UDC 004.424

SMART CONTRACTS FOR TASK MANAGEMENT IN CROWDSOURCING

Nurbay Kairbek Talgatuly

Master student, Astana IT University, Astana, Kazakhstan

Scientific Supervisor: Aliya Nugumanova

Traditional crowdsourcing platforms rely on centralized intermediaries to manage tasks, which introduces additional costs, delays, and security risks. Smart contracts offer a decentralized alternative by automating task creation, execution, and verification, ensuring transparency and reliability without the need for third-party intervention. This paper examines the application of smart contracts in crowdsourcing task management, discussing their architecture, key components, existing solutions, economic model, and potential improvements.

Keywords: Blockchain, Smart Contracts, Crowdsourcing, Decentralized Task Management.

INTRODUCTION

Crowdsourcing allows individuals to complete tasks posted by requesters in exchange for payment. While this model provides flexibility and access to a global workforce, traditional platforms like Upwork and Amazon Mechanical Turk operate under a centralized structure. This means that a single entity controls how tasks are assigned, how payments are processed, and how disputes are resolved.

This centralized control creates several challenges. First, platforms charge high service fees, often taking a significant percentage of workers' earnings. Second, payment processing can be slow, with workers waiting days or even weeks to receive their money due to reliance on banking systems. Additionally, platform rules are not always transparent, and users may have their accounts restricted or suspended without clear justification. Lastly, the task allocation process is not always fair, as it is often based on opaque ranking algorithms that may prioritize certain workers over others.

Blockchain technology addresses these challenges by introducing decentralized task management through smart contracts. Instead of relying on a central authority, transactions and agreements are executed automatically based on predefined conditions. This eliminates unnecessary intermediaries, ensuring transparent task distribution, instant payments, and greater security for both workers and requesters.

By leveraging blockchain, crowdsourcing platforms can become more fair, efficient, and resistant to manipulation.

LITERATURE REVIEW

Several studies have explored the integration of blockchain and smart contracts into the crowdsourcing domain to address challenges such as trust, transparency, and centralized control. Traditional platforms like Amazon Mechanical Turk and Upwork have been widely criticized for their opaque operations, high service fees, and centralized governance (Zhang, Xue, & Liu, 2019). These limitations have spurred the development of decentralized alternatives.

Li et al. (2019) introduced CrowdBC, a blockchain-based decentralized framework for crowdsourcing. Their system enables trustless collaboration between requesters and workers using smart contracts, focusing on identity management, reward enforcement, and task verification. Similarly, Ma et al. (2019) conducted a survey of blockchain applications in crowdsourcing, highlighting the potential of smart contracts to automate task handling, reputation tracking, and payment mechanisms. Their research confirms that blockchain systems can enhance fairness and security but notes scalability and verification challenges.

Khan et al. (2021) reviewed the broader applications of smart contracts and identified key challenges such as high gas fees, vulnerability to attacks (e.g., reentrancy), and the complexity of integrating off-chain data. These insights are crucial for implementing smart contracts in crowdsourcing, where real-world task validation often involves subjective judgment.

Singh et al. (2019) addressed the formal verification of smart contracts and proposed approaches to reduce the risk of vulnerabilities in contract logic. Given that trust in automation is critical for decentralized systems, such verification methods can significantly enhance the reliability of crowdsourcing platforms.

While the aforementioned systems offer foundational models, most lack adaptable tokenomics and flexible pricing mechanisms. Our work builds upon these frameworks by incorporating dynamic task pricing, reputation-based access, and staking-based accountability, expanding the economic robustness of decentralized crowdsourcing.

SMART CONTRACTS AND THEIR APPLICATION IN CROWDSOURCING

Smart contracts are self-executing agreements stored on the blockchain, designed to function without intermediaries. Unlike traditional contracts, which rely on centralized authorities for enforcement, smart contracts automatically execute transactions when specific conditions are met. This makes them a powerful tool for ensuring trust and transparency in digital interactions.

In crowdsourcing, where multiple participants interact remotely, ensuring fairness, efficiency, and security is crucial. Smart contracts enable automated task management, ensuring that workers receive payments only when they successfully

complete tasks according to pre-agreed rules. Since these agreements are recorded on the blockchain, they are tamper-proof and publicly verifiable, eliminating disputes over payments or task completion.

By removing reliance on third-party platforms, smart contracts streamline the crowdsourcing process, making it more inclusive, cost-effective, and resistant to fraud. This shift toward decentralized task management paves the way for a new generation of crowdsourcing platforms that empower both workers and requesters with greater control and transparency.

Smart contracts transform crowdsourcing by automating task management, ensuring fairness, and removing the need for intermediaries. They introduce several essential functions that streamline the process for both requesters and workers:

- Task Creation and Posting. Requesters can create and publish tasks directly on the blockchain, specifying descriptions, deadlines, and rewards. Since all information is stored immutably, it remains accessible and cannot be altered unfairly.
- Deposit Locking for Secure Payments. To prevent non-payment issues, requesters must deposit the agreed-upon reward into the smart contract when posting a task. These funds are securely locked until the task is completed, ensuring that workers do not have to worry about payment reliability.
- Task Execution and Verification. Once a worker completes a task, the smart contract ensures that it meets the predefined requirements. Verification can be performed automatically through built-in logic or manually by designated reviewers. This reduces disputes and ensures that only valid work gets rewarded.
- Dispute Resolution Mechanisms. In cases of disagreement, smart contracts can include arbitration processes, where a neutral party or a decentralized voting system determines the outcome. This prevents unfair treatment and removes the risk of centralized platform bias.
- Automated Payment Processing. When a task is verified as complete, the smart contract instantly transfers the payment to the worker without delays or intermediaries. This creates a seamless and transparent payment process, making crowdsourcing more efficient and accessible.

By integrating these functions, smart contracts enhance trust, reduce administrative overhead, and empower both workers and requesters with greater autonomy.

SMART CONTRACT ARCHITECTURE FOR CROWDSOURCING

A smart contract for crowdsourcing serves as the backbone of a decentralized task management system, ensuring that tasks are created, verified, and rewarded fairly. At its core, the contract records all essential details, such as task descriptions, deadlines, and rewards, directly on the blockchain. This guarantees that once a task is published, its terms cannot be altered or manipulated, providing transparency for both requesters and workers.

One of the most important aspects of this system is the verification process. When a worker completes a task, the smart contract must determine whether it meets the required conditions. This can be handled automatically through pre-programmed logic or manually by designated reviewers. Automation is particularly useful for structured tasks, such as data validation or code submissions, while manual review is more suitable for subjective evaluations like content moderation or design work.

To further enhance trust, smart contracts also handle payment security. Unlike traditional platforms where workers rely on intermediaries for payouts, a blockchain-based system requires requesters to lock funds into the contract when they post a task. This ensures that once verification is complete, the payment is released instantly without delays or the risk of non-payment. Since transactions are executed on the blockchain, the entire process remains secure, transparent, and resistant to fraud.

By integrating task management, verification, and automated payments, smart contracts eliminate many inefficiencies found in traditional crowdsourcing models. They provide a streamlined, fair, and self-sustaining ecosystem where all participants interact with confidence, free from the risks of manipulation or delayed transactions.

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract Crowdsourcing {
struct Task {
   address creator;
   address worker;
   string description;
   uint256 reward;
   bool completed;
   bool disputed;
}
mapping(uint256 => Task) public tasks;
mapping(address => uint256) public ratings;
uint256 public taskCount;
function createTask(string memory _description, uint256 _reward) public payable
{
   require(msg.value == _reward, "Reward must be deposited");
tasks[taskCount] = Task(msg.sender, address(0), _description, _reward, false,
false);
   taskCount++;
   }
function acceptTask(uint256 _taskId) public {
   require(tasks[_taskId].worker == address(0), "Task already taken");
   tasks[_taskId].worker = msg.sender;
}
```

```
function completeTask(uint256 _taskId) public {
   require(msg.sender == tasks[_taskId].worker, "Only worker can complete");
   tasks[_taskId].completed = true;
   ratings[msg.sender] += 1;
   payable(tasks[_taskId].worker).transfer(tasks[_taskId].reward);
}
function disputeTask(uint256 _taskId) public {
   require(msg.sender == tasks[_taskId].creator, "Only creator can dispute");
   tasks[_taskId].disputed = true;
}
                                                    }
```

ECONOMIC MODEL AND TOKENOMICS

The economic structure of a blockchain-based crowdsourcing system relies on a carefully designed token model that ensures fair compensation, incentivizes quality work, and discourages fraudulent activity. Task pricing can be either fixed, where requesters set a predefined reward, or dynamic, adjusting based on factors such as task complexity, demand, or worker reputation. This flexibility allows the system to accommodate both simple microtasks and highly specialized work that requires greater expertise.

A key feature of this model is the reputation system, where workers build credibility over time by successfully completing tasks. Their performance history is stored on the blockchain, making it transparent and verifiable. Higher-rated workers may gain access to premium tasks with better rewards, while requesters can use reputation scores to identify reliable contributors, reducing the risk of low-quality submissions.

To further enhance security and prevent abuse, the system can implement staking mechanisms, requiring workers to deposit a small amount of collateral before taking on tasks. If a worker is found to be engaging in fraudulent behavior—such as submitting incomplete or low-effort work—their staked amount may be partially or fully forfeited. This creates an additional layer of accountability, ensuring that participants act in good faith while discouraging spam or automated task farming.

By integrating dynamic pricing, reputation tracking, and staking, the tokenomics of a blockchain-based crowdsourcing platform creates an ecosystem that is self-regulating, transparent, and fair for both workers and requesters.

CONCLUSION

The integration of smart contracts into crowdsourcing platforms brings greater transparency, security, and efficiency by eliminating unnecessary intermediaries and automating critical processes. By ensuring that all transactions are recorded on the blockchain, smart contracts create a tamper-proof and trustless environment where task assignments, payments, and dispute resolutions are handled with minimal human intervention.

Furthermore, the introduction of reputation-based ratings, arbitration mechanisms, and staking systems strengthens the platform's resistance to fraud and low-quality contributions. Workers are incentivized to maintain high performance, while requesters can confidently engage with a decentralized system that ensures fair treatment for all participants.

As blockchain technology continues to evolve, decentralized crowdsourcing has the potential to redefine the future of work, offering a more inclusive, efficient, and secure alternative to traditional platforms.

REFERENCES

1. R. Zhang, R. Xue, and L. Liu, "Security and privacy on blockchain," ACM Comput. Surv., vol. 52, no. 3, Art. 51, pp. 1–34, Jul. 2019. [Online]. Available: https://doi.org/10.1145/3316481

2. M. Li, J. Weng, A. Yang, W. Lu, Y. Zhang, L. Hou, J. Liu, Y. Xiang, and R. H. Deng, "CrowdBC: A blockchain-based decentralized framework for crowdsourcing," IEEE Trans. Parallel Distrib. Syst., vol. 30, no. 6, pp. 1251–1266, Jun. 2019. [Online]. Available: https://ink.library.smu.edu.sg/sis_research/4625

3. Y. Ma, Y. Sun, Y. Lei, N. Qin, and J. Lu, "A survey of blockchain technology on security, privacy, and trust in crowdsourcing services," World Wide Web, vol. 23, no. 3, pp. 1123–1143, Sep. 2019. [Online]. Available: https://doi.org/10.1007/s11280-019-00735-4

4. S. N. Khan, F. Loukil, C. Ghedira-Guegan, E. Benkhelifa, and A. Bani-Hani, "Blockchain smart contracts: Applications, challenges, and future trends," Peer-to-Peer Netw. Appl., vol. 14, pp. 2901–2925, Apr. 2021. [Online]. Available: https://doi.org/10.1007/s12083-021-01127-0

5. A. Singh, R. M. Parizi, Q. Zhang, K.-K. R. Choo, and A. Dehghantanha, "Blockchain smart contracts formalization: Approaches and challenges to address vulnerabilities," Comput. Secur., vol. 88, Art. 101654, Oct. 2019. [Online]. Available: https://doi.org/10.1016/j.cose.2019.10165

СМАРТ-КОНТРАКТЫ ДЛЯ УПРАВЛЕНИЯ ЗАДАНИЯМИ В КРАУДСОРСИНГЕ

Каирбек Нурбай Талгатулы
Научный руководитель: Алия Нугуманова

Традиционные краудсорсинговые платформы полагаются на централизованных посредников для управления заданиями, что приводит к дополнительным затратам, задержкам и рискам для безопасности. Смарт-контракты предлагают децентрализованную альтернативу, автоматизируя

создание, выполнение и проверку заданий, обеспечивая прозрачность и надёжность без участия третьих сторон. В данной статье рассматривается применение смарт-контрактов для управления заданиями в краудсорсинге, обсуждается их архитектура, ключевые компоненты, существующие решения, экономическая модель и возможные улучшения.

Ключевые слова: блокчейн, смарт-контракты, краудсорсинг, децентрализованное управление заданиями

РАУДСОРСИНГТЕГІ ТАПСЫРМАЛАРДЫ БАСҚАРУҒА АРНАЛҒАН СМАРТ-КОНТРАКТТАР

Каирбек Нурбай Талғатулы

Ғылыми жетекші: Алия Нугуманова

Дәстүрлі краудсорсинг платформалары тапсырмаларды басқару үшін орталықтандырылған делдалдарға сүйенеді, бұл қосымша шығындарға, кідірістерге және қауіпсіздік тәуекелдеріне әкеледі. Смарт-контракттар тапсырмаларды құруды, орындауды және тексеруді автоматтандырып, үшінші тараптардың қатысуынсыз ашықтық пен сенімділікті қамтамасыз ете отырып, орталықтандырылмаған балама ұсынады. Бұл мақалада краудсорсингтегі тапсырмаларды басқару үшін смарт-контракттарды қолдану қарастырылады, олардың архитектурасы, негізгі компоненттері, қолданыстағы шешімдер, экономикалық моделі және ықтимал жетілдірулер талқыланады.

Кілт сөздері: блокчейн, смарт-контракттар, краудсорсинг, тапсырмаларды орталықтандырылмаған басқару.