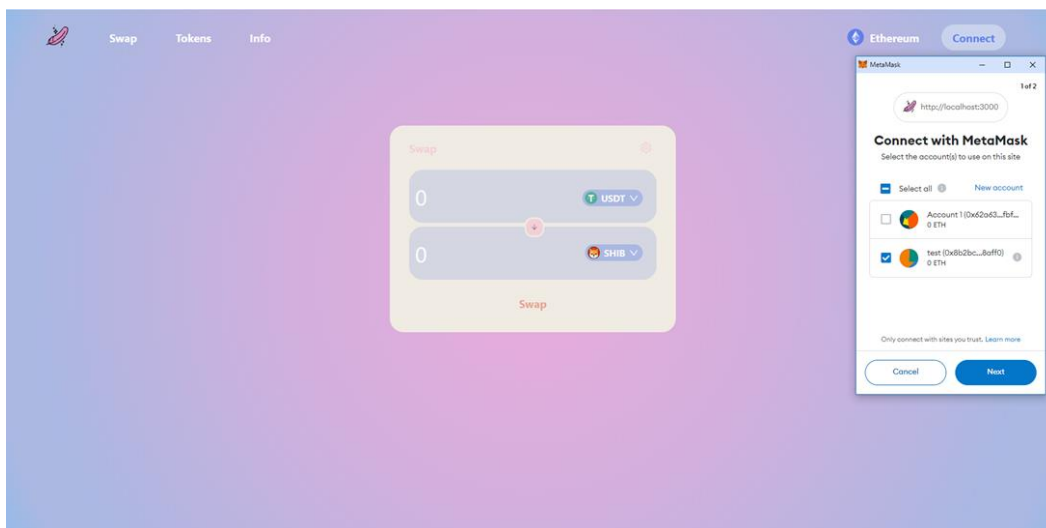


Рисунок 3.17 – Авторизація

Джерело: розроблено автором

Після цього можна обрати пару та кількість токенів зі списку (рис. 3.18), отримати актуальну ціну на них і здійснити обмін, натиснувши кнопку "Swap".

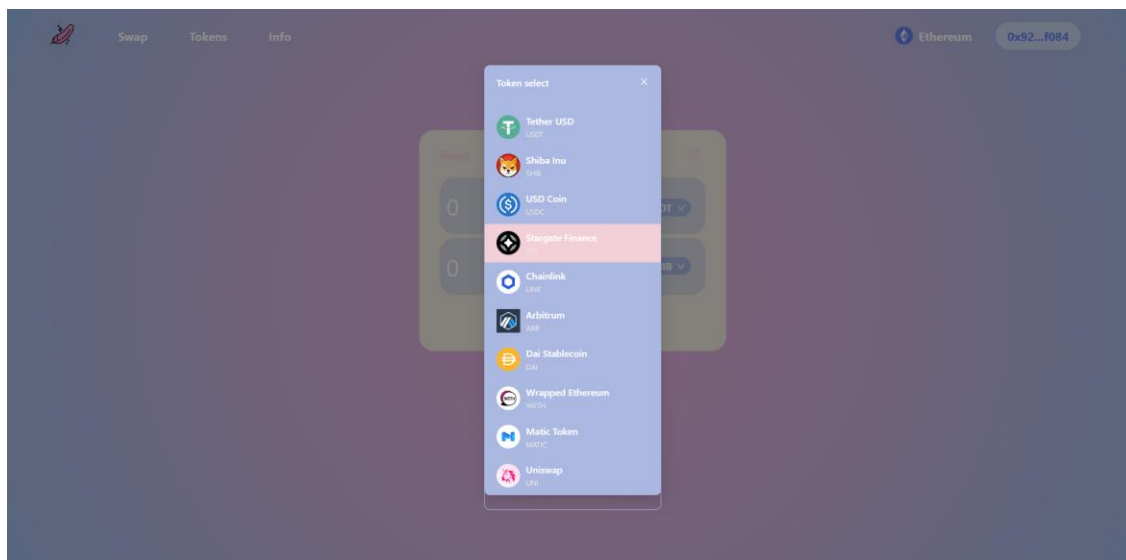


Рисунок 3.18 – Відкриття модального меню списку токенів

Джерело: розроблено автором

Прийнявши транзакцію в розширенні гаманці, ми отримаємо повідомлення (рис.3.19) про те, що транзакція виконується, а потім й про результат (рис. 3.20)

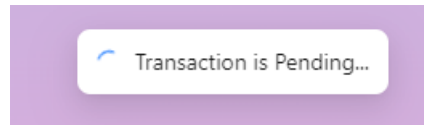


Рисунок 3.19 – Повідомлення про процес виконання транзакції

Джерело: розроблено автором

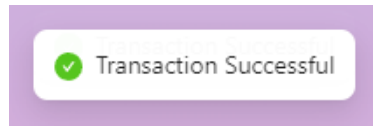


Рисунок 3.20 – Повідомлення про успішне виконання

Джерело: розроблено автором

У вебдодатку також реалізована сторінка Tokens (рис.3.20) зі списком доступних на біржі токенів, з їх описом та актуальним курсом відносно долара.

Image	Name	Ticker	Price (USD)
	Tether USD	USDT	1
	Shiba Inu	SHIB	0.00002395
	USD Coin	USDC	1
	Stargate Finance	STG	0.46
	Chainlink	LINK	16.99
	Arbitrum	ARB	0.978

Рисунок 3.21 – Сторінка Tokens

Джерело: розроблено автором

Іншою сторінкою є Info (рис. 3.21), де знаходяться відповіді на часті запитання, освітні матеріали, які допоможуть ознайомитися зі світом криптовалют, та інструкція щодо використання сайту.

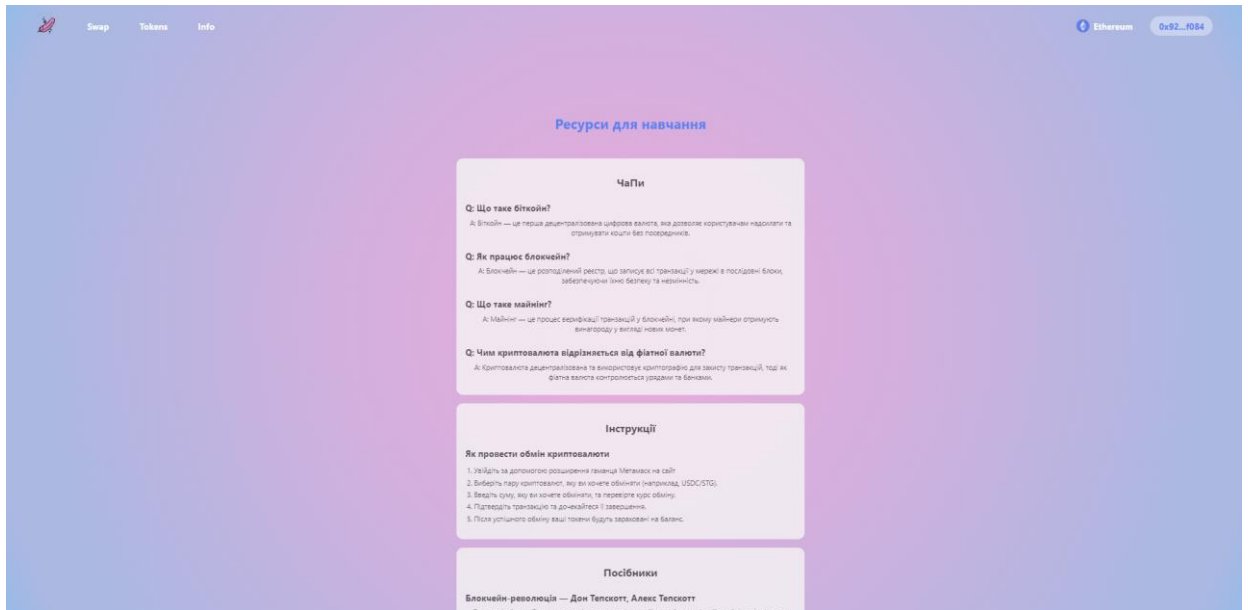


Рисунок 3.22 – Сторінка info

Джерело: розроблено автором

3.3 Тестування розробки

Для тестування бекенду можна використати перевірку роботи маршрутів за допомогою інструментів для перевірки API. Далі буде показана перевірка маршрутів за допомогою ПО Postman.

Після запуску серверу, надсилаємо GET-запит за маршрутом tokenPrice до нашого серверу та дивимося на результат, який зображено на рисунку 3.23. Для прикладу було взято два контракти токенів, ARB та USDC. Запит повертає нам правильні ціни двох токенів та їх співвідношення, а отже він працює без помилок.

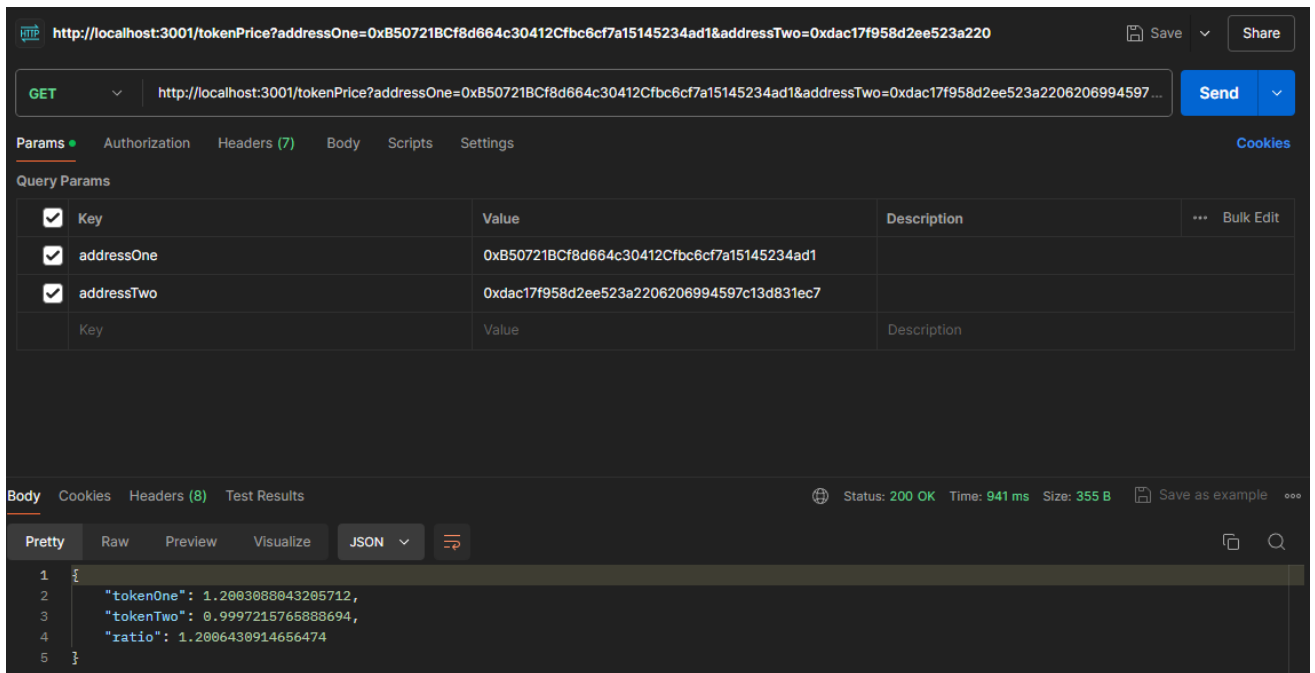


Рисунок 3.23 – Результат перевірки маршруту tokenPrice

Джерело: розроблено автором

Перевіряємо маршрут tokenPrices, який використовується для отримання цін токенів для сторінки Tokens. Результат зображено на рисунку 3.24, як можемо побачити, все знову працює без помилок і сервер надсилає нам ціну токенів відносно до usd.

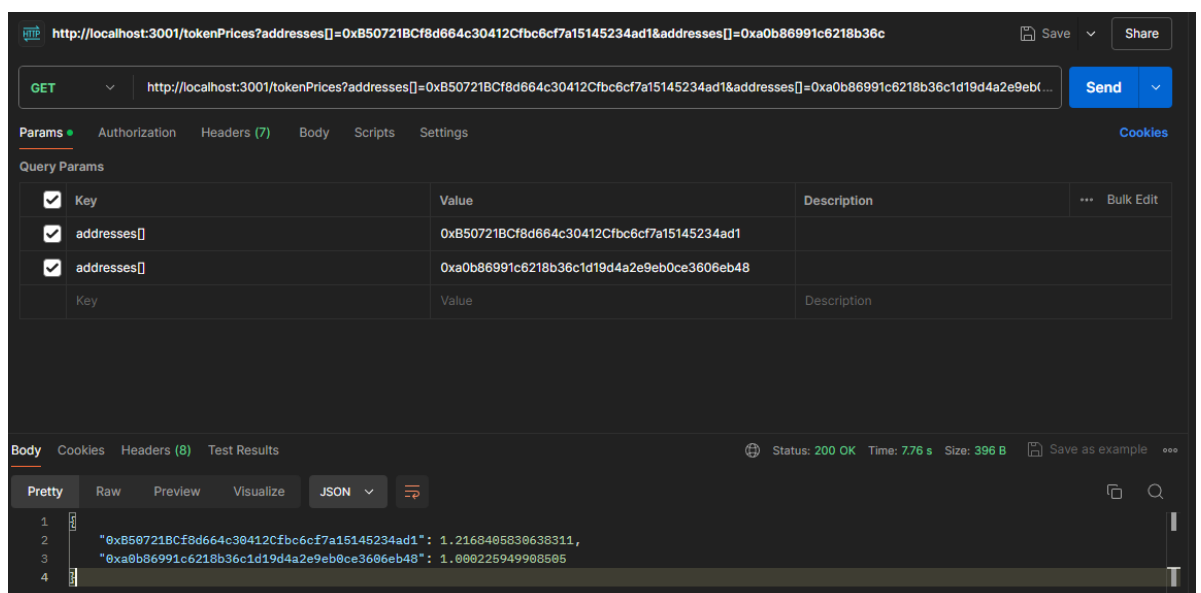


Рисунок 3.24 – Результат перевірки маршруту tokenPrices

Джерело: розроблено автором

У вебдодатку підтримки діяльності децентралізованої біржі маємо можливість протестувати вікно обміну. Завдяки використанню розширення гаманця, наприклад MetaMask, користувач не зможе відправити неправильні параметри до смарт-контракту. Навіть якщо в нього б вийшло, то транзакція не пройшла б далі на етапі перевірки у вузлах. При неправильному введенні (рис. 3.22) даних замість кількості токенів для обміну кнопка обміну є неактивною, виводиться текст про помилку.

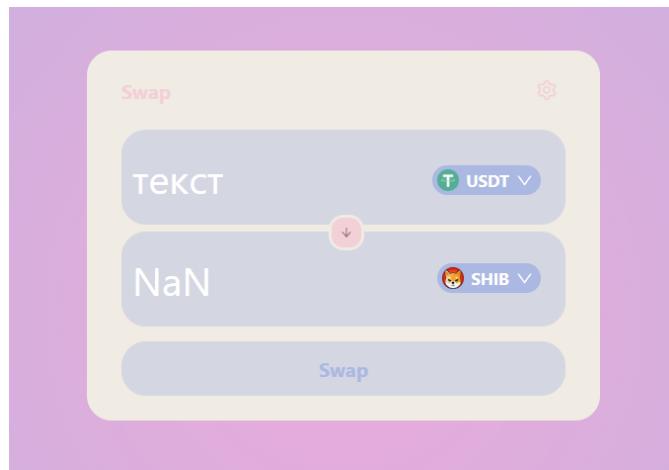


Рисунок 3.25 – Перевірка введення даних

Джерело: розроблено автором

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи бакалавра було розроблено вебдодаток підтримки функціонування децентралізованої біржі. На першому етапі було досліджено актуальність теми, визначено цільову аудиторію та проаналізовано предметну область. Аналіз існуючих аналогів децентралізованих бірж дозволив виявити їх переваги та недоліки, що стало основою для розробки власної стратегії щодо розвитку продукту.

Беручи до уваги всі потреби та вимоги щодо продукту, було обрано методи та засоби його реалізації. Інтеграція з Moralis Web3 API, 1Inch Aggregator і Wagmi забезпечить надійність і ефективність обробки транзакцій у мережах блокчейну.

В результаті проектування роботи додатку:

- Створено інтерфейс користувача за допомогою HTML, CSS, Javascript та його бібліотеки React.
- Здійснено розробку бекенду за допомогою Node.js
- Використано бібліотеку Wagmi для забезпечення безпечної та зручної авторизації користувачів
- Реалізовано отримання актуальних даних через Moralis Web3 API
- Інтегровано можливість відправки транзакцій до блокчейну Ethereum через 1inch API

Створення прототипу платформи дозволило відобразити основний функціонал біржі, а фронтенд і бекенд реалізація за допомогою React і Node.js забезпечила зручність використання і масштабованість системи.

Тестове впровадження вебдодатку дозволяє зробити висновки про адекватність розробленого додатку, коректність проведення операцій. Впровадження додатку дозволяє забезпечити користувачам високий рівень безпеки та анонімності при роботі з децентралізованою біржею.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1 Alim Al Ayub Ahmed. The Rise of DeFi: Transforming Traditional Finance with Blockchain Innovation. [Electronic resource]. URL: https://www.researchgate.net/publication/378224713_The_Rise_of_DeFi_Transforming_Traditional_Finance_with_Blockchain_Innovation (дата звернення: 18.04.2024).
- 2 Singh M., Mattackal Pauline L. Cryptoverse: Let's talk about DEX, baby. [Electronic resource]. URL: <https://www.reuters.com/markets/currencies/cryptoverse-lets-talk-about-dex-baby-2022-11-22/> (дата звернення: 18.04.2024).
- 3 JOSEPH R. DECENTRALIZED EXCHANGES: THE FUTURE OF EXCHANGES? [Electronic resource]. URL: https://www.square-management.com/wp-content/uploads/2022/01/square_white-paper-decentrated-exchange.pdf (date of access: 18.04.2024).
- 4 Allaire J. Blockchain is in from the cold – and stablecoins are set to change the financial system forever. [Electronic resource].. URL: <https://www.weforum.org/agenda/2024/01/blockchain-change-world-finance-stablecoins-internet/> (дата звернення: 18.04.2024).
- 5 Що таке Ethereum? (ETH). Посібник із використання протоколу ETH для початківців. [Electronic resource]. URL: <https://www.kraken.com/uk-ua/learn/what-is-ethereum-eth> (дата звернення: 18.04.2024).
- 6 Могайар У. Блокчейн для бізнесу. 2018. 224 с.
- 7 Uniswap Interface. [Electronic resource]. URL: <https://app.uniswap.org/swap> (дата звернення: 19.04.2024).
- 8 Що таке Ефіріум простими словами. Чим Ефір відрізняється від біткоїна. Мінфін - все про фінанси: новини, курси валют, банки. [Electronic resource]. URL:

<https://minfin.com.ua/ua/invest/articles/ethereum-prostymi-slovami/> (дата звернення: 19.04.2024).

9 SushiSwap. [Electronic resource]. URL: <https://www.sushi.com/swap> (дата звернення: 20.04.2024).

10 CryptoDiffer N. [Electronic resource]. Що таке форк? Яке значення він має для блокчейну та чому відбувається. URL: <https://tsecrypto.com/article/shho-take-fork-yake-vin-znachennya-dlya-blokchejnu-ta-chomu-vidbuvayetsya/>. (дата звернення: 20.04.2024).

11 PancakeSwap. [Electronic resource]. URL: <https://pancakeswap.finance/swap> (дата звернення: 20.04.2024).

12 Що таке фронтенд: етапи розробки та технології [Electronic resource].. URL: <https://wezom.com.ua/ua/blog/chto-takoe-front-end-razrabotka> (дата звернення: 21.04.2024).

13 Projector Creative & Tech Institute. React: Що таке React? Як почати вивчати Реакт? Основні навички. [Electronic resource]. URL: <https://cases.media/en/article/sho-take-react-js-yak-pochati-vivchati-reakt-navichki-dlya-react-developer> (дата звернення: 21.04.2024).

14 Легше, швидше, масштабніше. Чому топові компанії використовують Node.js. [Electronic resource]. URL: <https://prjctr.com/mag/whynode> (дата звернення: 22.04.2024).

15 Moralis Web3 Documentation - Step-by-Step Web3 API Tutorials [Electronic resource]. URL: <https://docs.moralis.io> (дата звернення: 24.04.2024).

16 1inch Network. [Electronic resource]. URL: <https://docs.1inch.io/docs/aggregation-protocol/introduction/> (дата звернення: 25.04.2024).

17 Why Wagmi [Electronic resource]. URL: <https://wagmi.sh/react/why> (дата звернення: 10.05.2024).

18 Що таке хеш в блокчейні [Electronic resource]. URL: <https://tradinginfo.club/що-таке-хеш-в-блокчейні/> (дата звернення: 17.05.2024).

19 The Complete Guide To Understand IDEF Diagram | EdrawMax Online. [Electronic resource]. URL: <https://www.edrawmax.com/article/the-complete-guide-to-understand-idef-diagram.html> (дата звернення: 17.05.2024).

20 IDEF0 Diagram. Vitech [Electronic resource]. URL: <https://vitechcorp.com/resources/CORE/onlinehelp/desktop/Views/IDEF0.htm> (дата звернення: 18.05.2024).

21 Use Case Diagrams | Unified Modeling Language (UML) - GeeksforGeeks. [Electronic resource]. URL: <https://www.geeksforgeeks.org/use-case-diagram/> (дата звернення: 20.05.2024).

22 Основи та принципи технології блокчейн. [Electronic resource]. URL: <https://www.bitbon.space/ua/knowledge-base/distributed-ledger-technologies-blockchain/technological-aspects-of-blockchain/foundations-and-principles-of-the-blockchain-technology> (дата звернення: 19.05.2024).

23 React Router. [Electronic resource]. URL: <http://fpm.dnu.dp.ua/2019/12/04/react-router/> (дата звернення: 21.05.2024).

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ **на розробку інформаційної системи** **«Вебдодаток підтримки функціонування** **децентралізованої біржі»**

ПОГОДЖЕНО:

Доцент кафедри інформаційних технологій

_____ Неня А.В.

Студент групи ІТ-01

_____ Прокопчук А.В.

Суми 2024

1 Призначення та мета вебдодатку підтримки функціонування децентралізованої біржі

1.1 Призначення вебдодатку підтримки функціонування децентралізованої біржі

Призначенням вебдодатку децентралізованої web3 біржі є створення інноваційної, анонімної та захищеної платформи для обміну криптовалютами без централізованого посередника.

1.2 Мета створення вебдодатку підтримки функціонування децентралізованої біржі

Метою створення децентралізованої web3 біржі є забезпечення безпечної, прозорої та ефективної платформи для торгівлі цифровими активами. Це передбачає забезпечення децентралізованої торгівлі, яка дозволить користувачам безпосередньо взаємодіяти один з одним без централізованого посередника. Вебдодаток підтримки функціонування децентралізованої біржі повинен бути розроблений опираючись на блокчейн технології. Необхідно інтегрувати смарт-контракти, для автоматизації та захисту транзакцій.

1.3 Цільова аудиторія

Цільова аудиторія децентралізованої web3 біржі включає широке коло користувачів, з унікальними потребами та інтересами. Основні групи цільової аудиторії: трейдери, інвестори, розробники децентралізованих додатків і люди, зацікавлені в децентралізації та інших фінансових інноваціях.

2 Вимоги до проекту

Вимоги до проекту в цілому

2.1.1 Вимоги до структури та функціонування

Децентралізована web3 біржа має відповідати певним структурним та функціональним вимогам. Вони включають: інтуїтивно зрозумілий інтерфейс користувача, надійні смарт-контракти для забезпечення безпеки транзакцій, а також використання технології блокчейну, для децентралізації управління та зберігання даних.

2.1.2 Вимоги до персоналу

Розробка та управління децентралізованої web3 біржі потребує команди експертів з глибокими знаннями у сферах блокчейну, веб-розробки та кібербезпеки, а також досвідчених менеджерів проектів і аналітиків. Персонал повинен володіти навичками адаптивності та командної роботи, щоб ефективно реагувати на ринкові умови та потреби користувачів, що швидко змінюються.

2.1.3 Вимоги до збереження інформації

Вимоги до збереження інформації на децентралізованій web3 біржі вимагають надійного захисту даних та їх конфіденційності. Це включає в себе використання методів шифрування (хешування), щоб гарантувати, що деталі транзакцій залишаються захищеними від несанкціонованого доступу.

2.1.4 Вимоги до розмежування доступу

Система повинна мати різні рівні доступу та контролювати дії користувача залежно від їх ролі та повноважень. Це забезпечить безпеку даних та операцій на біржі. Також важливо мати систему контролю доступу та моніторинг активності на веб-сайті для того, щоб виявити підозрілу поведінку.

2.2 Структура вебдодатку підтримки функціонування децентралізованої біржі

2.2.1 Загальна інформація про структуру веб-сайту

Стандартна структура сайту децентралізованої біржі зазвичай включає кілька ключових сторінок та елементів для забезпечення простоти використання, доступності інформації та функціональності. Ось приклад структури:

оловна Сторінка:

- Вступна інформація про проект, його місію та переваги використання.
- Інформаційні банери з актуальними новинами, оновленнями та спеціальними пропозиціями.
- Огляд ринкової інформації, включно з цінами на популярні криптовалюти та відсотком зміни ціни за останні 24 години

торінка Обміну:

- Інтерфейс для обміну валют, де користувачі можуть легко обрати токени, які вони хочуть обміняти, та вказати їх кількість.
- Інформація про поточний курс обміну та комісії за транзакцію.
- Кнопка виконання обміну ("Swap"), яка дозволяє користувачам швидко здійснити транзакцію.

торінка Токенів:

- Перелік доступних токенів для торгівлі на біржі.
- Детальна інформація про кожен токен, включаючи ціну, назву та скорочений тікер.

світній Розділ:

- Ресурси для навчання нових користувачів, включаючи посібники, ЧаПи та інструкції з використання сайту

2.2.2 Навігаційне меню

Для зручної навігації повинно бути створене меню в верхній частині кожної сторінки вебдодатку. Також на ньому треба розмістити кнопку під'єднання гаманця.

Управління контентом

Управління контентом на вебдодатку децентралізованої біржі відіграє важливу роль у забезпеченні актуальної інформації. Всі дані мають оновлюватися в режимі реального часу.

2.2.4 Дизайн веб-сайту

Дизайн повинен бути мінімалістичним та сучасним, з чіткими лініями та відсутністю зайвих елементів. Легкодоступні та зрозумілі вкладки навігації для основних функцій сайту, таких як обмін (Swap), інформація про токени (Tokens) та освітня інформація (Info).

Центральне місце на головній сторінці призначене для функції обміну з полями для вибору валют "З" і "На", введення суми та чіткою кнопкою "Обмін" (Swap).

Дизайн має бути адаптивним для різних типів пристроїв, включаючи настільні комп'ютери, ноутбуки, планшети та смартфони. Шаблон майбутнього програмного продукту зображено на рисунку А.1



Рисунок А.1 – Схема головної сторінки

Джерело: розроблено автором

2.2.5 Система навігації (карта веб-сайту)

Карта вебдодатку зображена на рисунку А.2.

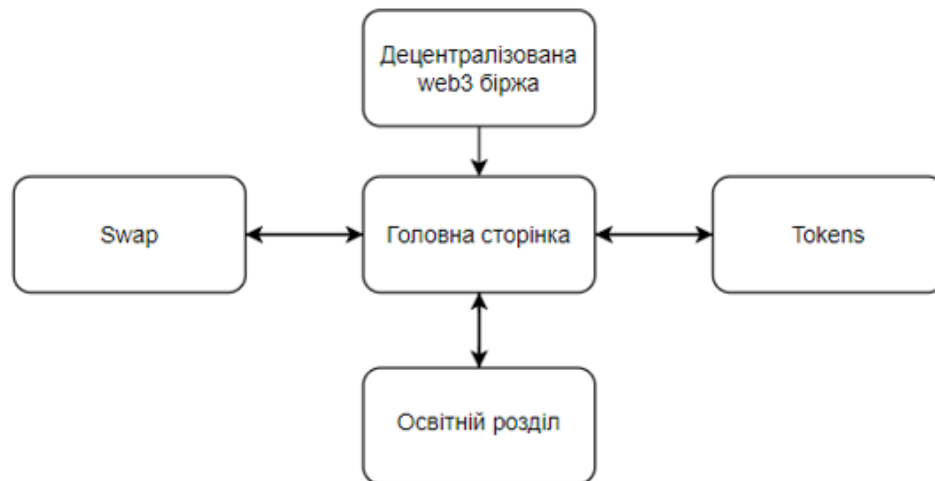


Рисунок А.2 – Схема навігації сайту

Джерело: розроблено автором

2.3 Вимоги до функціонування системи

2.3.1 Потреби користувача

Потреби користувача, визначені на основі рішення замовника, представлені у таблиці А.1.

Таблиця А.1 – Потреби користувача

	Потреби користувача	Джерело
	Ознайомлення з функціями та можливостями біржі	Клієнт, Відвідувач
	Інструкції як провести обмін токенів	Клієнт, Відвідувач
	Перегляд актуальних курсів та статистики біржі	Клієнт, Відвідувач
	Доступ до історії транзакцій та персональних портфоліо	Клієнт
	Функціонал для запиту технічної підтримки та консультацій	Клієнт
	Ресурси для освіти та самонавчання користувачів	Клієнт, Відвідувач
	Функція зворотного зв'язку для відгуків користувачів	Клієнт
	Доступ до інструментів моніторингу та аналітики для адміністраторів	Адміністратор
	Адміністративний доступ для управління контентом сайту	Адміністратор

Джерело: розроблено автором

2.3.2 Функціональні вимоги

Проаналізувавши потреби користувачів та функціональні можливості, необхідні для забезпечення ефективної роботи децентралізованої web3 біржі, було визначено наступні функціональні вимоги:

- Авторизація користувачів за допомогою розширення MetaMask
- Реалізація інтуїтивно зрозумілого інтерфейсу
- Обмін токенів з можливістю швидкого вибору
- Можливість управління контентом в режимі адміністратора
- Забезпечити доступ до детальної інформації про кожен токен та доступні торговельні пари на біржі.

2.3 Вимоги до видів забезпечення

2.3.1 Вимоги до технологій

Вебдодаток реалізований з допомогою:

2.3.2 Вимоги до лінгвістичного забезпечення

Весь текст у вебдодатку має бути виконаний українською або англійською мовами.

2.3.3 Вимоги до програмного забезпечення

Для забезпечення стабільної роботи web3 біржі вимоги до сучасних веббраузерів повинні відображати сучасні тенденції та технічні стандарти. Рекомендовані версії браузера – це останні стабільні версії, які регулярно оновлюються та підтримують необхідні веб-технології, такі як HTML5, CSS3, WebAssembly, та JavaScript ES6+. Це включає:

- Google Chrome (остання версія)
- Mozilla Firefox (остання версія)
- Apple Safari (остання версія для macOS та iOS)
- Microsoft Edge (остання версія, що базується на Chromium)
(остання версія)

Також слід забезпечити підтримку мобільних браузерів для користувачів, які використовують смартфони та планшети. Забезпечення адаптивного дизайну та функціональності на різних пристроях є ключовим фактором забезпечення доступності та зручності використання платформи.

3 Склад і зміст робіт зі створення Децентралізованої web3 біржі

Детальний опис етапів створення Децентралізованої web3 біржі наведено в таблиці А.2.

Таблиця А.2 – Етапи створення веб-сайту

№	Склад і зміст робіт	Строк розробки
	Розробка концепції та проектування архітектури	10 днів
	Верстка сторінок web-додатку	15 днів
	Реалізація смарт-контрактів для обміну	20 днів
	Інтеграція з блокчейн-мережами	15 днів
	Розробка системи аутентифікації та безпеки	12 днів

Продовження табл. А.2

	Розміщення на сервері та запуск біржі	1 день
	Перевірка працездатності	2 дні
	Фінальний реліз	1 день
	Загальна тривалість робіт	днів

Джерело: розроблено автором

ДОДАТОК Б. ПЛАНУВАННЯ РОБІТ

Децентралізовані біржі поступово стають важливим етапом у розвитку фінансових технологій. Оскільки зараз блокчейн технології стрімко розвиваються, а інтерес до криптовалют збільшується, то ці біржі стають важливими інструментами. Вебдодаток підтримки функціонування децентралізованої біржі сприятиме зростанню кількості зареєстрованих користувачів та активних учасників ринку. Це буде досягнуто завдяки конкурентоспроможним тарифам та гарантованій безпеці та повній анонімності.

Деталізація мети проєкту методом SMART. Деталізацію мети проєкту виконують SMART-методом, який дозволяє більш конкретно представити призначення розроблюваного продукту.

Для виконавця даного проєкту формат постановки SMART-мети такий: «Розробка та запуск децентралізованої web3 біржі з залученням мінімум 1000 активних користувачів до кінця 4 курсу (1 червня 2024р.)»

Результати деталізації мети даного проєкту представлено в таблиці Б.1.

Таблиця Б.1 – Деталізація мети проєкту методом SMART

Specific (конкретна)	Розробити та запустити децентралізовану web3 біржу з використанням Moralis Web3 API, 1Inch Aggregator, Wagmi, React, та NodeJS. Мета полягає у створенні безпечного, ефективного, та інтуїтивно зрозумілого інтерфейсу для спрощення торгівлі криптовалютами.
Measurable (вимірювана)	Критеріями успіху є залучення мінімум 1000 активних користувачів за перші 6 місяців, обробка щонайменше 500 транзакцій на день, та досягнення позитивних відгуків від користувачів (більше 80% задоволених).

Продовження табл. Б.1.

Achievable (досяжна узгоджена)	Проект має достатній фінансовий бюджет та технічні ресурси для початку розробки, включаючи кваліфіковану команду розробників та доступ до необхідних технологій.
Relevant (реалістична)	Розробка DEX відповідає потребам сучасного ринку криптовалют та сприяє стратегічним цілям компанії у сфері блокчейн-технологій, забезпечуючи більш широкий доступ до децентралізованої торгівлі.
Time-framed (обмежена в часі)	Конкретний термін виконання – до кінця 4 курсу (1 червня 2024р.)

Джерело: розроблено автором

Планування змісту робіт. Структура розподілу робіт, або WBS, є ключовим інструментом проєктного менеджменту. Вона використовується для поділу складного проєкту на менші керовані компоненти або робочі пакети. Діаграма дає змогу зрозуміло визначити обсяг роботи проєкту та сприяє ефективному плануванню, виконанню, контролю та оцінці його прогресу. WBS для нашого проєкту розробки вебдодатку підтримки функціонування децентралізованої біржі деталізує кожен етап та підетап роботи. Організовує їх у логічну ієрархію від загального до конкретного. На рисунку Б.1 представлено WBS з розробки децентралізованої web3 біржі «LightSwap».

Планування структури виконавців. Наступним етапом після декомпозиції процесів є розробка організаційної структури виконавців або OBS. Вона відображає розподіл організаційних відповідальностей у проєкті. Ця структура є необхідною для визначення ролей, відповідальностей та відносин між учасниками проєкту. OBS нашого проєкту визначає, як учасники розробки взаємодіють для досягнення проєктних цілей. На рисунку Б.2 представлено організаційну структуру

планування проєкту.

Список виконавців, що функціонують в проєкті описано в таблиці Б.2.

Таблиця Б.2 – Виконавці проєкту

Роль	ПІБ	Проєктна роль
Розробник	Прокопчук А.В.	Виконує front-end, back-end розробку та інтеграцію з блокчейном
Проєктувальник	Прокопчук А.В.	Виконує проєктування бази даних та розробляє структуру web-додатку.
Тестувальник	Прокопчук А.В.	Відповідає за тестування функціоналу та дизайну web-додатку.
Керівник проєкту	Неня А.В.	Формує завдання на розробку проєкту.
Менеджер проєкту	Прокопчук А.В.	Відповідає за виконання термінів, розподіл ресурсів та завдань між учасниками. Виконує збір та аналіз даних.

Джерело: розроблено автором

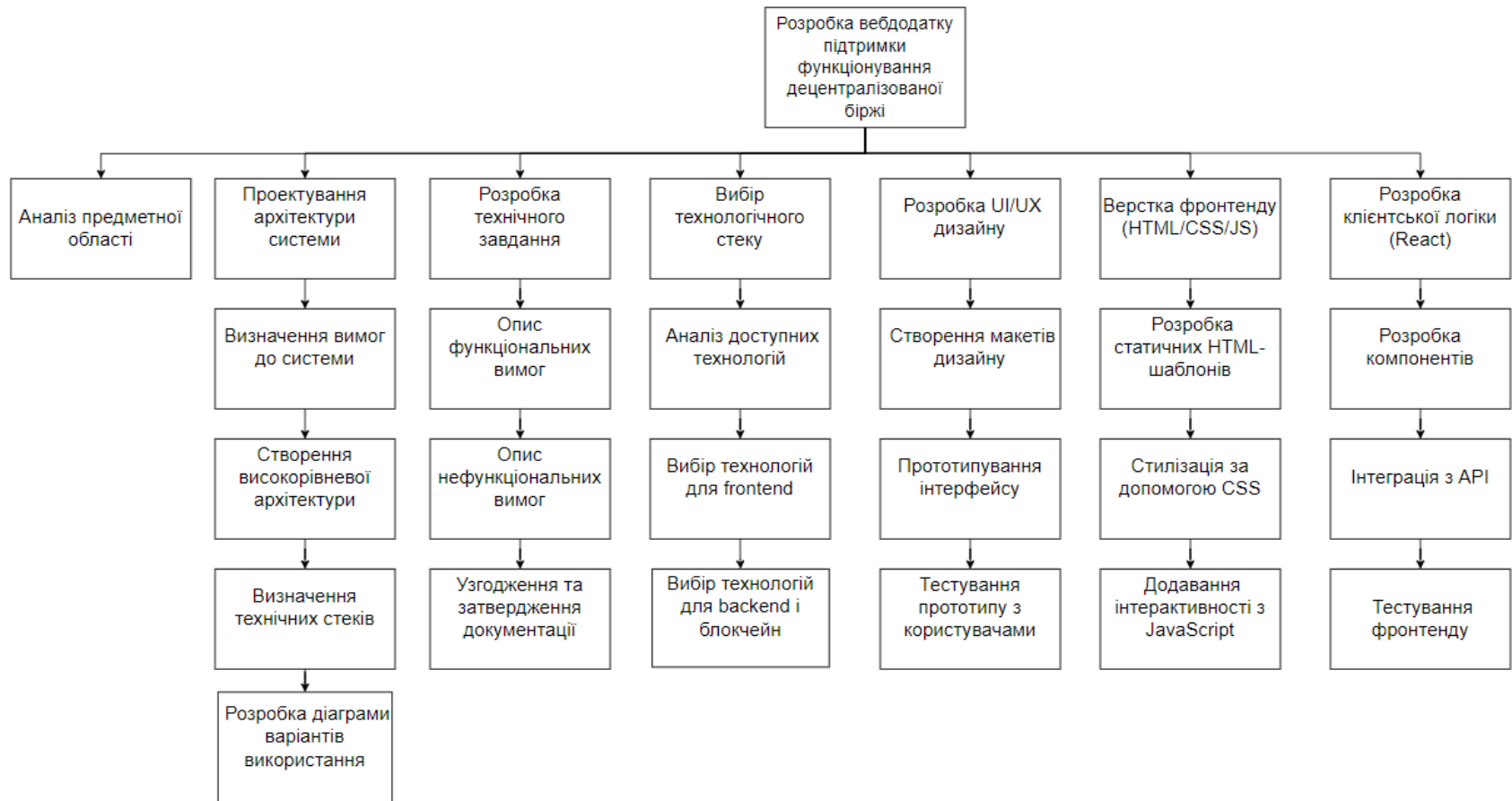


Рисунок Б.1 – WBS-структура робіт проекту

Джерело: розроблено автором

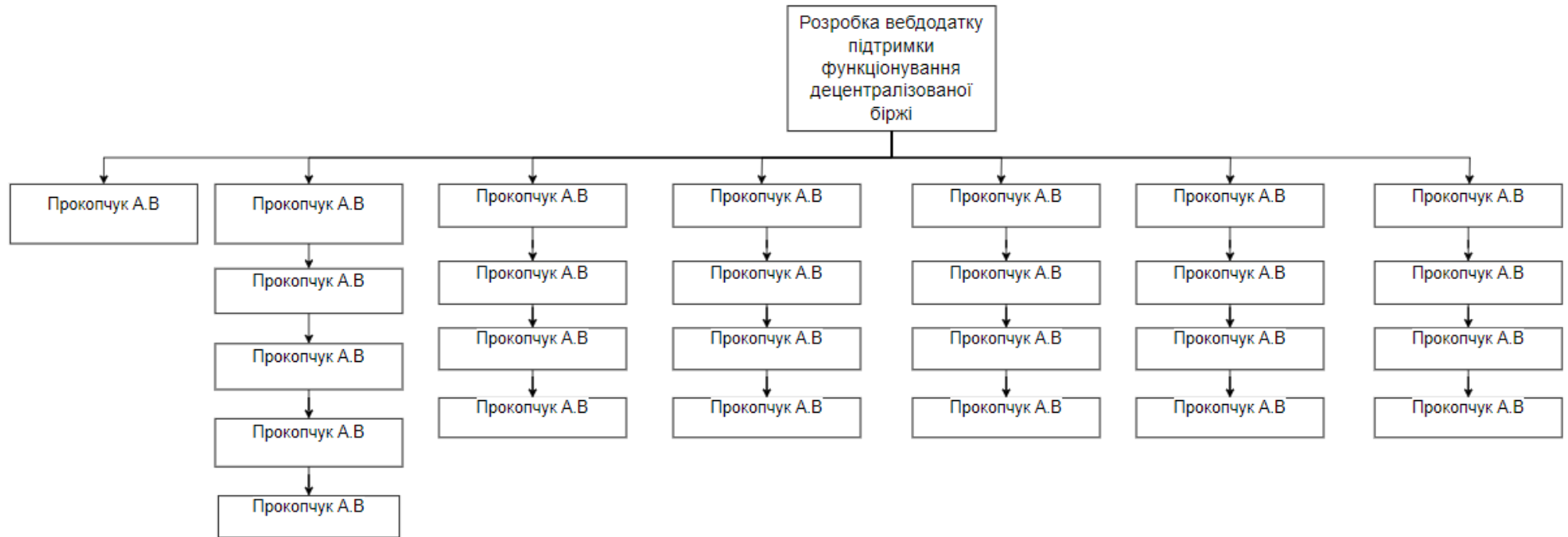


Рисунок Б.1 – OBS-структура робіт проекту

Джерело: розроблено автором

Діаграма Ганта. Створення діаграми Ганта є ключовим кроком у процесі планування проєкту. Ця діаграма представляє собою детальний розклад виконання робіт з точним визначенням дат. Вона дозволяє чітко уявити тривалість різних процесів проєкту.

Календарний графік проєкту представлено на рисунках Б.3-Б.5.

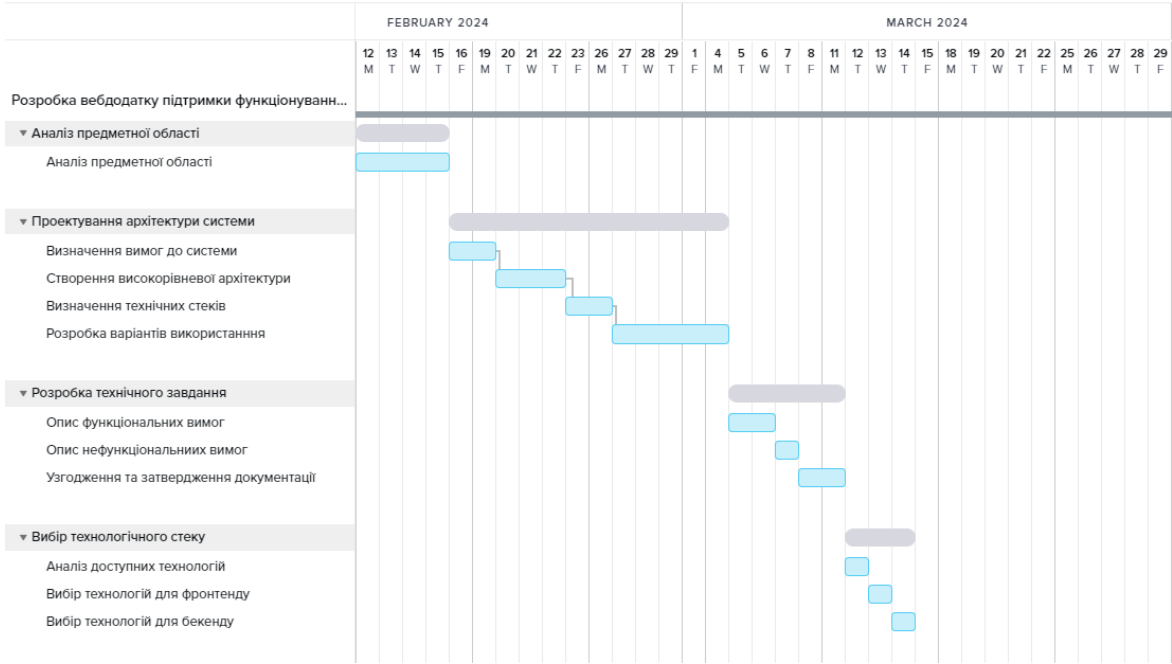


Рисунок Б.3 – Календарний графік проєкту
Джерело: розроблено автором

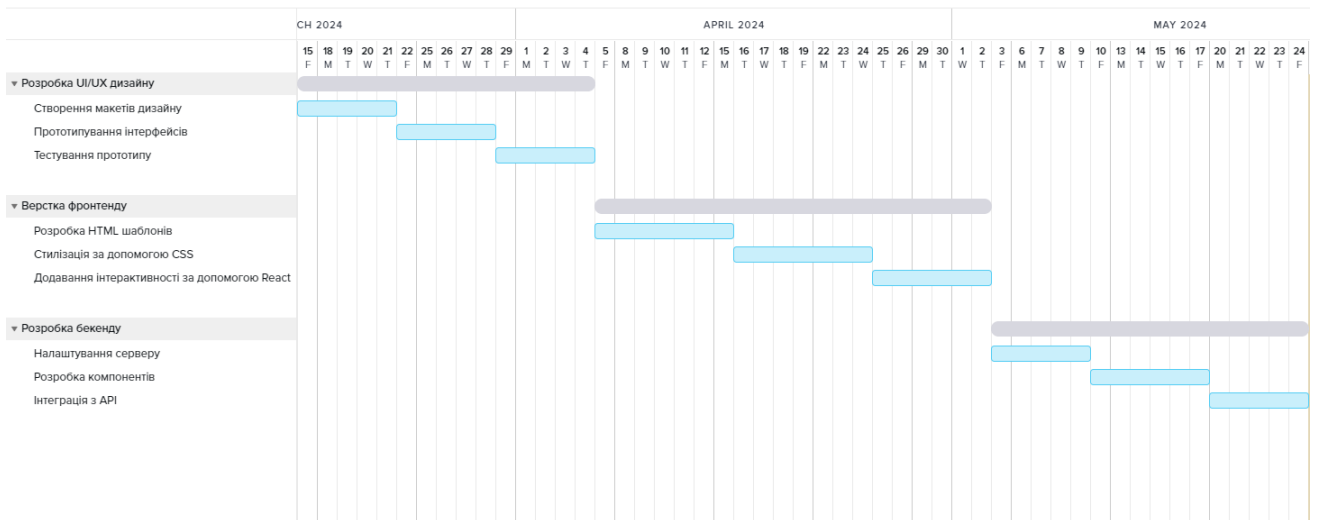


Рисунок Б.4 – Продовження календарного графіку проєкту
Джерело: розроблено автором

Управління ризиками проєкту. При проведенні якісної оцінки ризиків основним завданням є ідентифікація ризиків, які вимагають негайного вирішення. Реакція на ризики буде залежати від їх значущості. У таблиці Б.3 надано перелік ризиків даного проєкту.

Таблиця Б.3 – Ризики проєкту

№ ризику	Назва (опис) ризику
1	Технічні недоліки у розробці: Непередбачені проблеми в технічному виконанні проєкту.
2	Зміни у блокчейн-регулюванні: Ризик змін у законодавстві, які можуть вплинути на роботу блокчейну.
3	Проблеми з безпекою даних: Потенційні хакерські атаки або витіки даних, що можуть завдати шкоди проєкту.
4	Зміни в технологіях розробки: Зміни які можуть змінити інструменти та підходи до розробки
5	Затримки у розробці: Потенційні затримки в розробці через різні чинники, такі як проблеми в команді або технічні труднощі.
6	Проблеми з масштабуванням системи: Труднощі або обмеження, пов'язані з масштабуванням біржі.
7	Відсутність досвіду: відсутність достатнього досвіду розробки децентралізованих додатків
8	Проблеми зі зберіганням даних: Технічні виклики або обмеження, що стосуються зберігання великих обсягів даних.
9	Зміни в технологічних стандартах: Ризик швидких змін у технологічних стандартах, які можуть вплинути на проєкт.
10	Проблеми із забезпеченням ліквідності: Ризик недостатньої ліквідності на біржі, що може вплинути на її функціонування та привабливість для користувачів.

Джерело: розроблено автором

Результати оцінки ризиків надано у таблиці Б.4.

Таблиця Б.4 – Результати визначення ймовірності, впливу та рангу ризиків проєкту

№ ризику	Назва (опис) ризика	Ймовірність (0,1-0,9)	Вплив (0,05-0,8)	Р а н г
1	Технічні недоліки у розробці	0.2	0.4	0.08
2	Зміни у блокчейн-регулюванні	0.2	0.7	0.14
3	Проблеми з безпекою даних	0.1	0.4	0.04
4	Зміни в технологіях розробки	0.2	0.3	0.06
5	Затримки у розробці	0.3	0.4	0.12
6	Проблеми з масштабування системи	0.1	0.5	0.05
7	Відсутність досвіду	0.2	0.3	0.06
8	Проблеми зі зберіганням даних	0.1	0.4	0.04
9	Зміни в технологічних стандартах	0.3	0.4	0.12
10	Проблеми із забезпеченням ліквідності	0.2	0.6	0.12

Джерело: розроблено автором

Таблиця Б.5 представляє шкалу для класифікації ризиків за величиною впливу на проєкт та ймовірністю їх виникнення.

Таблиця Б.5 – Шкала оцінювання ризиків за ймовірністю виникнення та величиною впливу

Оцінка	Ймовірність виникнення	Вплив ризику	Тип ризику
1	Низька	Низький	Прийнятні
2	Середня	Середній	Виправдані
3	Висока	Високий	Недопустимі

Джерело: розроблено автором

Для мінімізації впливу ризиків на проєкт необхідно розробити план реагування на них. Цей процес включає оцінку потенційних наслідків для проєкту та розробку заходів реагування. Аналіз базується на критеріях, вказаних у таблиці Б.4. В результаті планування реагування на ризики була створена матриця, що відображає ймовірність та вплив цих ризиків (див. рис. Б.4), де зелений колір позначає прийнятні ризики, жовтий – виправдані, а червоний – недопустимі.

Ймовірність виникнення ризику	Вплив ризику				
	0,05	0,1	0,2	0,4	0,8
0,9	0,045	0,09	0,18	0,36	0,72
0,7	0,035	0,07	0,14	0,28	0,56
0,5	0,025	0,05	0,10 R3, R4	0,20	0,40
0,3	0,015	0,03 R1	0,06	0,12	0,24
0,1	0,005	0,01	0,02	0,04	0,08 R2

Рисунок Б.4. – Матриця ймовірності та впливу

Джерело: розроблено автором

Класифікація ризиків проєкту за рівнем, відповідно до отриманого значення

індексу, представлена у таблиці Б.5.

Таблиця Б.5 – Шкала оцінювання ризику за рівнем

№	Назва	Межі	Ризики, які входять (номера)
1	Прийнятні	$0,005 \leq R \leq 0,05$	6, 8, 3
2	Виправдані	$0,05 < R \leq 0,14$	2, 9, 10, 1, 4, 5, 7
3	Недопустимі	$0,14 < R \leq 0,72$	-

Джерело: розроблено автором

Таблиця Б.6 – Ризики та стратегії реагування на них

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	Тип стратегії реагування	План А	План Б
RS_1	Новий	Технічні недоліки у розробці	0.2	0.4	0.08	Зменшення	Передбачити буферний час для виправлення помилок	Розробка альтернативних рішень, задіяння резервної команди
RS_2	Новий	Зміни у блокчейн-регулюванні	0.2	0.7	0.14	Зменшення	Моніторинг законодавчих змін, лобіювання інтересів	Адаптація проекту до нових законодавчих вимог
RS_3	Новий	Проблеми з безпекою даних	0.1	0.4	0.04	Зменшення	Застосування сучасних технологій шифрування	Розробка плану відновлення даних
RS_4	Новий	Зміни в технологіях розробки	0.2	0.3	0.06	Передача	Стеження за новими технологіями та швидка адаптація до змін	Перегляд і впровадження нових технологій
RS_5	Новий	Затримки у розробці	0.3	0.4	0.12	Зменшення	Постійний контроль за графіком проекту	Реалюкування ресурсів та перегляд графіка
RS_6	Новий	Проблеми з масштабуванням системи	0.1	0.5	0.05	Зменшення	Планування модульної архітектури	Пошук альтернативних технічних рішень

Продовження таблиці Б.6.

ID	Статус ризику	Опис ризику	Ймовірність виникнення	Вплив ризику	Ранг ризику	Тип стратегії реагування	План А	План Б
RS_7	Новий	Відсутність досвіду	0.2	0.3	0.06	Зменшення	Регулярне самонавчання та консультації з фахівцями	Залучення зовнішніх консультантів або менторів
RS_8	Новий	Проблеми зі зберіганням даних	0.1	0.4	0.04	Зменшення	Використання надійних технологій зберігання даних	Реалізація плану надзвичайного відновлення даних
RS_9	Новий	Зміни в технологічних стандартах	0.3	0.4	0.12	Зменшення	Постійне оновлення знань команди про нові технології	Адаптація проєкту під нові технічні стандарти
RS_10	Новий	Проблеми із забезпеченням ліквідності	0.2	0.6	0.12	Зменшення	Розробка стратегії приваблення користувачів і збільшення торговельного обороту	Пошук альтернативних джерел ліквідності

Джерело: розроблено автором

