Специфікація програмного продукту

1 ВСТУП

1.1 Огляд проекту

Сукупним результатом багатовікової еволюції ринку є народження централізованої інфраструктури, яка забезпечує стандартизацію контрактів, ліквідність, регуляторний контроль та зручність обміну. Саме на цьому підґрунті в кінці 2000-х з’явилася технологія блокчейн і криптовалюти, які кинули виклик традиційній монополії бірж та банків.

1.2 Мета

Метою роботи є розробка децентралізованої системи, що дозволяє користувачам безпечно і швидко здійснювати обмін токенів, створювати пули ліквідності, брати участь у запуску нових проектів, а також взаємодіяти з цифровими активами без участі посередників.

1.3 Область застосування

Серверну частині можна запустити на операційний системі Linux, де встановлено мову програмування Rust 1.84 або вище та Docker. Смартконтракти розгортаються в локальній мережі Solana Testnet, яку також потрібно розгортати на сервері з Linux та встановленим інструментарієм solana-cli. Проте смартконтракти можуть бути розгорнуті в публічних мережах Solana Mainnent та Solana Devnet.

1.4 Короткий зміст

Цільова аудиторія буде звертати увагу на: зручність використання механізмів обміну, справедливе ціноутворення та швидкість роботи додатку.

1.5 Означення та абревіатури

DeFi – Decentralized Finance (децентралізовані фінанси)

AMM – Automated Market Maker (алгоритм автоматичного маркет-мейкінгу)

DEX – Decentralized Exchange (Децентралізована біржа)

2 ЗАГАЛЬНИЙ ОПИС

2.1 Перспективи продукту

Розроблювана система надасть користувачам можливість здійснювати обмін токенами в мережі Solana та брати участь в запуску нових проектів.

2.2 Функції продукту

Система матиме наступний функціонал для користувачів: створення пулів ліквідності, додавання ліквідності, вивід ліквідності, обмін токенів в пулі, збір нагород для платформи.

Система матиме наступний функціонал для адміністраторів: створення конфігурацій пулів ліквідності, зміна конфігурацій пулів, створення конфігурацій лаунчпулів, зміна конфігурацій пулів, передача прав, створення та ініціалізація лаунчпулів.

2.3 Характеристики користувачів

Цільовим користувачем виступає користувач знайомий з блокчейном Solana та володіючий активами в цій мережі.

2.4 Загальні обмеження

2.4.1 Операційне середовище

Середа розробки серверної частини: JetBrains Rustrover 2024.3.7.

В якості СКБД було використано ScyllaDB.

2.4.2 Технологія розробки

Rust, Anchor, Solana, TypeScript.

2.5 Припущення й залежності

Припущення:

1. Система мати попит серед користувачів, які потребують децентралізованого середовища для обміну токенів.

Залежності:

1. Початкова версія матиме підтримку лише CPMM пулів ліквідності.

2. Блокчейн частина додатку буде розміщена в Solana Testnet з подальшою можливістю розгортання в Mainnet.

3. Всі критично важливі операції, такі як відправка транзакцій в блокчейн відбуваються ізольовану на клієнтській частині задля безпеки приватного ключа користувач.

3 КОНКРЕТНІ ВИМОГИ

3.1 Вимоги до інтерфейсів

3.1.1 Апаратний інтерфейс

Апаратний інтерфейс платформи реалізується через веб-додаток, який взаємодіє з сервером та вузлом Solana.

3.1.2 Комунікаційний протокол

HTTP/HTTPS, Solana JSON RPC, Websocket.

3.1.3 Вимоги до пам’яті

Для GNU/Linux (64-bit):

- 4 GB RAM;

- 4 GB дискового простору для роботи Scylla DB.

Розподіл пам’яті по компонентах GNU/Linux може відрізнятись залежно від дистрибутиву.

3.2 Атрибути програмного продукту

3.2.1 Безпека

Усі критичні дані користувача зберігаються на блокчейні, а приватні ключі захищені браузерним розширенням гаманця.

3.2.2 Супроводжуваність

Після випуску програмного продукту планується додавання нового функціоналу та покращення досвіду користування для користувача.

3.2.3 Переносимість

Серверний додаток повинен працювати на усіх серверах які відповідають технічним вимогам та вимогам до пам’яті.

3.2.4 Продуктивність

Серверний додаток повинен обробляти події на блокчейні не більше ніж за 400 мілісекунд.

3.2.5 Вимоги до бази даних

Вимоги до бази даних не висуваються.