

АРХІТЕКТУРА ОПЛАТИ ЗА ПРЕМІУМ-ПІДПИСКУ НА СЕРВІС З ВИКОРИСТАННЯМ КРИПТОГАМАНЦЯ ЧЕРЕЗ SOLIDITY І METAMASK

Омельченко В.В., Романовський І.М., Лендел Я.Р., Саманцов О.О.
e-mail: vladyslav.omelchenko3@nure.ua, illia.romanovskiyi@nure.ua,
yaroslav.lendel@nure.ua, oleksandr.samantsov@nure.ua

Харківський національний університет радіоелектроніки, каф. ПІ
м. Харків, Україна

This study explores the implementation of a blockchain-based payment system for premium subscriptions using Solidity smart contracts and Metamask integration. The proposed solution ensures security, transparency, and automation of transactions by eliminating intermediaries. The system consists of a smart contract, a Metamask-integrated web interface, and a backend for data synchronization. Testing in the Ethereum test network confirmed its reliability and efficiency. While challenges like user accessibility and crypto price volatility exist, this approach enhances trust and optimizes digital payment processes.

У сучасному цифровому світі, де технології блокчейн та криптовалюти стрімко розвиваються, інтеграція децентралізованих фінансових інструментів у вебсервіси стає все більш актуальною. Це відкриває нові можливості для бізнесів та користувачів, забезпечуючи прозорість, безпеку та автономність фінансових операцій [1]. Однією з перспективних областей застосування є впровадження оплати преміум-підписок за допомогою смарт-контрактів та криптогаманців.

Метою даної роботи є розробка інноваційного механізму оплати преміум-підписки для вебсервісу, який базується на використанні смарт-контрактів, написаних на мові програмування Solidity, та інтеграції з популярним криптогаманцем Metamask. Такий підхід дозволяє автоматизувати процеси оплати, забезпечити високий рівень безпеки транзакцій та надати користувачам зручний інтерфейс для взаємодії з блокчейн-технологіями [2].

Основні завдання, які ставляться в рамках цієї роботи, включають:

1. Розробка смарт-контракту для обробки платежів: створення безпечного та ефективного смарт-контракту, який буде відповідати за прийом платежів від користувачів, збереження інформації про статус їх підписок та управління доступом до преміум-функцій сервісу.
2. Реалізація механізму підтвердження підписки: забезпечення автоматичного оновлення статусу підписки користувача після здійснення оплати, а також розробка системи сповіщень про успішність транзакцій та терміни дії підписки.

3. Розробка клієнтського інтерфейсу для взаємодії з блокчейном: інтеграція Metamask у вебінтерфейс сервісу, що дозволить користувачам легко та безпечно здійснювати платежі, переглядати статус своїх підписок та керувати ними безпосередньо через браузер.

Застосування мови програмування Solidity для написання смарт-контрактів на платформі Ethereum надає можливість створювати децентралізовані додатки (dApps) з високим рівнем безпеки та прозорості [3]. Смарт-контракти дозволяють автоматизувати виконання умов угод, виключаючи потребу в посередниках та мінімізуючи ризики шахрайства. Інтеграція ж з Metamask забезпечує користувачам зручний доступ до їхніх криптовалютних активів та можливість взаємодії з блокчейном без необхідності встановлення додаткового програмного забезпечення.

Архітектура запропонованої системи складається з трьох основних компонентів:

1. Смарт-контракт, написаний на Solidity, який відповідає за обробку транзакцій, збереження даних про підписки та управління доступом до преміум-функцій. Приклад смарт-контракту наведено на рисунку 1.

```
1 // SPDX-License-Identifier: GPL-3.0
2
3 pragma solidity >=0.8.2 <0.9.0;
4
5 contract WalletContract {
6     address payable public owner;
7
8     constructor(){ 240120 gas 215600 gas
9         owner = payable(msg.sender);
10    }
11
12    function getBalance() public view returns(uint256){ 2573 gas
13        require(owner == msg.sender, "Not allowed");
14        return address(this).balance;
15    }
16
17    function withdraw() public { infinite gas
18        require(owner == msg.sender, "Not allowed");
19        require(address(this).balance > 0, "No funds available");
20
21        owner.transfer(address(this).balance);
22    }
23
24    receive() external payable {} undefined gas
25 }
26
```

Рисунок 1 – Приклад Смарт-контракту

2. Веб-інтерфейс з інтеграцією Metamask, що надає користувачам можливість авторизації, здійснення платежів та перегляду статусу своїх підписок у реальному часі.

3. Бекенд-сервер, який забезпечує синхронізацію даних між блокчейном та вебзастосунком, обробляє запити користувачів та надає необхідну інформацію для коректної роботи сервісу.

Для оцінки ефективності запропонованого рішення було проведено тестування в тестовій мережі Ethereum (Rinkeby). Результати показали, що система здатна обробляти транзакції швидко та безпечно, забезпечуючи користувачам надійний доступ до преміум-функцій сервісу. Порівняно з традиційними методами оплати, використання блокчейн-технологій дозволяє знизити витрати на транзакції та підвищити довіру користувачів до сервісу.

Однак, впровадження таких технологій не позбавлене викликів. Однією з основних проблем є забезпечення зручності використання для користувачів, які не знайомі з криптовалютами та блокчейном. Для цього необхідно розробити інтуїтивно зрозумілий інтерфейс та надати детальні інструкції щодо використання Metamask та здійснення платежів. Крім того, варто враховувати можливі коливання вартості криптовалют, що можуть впливати на вартість підписок, та забезпечити механізми для мінімізації таких ризиків.

Запропонований підхід до оплати преміум-підписок за допомогою смарт-контрактів та інтеграції з Metamask відкриває нові горизонти для розвитку вебсервісів, надаючи користувачам безпечні, прозорі та автоматизовані способи здійснення платежів. Це сприяє підвищенню довіри до сервісу, розширенню його функціональних можливостей та залученню нових користувачів, які цінують сучасні технологічні рішення.

Список використаних джерел:

1. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. <https://bitcoin.org/bitcoin.pdf>
2. Wood, G. (2014). Ethereum: A Secure Decentralised Generalised Transaction Ledger. <https://ethereum.github.io/yellowpaper/paper.pdf>
3. Dannen, C. (2017). Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners. Apress. <https://link.springer.com/book/10.1007/978-1-4842-3086-4>