A Scoring System For Multiplayer Game Base On Blockchain Technology

Yunifa Miftachul Arif
Department of Informatics Engineering
Universitas Islam Negeri Maulana
Malik Ibrahim
Malang, Indonesia
yunif4@ti.uin-malang.ac.id

Muhammad Naufal Firdaus
Department of Informatics Engineering
Universitas Islam Negeri Maulana
Malik Ibrahim
Malang, Indonesia
16650013@student.uin-malang.ac.id

Hani Nurhayati
Department of Informatics Engineering
Universitas Islam Negeri Maulana
Malik Ibrahim
Malang, Indonesia
hani@ti.uin-malang.ac.id

2021 IEEE Asia Pacific Conference on Wireless and Mobile (APWiMob) | 978-1-7281-9475-2/21/$31.00 ©2021 IEEE | DOI: 10.1109/APWIMOB51111.2021.9435249

***Abstract*—An essential part of game development, especially in the multiplayer genre, is the scoring system. Data assurance on each device and data security is a significant issue in developing the assessment system. In this paper, our scoring system creates with blockchain-based data sharing for multiplayer games. We implemented the data design on the Unity game engine using the ethereum framework. The test results show that each device can connect to the network and can display the player's update score via the leaderboard. Some ethereum variables that affect the score data transaction process include gas prices, gas limits, and the gas used. The transaction speed test results show that the faster the gas limit value, the faster the transaction validation process will make the data transaction process faster. In this study, the gas limit value 500000 is the most optimal to get player score data transactions. While the average cost incurred for each data delivery score is 0.00996434 ETH.**

Keywords—scoring system, multiplayer game, data sharing, blockchain

I. Introduction

The scoring system is an essential part of game mechanics, especially for multiplayer games. The game can reward each player for a task or mission that has been completed [1]. Visualization of the score in the game includes points, virtual currencies, or badges [2]. The existence of a score in the game will stimulate sportsmanship and competition between players. On the other hand, the award in the form of a score can motivate and increase the player's fun [3]. Games with good score visualization can also increase the challenge for players to get the highest score. On the other hand, the score's visualization can measure the game's educational aspect [4]. Therefore, the score information that is always updated on the player's device through the data-sharing system is essential besides the data security factor.

In multiplayer games, player score data is generally stored in a central cloud server database. Each player device must have access rights to the database server to get the score data update. This concept is also known as centralized architecture, where every data transmission in the network must go through the middle node [5]. The main drawback is that if the central node has a problem, it will affect network capability and performance [6]. A scoring system requires developing a better data sharing architecture by using the decentralized concept to improve the score player data's circulation capability. Each node can send data directly without waiting for the central node's approval [7][8]. This advantage helps the scoring system ensure data availability at each node without depending on the central node. In other words, the score data will be stored on each player's device.

Another problem with data sharing systems that are commonly used in game development today is a threat to user data privacy, where users do not know how to manage and distribute data by the manager. Privacy is a user's right to maintain confidentiality and control over the information they have when given to other parties. Therefore, data sharing in the scoring system should also have the ability to protect data players as game users [9]. One of the new technology developments in data sharing systems with a decentralized concept with security priorities is blockchain [10]. Apart from being decentralized and better security, blockchain also has several characteristics, including immutability, anonymity, and increased capacity [11]. Apart from handling cryptocurrency on Bitcoin [12], blockchain can also be implemented in other fields, for example, tourism rating data sharing [6], transaction simulation in a serious game [13], and medical data sharing [14].

In this study, blockchain is proposed to handle data circulation in the game scoring system to be accessed and visualized directly on each player's device while still paying attention to data security factors. Furthermore, we use Ethereum as a platform to implement blockchain technology into the game scoring system. Ethereum is a smart contract-based blockchain technology platform [15]. In this study, the platform was integrated into a tourism-themed game built using the Unity Game Engine. In the testing phase, we analyzed data score transactions' success and speed between players based on changes in the value of several variables used in the platform.

We divide the content paper into several sections to explain the scoring system's research steps in this multiplayer game. Besides the Introduction, this paper also has other supporting sections, including Relate Work System Design and Blockchain Architecture, Result and Discussion, and Conclusion.

II. Related Work

The following are some studies that discuss scoring systems and blockchain implementation for games. First, paper [16] proposes a scoring system in human computation games. This study's scoring system uses the concept of multiple rewards implemented in a cooking-themed game, namely Cafe Flour Sack. In the experimental stage, this research uses two versions of the distribution of rewards, namely randomly or offered to players. Next, paper [3] proposes a concept scoring system design for videogame reward systems. The purpose of developing the reward system is to motivate and change players' behavior towards game content and increase pleasure through obtaining rewards and mechanisms. The paper also discusses the classification and characteristics of rewards suitable for players. This classification includes advancement, review, sociality, and cooperation. On the other hand, the paper [2] discussed implementing the scoring and reward system for health games. The goal is to increase player satisfaction to be motivated to use the game application regularly.

There is a paper that discusses the implementation of blockchain technology to support the development of computer games. However, no one has used it for developing data sharing on a scoring system. Paper [13] discussed blockchain in the tourism serious game transaction system with Bromo Mountain's theme. To implement blockchain technology in game development through a game engine, the author uses the ethereum platform. The research states that Ethereum blockchain has been successfully used to support the transaction system's visualization, although it is not clear how the smart contract model is used. In another study, paper [17] discusses transaction fees from implementing blockchain in queuing game on Bitcoin played by mainer and user. Transaction fees play an important role in ensuring the queuing game runs. Next on the paper [18], the concept of data-sharing on the blockchain is used to support the security system in overcoming cheats in multiplayer online games. Blockchain has a better data security system, so it is also used to increase security in multiplayer online games discussed in the paper [19].

III. System Design and Blockchain Architecture

In this research, we developed a blockchain-based data sharing for scoring data between players. A proposed system design consisting of several parts is seen in Figure 1. The interconnected parts include the player, scoring system, data sharing using ethereum blockchain, ethereum blockchain network, and other player devices.

Players use their devices, such as smartphones, tablets, or computers, to play games. We developed the game visualization, including a scoring system on the user's device, using the Unity game engine. As the script editor, we use Visual Studio to create an ethereum blockchain-based transaction framework and Metamask as the blockchain wallet's dashboard. The explanation is, this product is open source with a straightforward UI. This research utilizes the ethereum blockchain network to associate every player's gadget to connect with others. Furthermore, this paper describes the research steps of designing a game flow, score calculation, and score data sharing based on ethereum blockchain.

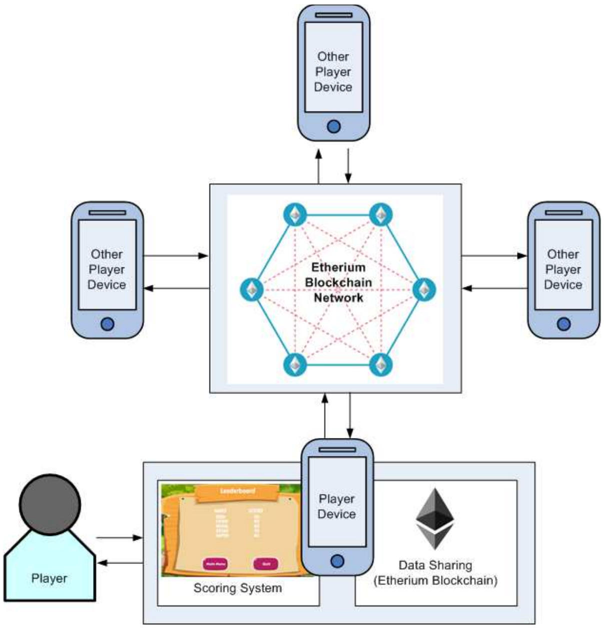


Figure 1: Proposed system design

A. Game Flow and Score Calculation

Figure 2 shows the game flow that the player must go through to get the score visualization. The game flow is divided into four parts, including the Menu, Blockchain Connection, Gameplay, and Scoring System. After the main menu displays, the process begins with a connection between the ethereum blockchain network and the player device. Furthermore, when the player presses the check balance button to find out the remaining balance, the system will retrieve the login's wallet address. After the player has successfully logged in, they can start playing the game at the specified time. Furthermore, the player's score data when collecting score points is sent to the etheruem blockchain network. Data is sent in the form of a transaction, with the remaining balance displayed on the player user interface while playing the game.

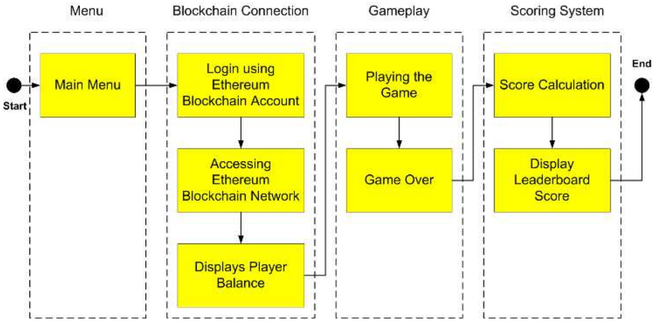


Figure 2: Game Flow

The calculation of the score system points displayed on the game leaderboard is shown in equation 1. The player point score PS is the result of the points that the player gets by dividing between the points P with time T. Point P is the number of points collected, and time T is when it takes the player to collect points.

$$PS = P/T \tag{1}$$

From the PlayerPoint results obtained by all players, the system looks for the largest value shown by Equation 2. WhereMS is the result of the calculation of all player points. While MAX is the highest value search function and APS are all the points the player gets.

$$MS = MAX(APS) \tag{2}$$

The formula used is to find the MAX value of all the points the players get. For each apple, the player has collected = + 5 points. The points earned are then stored by the system and displayed again to the player in the highest score leaderboard. The time variable is used as a differentiator and determines the highest score on the leaderboard to avoid the similarity of each player's points.

B. Blockchain-Based Data Sharing

In this study, we used Photon 2 and Blockchain as the Networking Engine Platform. Photon has many product branches that can be used for various devices and platforms. The product that will be used in this research is PUN (Photon Unity Networking). PUN uses the central server of Photon Networking to regulate the current data sharing flow. The blockchain network used in this study is the blockchain with the ethereum platform. Because the ethereum network's nature is public and peers to peer network, all connected nodes will know what data is being shared on the network. PUN and ethereum blockchain's function in this game is for online multiplayer games as data sharing between the system and players. When the game starts, the player's movements and interactions will be synchronized with the Photon Network.

Furthermore, each score generated by the player will be stored and processed with the Ethereum blockchain. The data shared via the blockchain network in this data-sharing game system is the player's score data. The score includes all the points the player gets when picking up objects in the form of apples scattered throughout the game.

Blockchain also supports transparency and long-term data storage on the network, meaning that this data can be seen by all computers connected to the network as long as there are still computers connected to the network. Data security is maintained because of the encryption process that covers it, unlike the central server network in general, which can only access data when using the central server network. The following is the flow of the application of blockchain technology for data sharing scores in this study.

- The process of forming a blockchain is due to transaction agreements between all connected nodes in the network. The initial agreement is written in a smart contract algorithm that contains program code that has met the terms of the transaction agreement.
- When the conditions in the smart contract are fulfilled, the first transaction will occur.
- In this study, these conditions are met when the player has accumulated all scores or time runs out, so the transaction occurs.
- The block creation process is carried out after the smart contract is running. A block has several parts, including the block index, block hash, previous block hash, transaction root hash, receipt root hash, timestamp, difficulty, nonce, gas limit, and gas. Each block is connected to the next block based on the similarity of the block hash's value and the previous hash in the next block [8].
- When the initial transaction occurs, the system uses the SHA-256 hashing algorithm. This algorithm is a hashing algorithm that converts an input into a 256-bit message. The initial process of hashing SHA-256 is message padding, where the insertion of the number 1 and the addition of bit 0 make the message congruent to 448 modulo 512. The second stage is parsing, when a message has previously been padded and becomes N 512 blocks bit. The next process is solving each block into 16 32 bit words that are expanded to 64 words called Message Expansion. 64 words that have been formed are then labeled, then processed with the SHA-256 function, which will get 8 variables that are given initial values for each function. The SHA-256 hashing process's final result is obtained from combining 8 computed variables [13][20].
- Transactions are stored in the initial block if no new transactions are made.
- However, if a new transaction is formed, the SHA-256 hashing process will be carried out again, forming a new block.
- The transaction, in this case, is the Score player value sent to the Blockchain network.
- Data is already stored on the Ethereum Blockchain network.
- Transactions are said to be successful if they have been verified in the blockchain network.

In this study, when the node has obtained the score, the system automatically sends the score data to the blockchain network. The smart contract runs to determine whether the conditions have been met. When not completed, the system will send an error message. Conversely, when the smart contract is fulfilled, the system will send data to the ethereum blockchain network. The next process is to form a block structure for the data, followed by the hashing process to produce new transactions. After being validated with Proof of Work (PoW), the block will be published on the blockchain network. Furthermore, through the blockchain network, player nodes get a score and the order of their position. The system then updates the remaining player balance.

C. Smart Contract

The existence of a smart contract is one of the advantages of the ethereum framework. Figure 3 shows smart contract creation flow to support the data sharing systemin of the ethereum framework [13]. We use ABI and byte code to create and access smart contracts. ABI and byte code are obtained from the compile process of the smart contract that was written previously. In this study, the data sent to the ethereum blockchain network is score data, so it is necessary to determine the right smart contract for storing the score data. Program Code 1 shows the smart contract algorithm that has been written in Solidity for the highest score, which is then compiled at www.remix.org.

978-1-7281-9475-2/21/$31.00 ©2021 IEEE

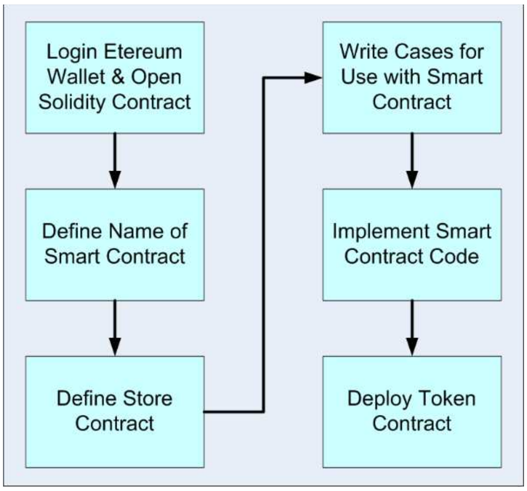

Login Etereum Wallet & Open Solidity Contract
Define Name of Smart Contract
Define Store Contract
Write Cases for Use with Smart Contract
Implement Smart Contract Code
Deploy Token Contract

Figure 3: Smart contract creation flow

Program Code 1 Smart Contract in Solidity

```
pragma solidity ^0.4.10;
contract PlayerScore {
    uint maxTopScores = 40;
    address owner;
    struct TopScore{
        address addr;
        int score;
    }
    function PlayerScore(){
         owner = msg.sender;
    }
    TopScore[] public topScores;
    mapping (address=>int) public userTopScores;
    function setTopScore(int256 score, uint8 v, bytes32 r, bytes32 s)
        var hash = sha3(msg.sender, owner, score);
        var addressCheck = ecrecover(hash, v, r, s);
        if(addressCheck != owner) throw;
        var currentTopScore = userTopScores[msg.sender];
        if(currentTopScore < score){
            userTopScores[msg.sender] = score;
        }
        if(topScores.length < maxTopScores){
            var topScore = TopScore(msg.sender, score);
            topScores.push(topScore);
        }
        else{
            int lowestScore = 0;
            uint lowestScoreIndex = 0;
            for (uint i = 0; i < topScores.length; i++)
        {
        TopScore currentScore = topScores[i];
        if(i == 0){
            lowestScore = currentScore.score;
            lowestScoreIndex = i;
        }
        else{
            if(lowestScore > currentScore.score){
               lowestScore = currentScore.score;
               lowestScoreIndex = i;
             }
        }
       if(score > lowestScore){
           var newtopScore = TopScore(msg.sender, score);
           topScores[lowestScoreIndex] = newtopScore;
        }
    }
    function getCountTopScores() returns(uint) {
        return topScores.length;
    }
}
```

IV. Result And Discussion

This chapter discusses the results of implementing a scoring system in a multiplayer game entitled Widow Waterfall. Our chapter also discusses testing data sharing results using the ethereum blockchain platform, including connection testing, gas limit testing, speed transaction testing, and development costs testing. The following are some of the variables that can affect the test results, including gas prices, gas limits, and the gas used. Gas price is the potential cost for each data-sharing transaction. Meanwhile, the gas limit is the gas usage limit for each transaction, and the gas used is the amount of gas used for each transaction process. In the trial phase, we used a wallet account provided by www.metamask.io. As for the ether transaction network, we use the Kovan Test Network.

A. *Gameplay and Scoring System Visualization*

In this study, we built a multiplayer game with 3D visualization. Figure 4 shows an example of visualizing a virtual game environment for players built using the Unity game engine. When playing the game, each player can see the acquisition of the score they get through the score leaderboard. Figure 5 shows an example of visualizing the leaderboard score to find out the score for each player.

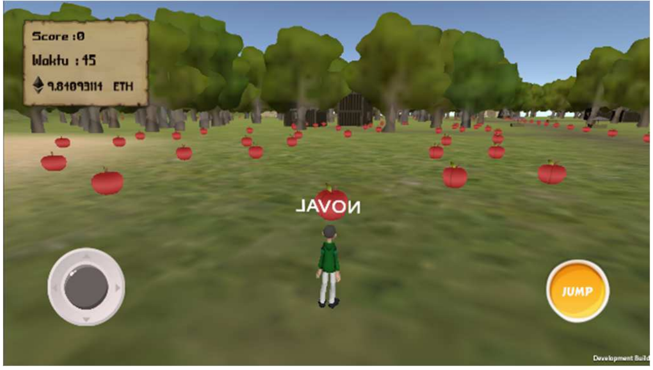
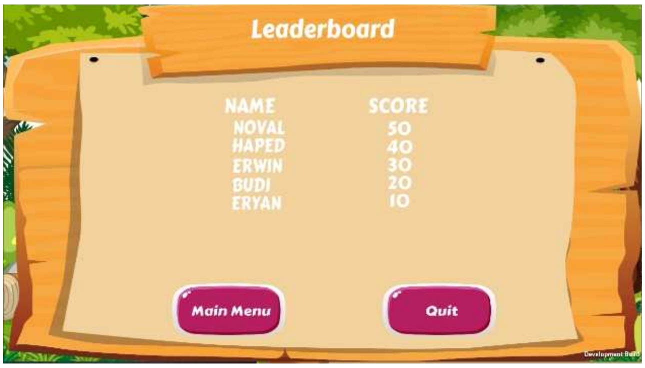


Figure 4: Virtual environment in the game



Leaderboard

NAME	SCORE
NOVAL	50
HAPED	40
ERWIN	30
BUDI	20
ERYAN	10

Figure 5: Leaderboard score

B. *Connection Testing*

This test is done to find out whether each device can be connected to the ethereum blockchain network. The connections that occur on each device are indicated by displaying the player's remaining Ethereum balance. The balance will be used to send and store score data obtained by players on the network. Testing was carried out using five different devices. The test is declared successful when the balance information display changes from "None" to "9,84093114 ETH". The number "9.84093114" is the remaining balance from the ethereum wallet and converted from the HexCode token "0x889202bb7ac7afd0" into a string that the system can read. Figure 6 shows five devices that have successfully connected to the ethereum blockchain network. Meanwhile, figure 7 shows that each device has successfully displayed a score update via the leaderboard.

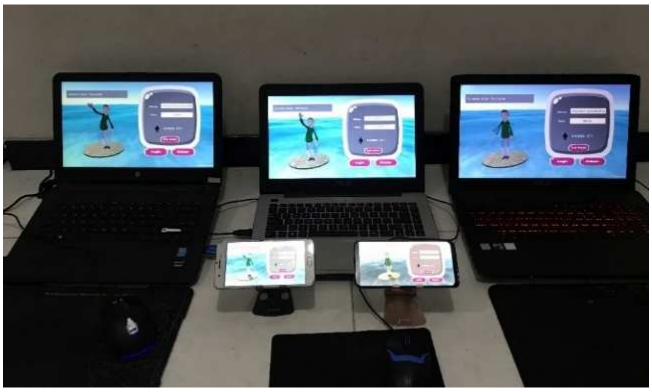


Figure 6: Connection testing to the ethereum network

Figure 7: Each device displays a leaderboard

C. Gass Limit Testing

Gas limit testing is carried out to determine the minimum gas limit used in sending score data as transactions in the widow waterfall game. Determination of the right gas is also a factor in the success of transactions on the ethereum blockchain network. Figure 8 shows the results of the gas limit testing used in the widow waterfall game research. The minimum gas limit can be used in transactions in the widow waterfall game with a maximum success rate of 500,000 gas.

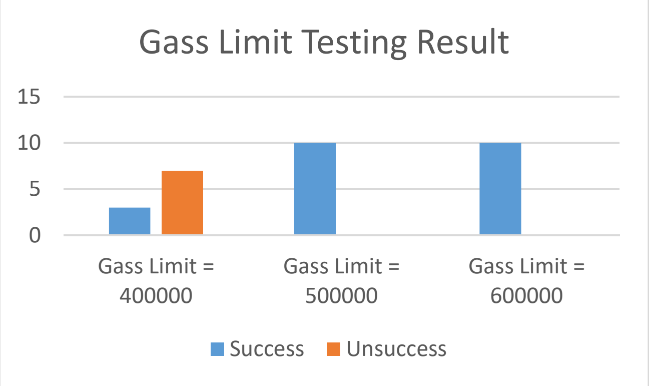


Figure 8: Gas limit testing result

D. Transaction Speed and Transaction Cost Testing

This test is done to determine how quickly a transaction will be confirmed by the Ethereum blockchain checking site, namely the https://kovan.etherscan.io/. Testing is done by changing the variable value of the gas limit that has been tested previously. Then perform transaction trials several times and calculate the average time it takes to process a transaction. According to [13], the formula for calculating the speed of a transaction is as follows.

$$\bar{T} = \frac{1}{n}\sum_{i=1}^{n} Ti \qquad (3)$$

In this study $\bar{T}$ represents the average time it takes to process a transaction. Meanwhile, Ti is the time required by the system to process transactions in the ethereum network. Next, n is the number of transaction experiments. The gas limit used in checking transaction speeds is 1,000,000, 900,000, 800,000, 700,000, 600,000 and 500,000. The gas limit value of 500,000 is the minimum value used to make transactions in the widow waterfall game. Figure 9 shows the time comparison based on a given gas limit. Where the greater the gas limit is given, the transaction validation process will also be faster.

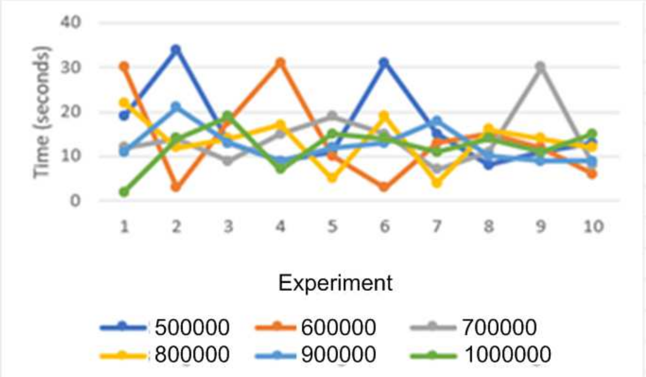


Figure 9: Transaction speed testing result

In this study, we also tested transaction costs. The goal is to find out the average cost that must be incurred for each score data transfer transaction in the game. As for the crypto ether dummy, we use Kovan Etherscan (KETH). Figure 10 shows the difference in ETH value in the wallet between before transactions and after transactions. From 10 sample transactions, each transaction requires an average of 0.00996434 ETH.

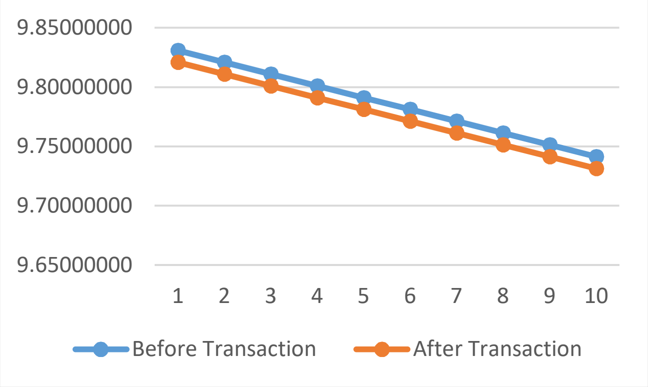


Figure 10: Transaction cost testing result

V. Conclusions

Through this paper, we propose a scoring system for multiplayer games. This data-sharing system uses the ethereum framework to implement the proposed blockchain technology on the Unity game engine. The test results show that each device can be connected to the network to see each player's updated score on the leaderboard. The transaction speed test results show that setting the gas price value influences each data score transaction's speed. The transaction processing time will be faster, the more significant the gas price value. Meanwhile, the gas limit test results show that the value of 500,000 is more optimal for use in data transactions. The average cost required for each data score transfer transaction is 0.00996434 ETH.

Acknowledgment

This research is supported by Multimedia Laboratory and Robotic Laboratory at Department of Informatics Engineering, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

References

[1] C.-I. Lee, I.-P. Chen, C.-M. Hsieh, and C.-N. Liao, "Design Aspects of Scoring Systems in Game," *Art Des. Rev.*, vol. 05, no. 01, pp. 26–43, 2017, doi: 10.4236/adr.2017.51003.

[2] M. M. Tizuka, E. W. G. Clua, and L. C. De Castro Salgado, "Investigating m-health gamification rewards elements for adults 50+," *2020 IEEE 8th Int. Conf. Serious Games Appl. Heal. SeGAH 2020*, no. August, 2020, doi: 10.1109/SeGAH49190.2020.9201658.

[3] H. Wang and C.-T. Sun, "Game Reward Systems: Gaming Experiences and Social Meanings," in *Digital Games Research Association DiGRA 2011 Conference*, 2012, pp. 1–15, doi: 10.1103/PhysRevB.51.8115.

[4] C. Harteveld and S. C. Sutherland, "The goal of scoring: Exploring the role of game performance in educational games," *Conf. Hum. Factors Comput. Syst. - Proc.*, vol. 2015-April, pp. 2235–2244, 2015, doi: 10.1145/2702123.2702606.

[5] D. Yaga, P. Mell, N. Roby, and K. Scarfone, "Blockchain technology overview," 2018. doi: 10.6028/NIST.IR.8202.

[6] Y. M. Arif, H. Nurhayati, S. Harini, S. M. Susiki Nugroho, and M. Hariadi, "Decentralized Tourism Destinations Rating System Using 6AsTD Framework and Blockchain," in *2020 International Conference on Smart Technology and Applications (ICoSTA)*, 2020, pp. 1–6, doi: 10.1109/icosta48221.2020.1570614662.

[7] D. Froelicher and P. Egger, "UnLynx : A Decentralized System for Privacy-Conscious Data Sharing," vol. 2017, no. 4, pp. 152–170, 2017, doi: 10.1515/popets-2017-0032.

[8] Y. M. Arif, H. Nurhayati, F. Kurniawan, S. M. S. Nugroho, and M. Hariadi, "Blockchain-Based Data Sharing for Decentralized Tourism Destinations Recommendation System," *Int. J. Intell. Eng. Syst.*, vol. 13, no. 6, pp. 472–486, 2020, doi: 10.22266/ijies2020.1231.42.

[9] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Q. Wu, "A survey of game theory as applied to network security," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, no. June 2014, 2010, doi: 10.1109/HICSS.2010.35.

[10] G. Zhang, T. Li, Y. Li, P. Hui, and D. Jin, "Blockchain-Based Data Sharing System for AI-Powered Network Operations," *J. Commun. Inf. Netw.*, vol. 3, no. 3, pp. 1–8, 2018, doi: 10.1007/s41650-018-0024-3.

[11] H. F. Atlam, A. Alenezi, M. O. Alassafi, and G. B. Wills, "Blockchain with Internet of Things: Benefits, challenges, and future directions," *Int. J. Intell. Syst. Appl.*, vol. 10, no. 6, pp. 40–48, 2018, doi: 10.5815/ijisa.2018.06.05.

[12] S. Nakamoto, "Bitcoin : A Peer-to-Peer Electronic Cash System," pp. 1–9, 2008.

[13] Y. M. Arif, R. P. Pradana, H. Nurhayati, S. M. S. Nugroho, and M. Hariadi, "A Blockchain-Based Multiplayer Transaction For Tourism Serious Game," in *International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*, 2020, pp. 138–143.

[14] X. Liu, Z. Wang, C. Jin, F. Li, and G. Li, "A Blockchain-Based Medical Data Sharing and Protection Scheme," *IEEE Access*, vol. 7, pp. 118943–118953, 2019, doi: 10.1109/access.2019.2937685.

[15] A. S. Bruyn, "Blockchain an Introduction," University Amsterdam, 2017.

[16] K. Siu and M. O. Riedl, "Reward systems in human computation games," *CHI Play 2016 - Proc. 2016 Annu. Symp. Comput. Interact. Play*, pp. 266–275, 2016, doi: 10.1145/2967934.2968083.

[17] J. Li, Y. Yuan, S. Wang, and F. Y. Wang, "Transaction Queuing Game in Bitcoin BlockChain," *IEEE Intell. Veh. Symp. Proc.*, vol. 2018-June, no. Iv, pp. 114–119, 2018, doi: 10.1109/IVS.2018.8500403.

[18] S. Kalra, U. C. Berkeley, and R. Sanghi, "Blockchain-based Real-time Cheat Prevention and Robustness for Multi-player Online Games," in *The 14th International Conference on emerging Networking EXperiments and Technologies (CoNEXT '18)*, 2018, pp. 178–190.

[19] S. P. Dhevi, A. S. Subash, and N. Harini, "Security In Online Games Using Blockchain," vol. 119, no. 15, pp. 1661–1669, 2018.

[20] D. Michel, "Conception of a Blockchain Client for secure transmission of production data," University of Stuttgart, 2018.