Prototype Blockchain Based Smart Contract For Freelance Marketplace System

by Sufa Atin

Submission date: 05-Mar-2024 03:51AM (UTC+0700)
Submission ID: 2311683471
File name: FullPaper-ICIC-2021_89-Irawan-UNIKOM-CameraReady.pdf (1.17M)
Word count: 4018
Character count: 22259

Prototype Blockchain Based Smart Contract For Freelance Marketplace System

Irawan Afrianto
Informatics Engineering Dept
Universitas Komputer Indonesia
Bandung, Indonesia
irawan.afrianto@email.unikom.ac.id

Christover Ramanda Moa
Informatics Engineering Dept
Universitas Komputer Indonesia
Bandung, Indonesia
moachristover@gmail.com

Sufa Atin
Informatics Engineering Dept
Universitas Komputer Indonesia
Bandung, Indonesia
sufaatin@email.unikom.ac.id

***Abstract*— Freelancing marketplace is a site or platform that connects two parties in processing service transactions at an hourly rate or per project. A conventional freelancing marketplace is a place for freelancers to find work and transact digitally. This study aims to propose a prototype of a freelancing marketplace system that is distributed and decentralized, secure, and transparent using smart contract-based blockchain technology. The method used in this study is a prototype which is a fast method of developing a software system. The developed prototype is a system that is based on the Ethereum public blockchain network, utilizes a smart contract mechanism in its transaction activities, and use IPFS in the storage and sharing of documents on it. According to the findings of the research, transaction data input in the freelancing marketplace system prototype environment can be executed by smart contracts and saved on the blockchain network, indicating that the transaction data will be stored more securely, tamper proof, and transparent.**

Keywords*— *blockchain, smart contract, prototype, freelance marketplace system

I. INTRODUCTION

Freelance is a freelance job that works without being tied to a company contract for an extended period. On the one hand, freelancers are employees because the company almost always hires them for periods that do not sell other than intangible professional knowledge [1]. Globally, India has the second largest freelance workforce after the United States [2]. Due to the significant increase, the freelancing marketplace users indirectly increase, leading to data and transaction vulnerability levels. Freelancing marketplaces are websites that match buyers of services sent electronically with sellers or freelancers who offer services on a per-job basis or at a fixed hourly rate [3]. However, the current freelancing system is still based on a centralized system, giving rise to problems with single failure, less transparent processing mechanisms, large payments to third parties, and often late payment processes [4].

On the other hand, various industrial technologies are starting to take advantage of blockchain technology, which is decentralized [5]. Blockchain is a digital transaction record, where individual records from users, called blocks, are linked together in one place, called a chain [6][7]. Blockchain technology uses the hash algorithm function to generate wallet addresses, namely Keccak-256. The excess of the hash function will result in a unique character set called a message digest [8]. The hash value in the calculated hash function algorithm has a size of 32 bytes and is represented using hexadecimal numbers (64 characters) [9]. The blockchain decentralized system runs on a peer-to-peer network to verify and manage each block using automation and governance protocols simultaneously [10], using a consensus mechanism in verifying transactions[11], as well as applying smart contracts to manage the transactions it does. [12]. A smart contract is a protocol that runs on top of the blockchain, which contains a set of rules by which the smart contract parties agree to interact under certain conditions [13][14]. The application of blockchain technology must look at the business processes that will be carried out, whether it will be implemented in a public blockchain environment that can be accessed by all participants[15], or made more closed by using a private blockchain that can only be accessed by a limited number of participants[16]. Researches on blockchain-based freelance systems that have been carried out are used to determine the function of trust and transparency[17], a more limited freelance model using hyperledger[18] and a prototype of an open Ethereum-based freelance system[19].

The purpose of this study is to propose a prototype model of a freelance market system using blockchain technology based on smart contracts. where this system will facilitate transactions between project owners and freelancers to be able to transact more securely, faster, more transparently, and eliminate the role of third parties in the transaction.

II. RESEARCH METHODOLOGY

According to Ogedebe [20], prototyping begins with gathering requirements, involving system developers and users to determine the objectives, functions and operational requirements of the system. The steps in prototyping are as follows: 1. Gathering Requirements. 2. Fast design process. 3. Build a prototype. 4. Evaluation and improvement. Gathering requirements involves meeting between developers and customers to determine the overall purpose of the software; identify needs in the form of an outline of the basic requirements of the system to be made. Design focuses on the representation of aspects of the software from the user's point of view; It includes input, process and output formats. Rapid design leads to the construction of a prototype, the prototype is evaluated by the user and the design analyst and is used to adjust the needs of the software to be developed. prototype is set to meet user needs, and at that time the developer understands more clearly and in detail what needs to be done. After the four prototyping steps are carried out, the next step is the manufacture or design of the actual product. The research flow can be seen in figure 1.

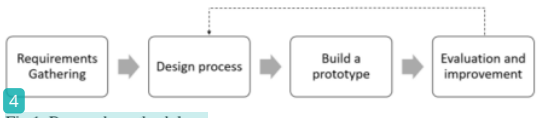

Fig.1. Research methodology

III. RESULT AND DISCUSSION

A. *System Architecture Analysis*

The system is developed based on web applications. The web application can be accessed anywhere and anytime, as long as the user is connected to the internet [21]. System architecture analysis is a system analysis carried out on a proposed application that aims to identify the built application system and provide a clear and overall picture of the application system. The following is an analysis of general system architecture (figure 2). The system's general architecture has 36 stages starting from the project owner posting projects/jobs to receiving work results files.

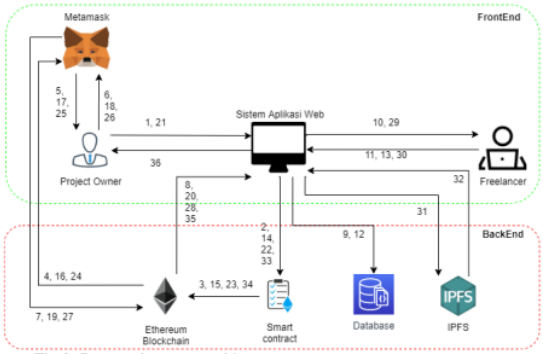

Fig.2. Proposed system architecture

The following is an explanation of the general system architecture starting from creating and posting projects until the work contract is completed according to figure 2. The project owner can create a project by posting it.The create / post project function can interact with smart contracts using Web3js so that data can be stored on the Ethereum Blockchain network. Data that can be stored on the Blockchain network include project titles, descriptions, estimates, budgets, categories, and work or project levels.The project creation / posting form data will be stored on the Ethereum Blockchain network. The Ethereum Blockchain requests confirmation from the Metamask wallet so that the project data is actually stored. The Chrome Metamask extension will pop up a confirmation dialog on the user's screen. The user i.e. project owner confirms the metamask dialog to save the post data to the Ethereum Blockchain network. Once confirmed, the data is successfully stored on the Ethereum blockchain network. The project listing will be displayed on the website page. Data for creating / posting projects is also stored in the database. Freelancers can view projects that have been posted and stored on the network and make selections according to their criteria and capacity. After the project that fits the capacity is obtained, the freelancer will comment on the project. Comments from freelancers will be stored in the database. Freelancers can also make an offer in advance with a fee rate that is set on their service. The offer provided by the Freelancer will interact with the Smart Contract. The bidding data will be stored back on the Ethereum Blockchain network. Ethereum Blockchain will re-verify dan the confirmation pop up on Metamask will be repeated. Confirmation will be received by freelancer. Confirmed offers and offer data will be stored on the Ethereum Blockchain network. Offering data will be displayed on the website page. The project owner can choose the offer of workers or freelancers according to the budget and the required criteria. Then the Project owner can create a single contract with the selected Freelancer worker and the inputted data will interact with the smart contract using Web3js. The contract data made include the period of the initial down payment (DP) to increase the sense of high trust in freelancers. Contract data that has been created will be stored on the Ethereum Blockchain network after being confirmed by both parties, namely the project owner and reelancer. The verification using Metamask and save confirmation pop up will appear in the dialog window. Confirmation of contract data storage from the Project owner who made the contract. Contract creation confirmed by project owner and contract will be kept until adjusted deadline after also confirmed by freelancer. The system can display a contract url link that can be shared and confirmed by freelancer workers to start the contract. Contract confirmed by freelancer so the deadline is running. Contracts stored on the Ethereum Blockchain network. After the deadline is over, freelancers can start uploading their work files. Work files are stored in the Interplanetary File System (IPFS) file distribution. IPFS will return the hash of each file uploaded on its distributed system to the system. The application system will forward the hash of the file that has been stored on IPFS to the Ethereum Blockchain network to be stored again. Hash file documents are stored on the Ethereum Blockchain network. Immediately after the term of the contract expires. The system will indicate that the contract is complete and the work fee that has been deposited will be automatically withdrawn to the freelancer worker's account. The results of the work can be downloaded by the project owner.

- **Sub-system architecture of create/post projects:** The project owner first creates or posts a project listing to the web application system. The system will interact with the smart contract using Web3js. The data on the completed form is saved to the Ethereum Blockchain network. The form data that is filled in is also stored in the database for the purposes of the website application system. (figure 3).

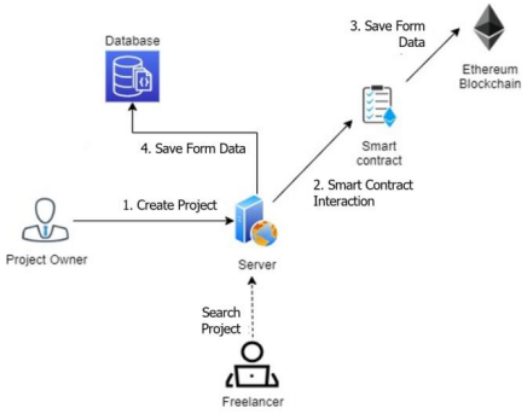

Fig.3. Create/post projects sub-system architecture

- **Sub-system architecture of bid projects**: Freelancers can bid on projects of interest and according to their capacity. The system will interact with the smart contract to execute the functions in the Smart Contract. Offer data is stored on the Ethereum Blockchain network (figure 4).

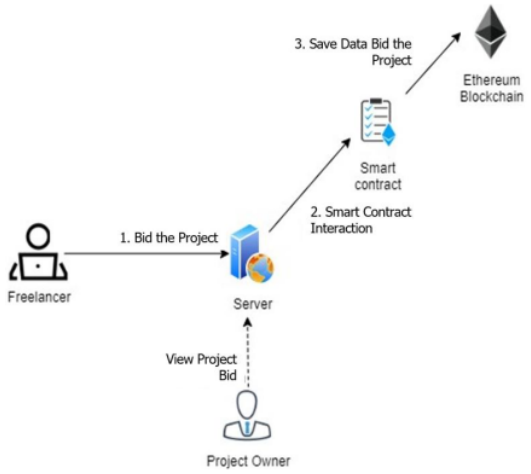

Fig.4. Project bidding sub-system architecture

- **Sub-system architecture of create project contracts**: After finding Freelancer workers who match the criteria for the project created, the project owner can make a work contract by filling out the form first, namely the end period and the initial down payment (DP). Once created, the Project Owner can share the contract link link to freelancers to be examined first before confirming the work. After both parties agree, the server on the system will interact with the functions in the smart contract using web3js. The project contract data is stored on the Ethereum Blockchain network (figure 5).

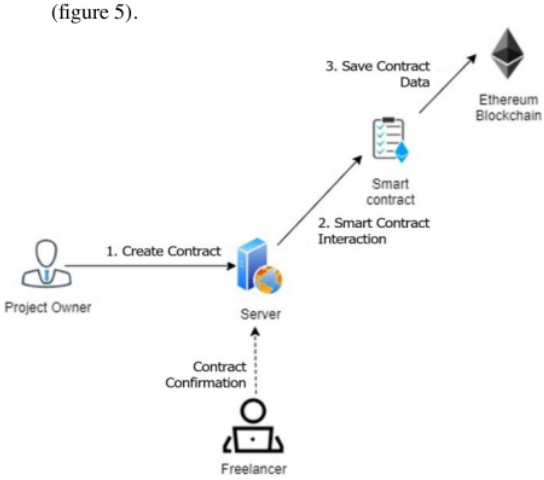

Fig.5. Creating project contracts sub-system architecture

- **Sub-system architecture of the store / sharing file**: After the specified time period has been completed, Freelancers can upload their work files to the system.The system will forward the work files to a decentralized file distribution based on Blockchain technology, namely IPFS (Interplanetary File System) and stored there. IPFS will return the hash of the files and documents that have been saved to the system. The website application system will forward the saved hash file to the smart contract to be stored on the Ethereum Blockchain network (figure 6).

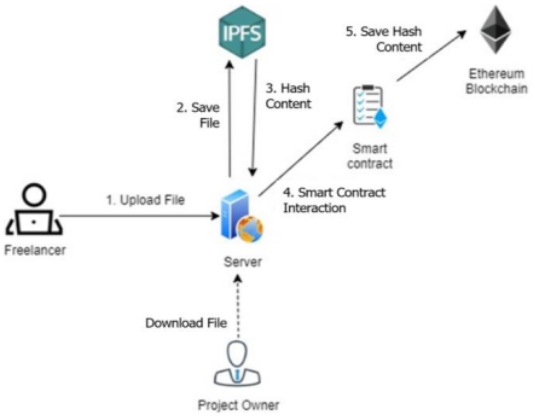

Fig.6. Store / sharing file sub-system architecture

B. Smart Contract Analysis

A smart contract implements code on the blockchain that aims to bind an agreement or agreement between several parties. Smart contracts play an important role in transactional system activities that are built. This is because smart contracts are an integral part of each activity carried out between actors in the system. *BuatProject.sol* is a smart contract created to facilitate project creation functions performed by project owners so that they can be stored in a blockchain environment. Meanwhile, *Penawaran.sol* is a smart contract that is used by project owners and freelancers to carry out bid-making and project bid-taking activities. *Kontrak.sol* is a smart contract that contains activities for making project contract agreements, including the deposit mechanism, sending work results to processing payments, all of which are stored in the blockchain. The relationship between the three smart contracts can be seen in figure 7.

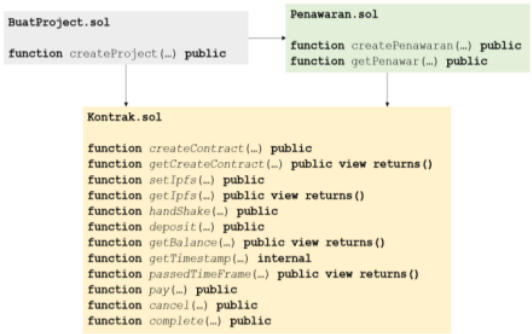

Fig.7. Smart contracts architecture

Smart contract interaction in the project creation process carried out by the project owner with the project module, ganache as a blockchain local environment platform and metamask as a crypto wallet for payment for project creation transactions (figure 8).

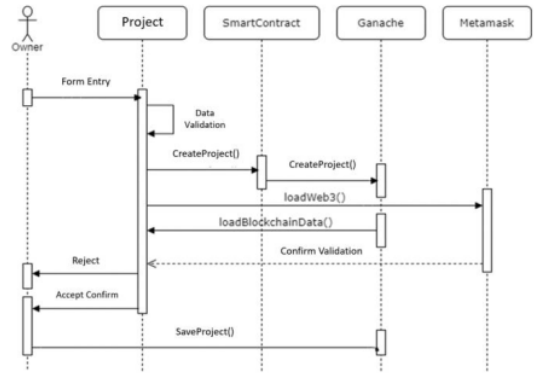

Fig.8. Smart contract interaction in create/post project process

Meanwhile, the interaction of smart contracts in the project bidding process carried out by freelancers is related to the project search module, bidding module and executed on the smart contract (figure 9).

After making a project bid, the next smart contract interaction is in the process of making a project contract which involves the bidding module, contract module, metamask and smart contract (figure 10).

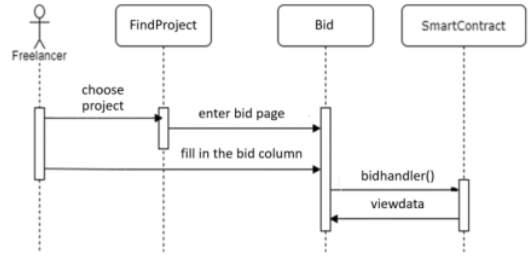

Fig.9. Smart contract interaction in bidding project process

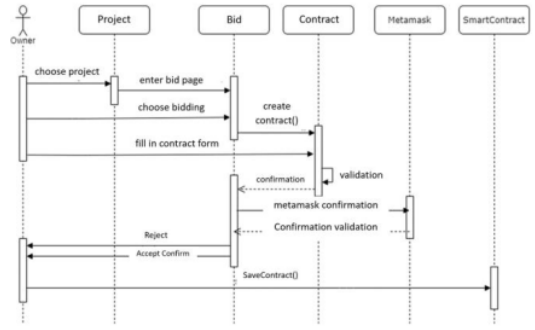

Fig.10. Smart contract interaction in create project contract process

Payment automation is carried out by smart contract interaction with the contract module, ganache and metamask, namely by depositing a number of tokens (ether) via metamask as a project fee, and sending it to freelancers automatically when the work has been completed (figure 11).

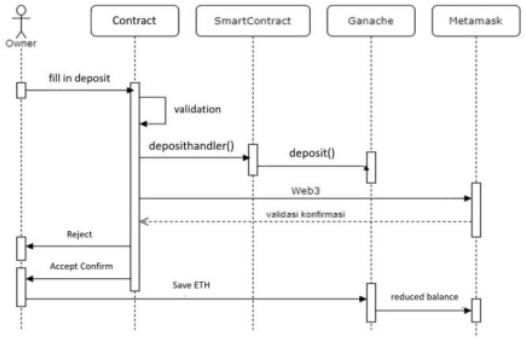

Fig.11. Smart contract interaction in payment project process

C. Use Case diagram

Define use case diagram is a depiction or representation of the relationship between users and the system. It consists of system user actors, namely project owners and freelancers as well as external actors such as Metamask, IPFS and Ethereum, where each of these actors can interact with use cases in the freelancing marketplace system, as shown in figure 12.

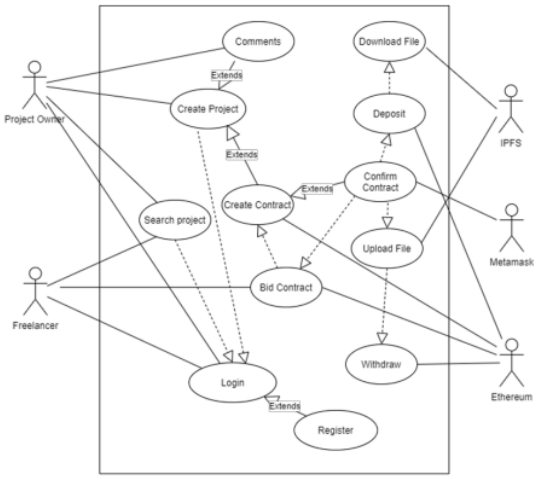

Fig.12. Use case diagram of freelancing marketplace system

D. Interface Implementation

The following are some of the interfaces on the freelance marketplace system, where these interfaces are used by users to interact with the system and show the information contained in it.. Interface implementation includes landing page (figure 13), register page (figure14), search project (figure 15), create project page (figure 16), project contract page (figure 17), bid project (figure 18), contract approval (figure 19), progress of woks (figure 20), and payment page (figure 21).

Fig.13. Landing page freelance marketplace system

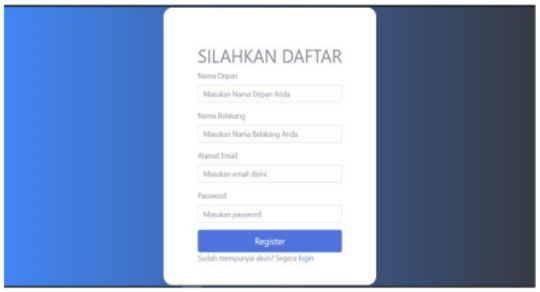

Fig.14. Register page freelance marketplace system

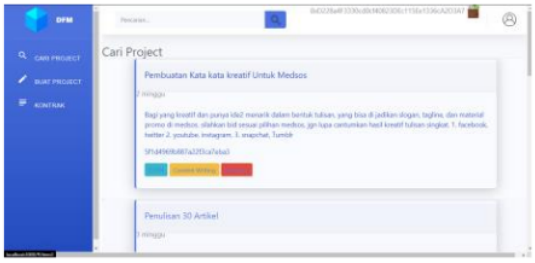

Fig.15. Search project page freelance marketplace system

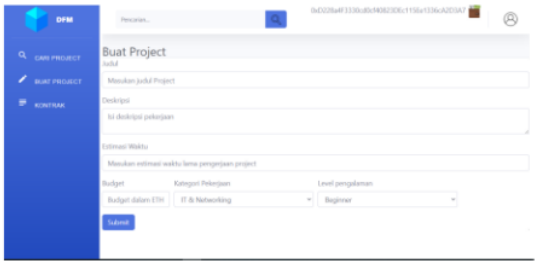

Fig.16. Create project page freelance marketplace system

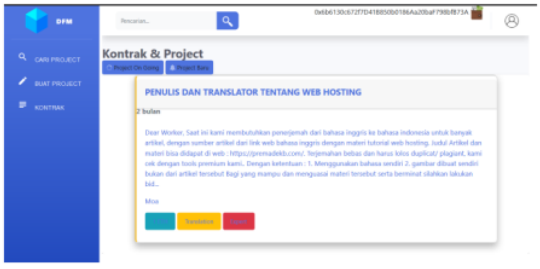

Fig.17. Project contract page freelance marketplace system

Fig.18. Bid project page freelance marketplace system

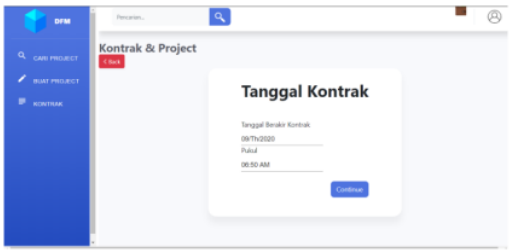

Fig.19. Contract approval page freelance marketplace system

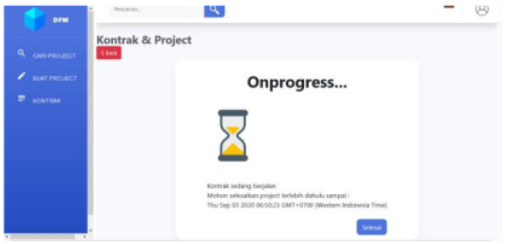

Fig.20. Progress of works page freelance marketplace system

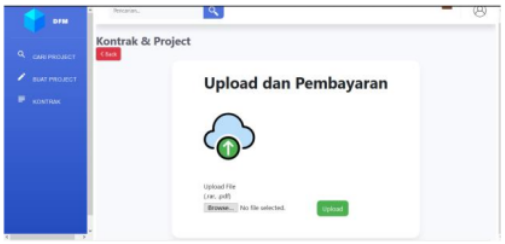

Fig.21. Payment page freelance marketplace system

E. System Testing

At the Alpha testing stage, the test that will be carried out on the system being built is blackbox testing. At this stage, testing will refer to the functionality design and system analysis from the previous chapter. Tests carried out at the following stage include functionality testing. The results of alpha testing with the blackbox method display the results of functional application testing that have been carried out in accordance with the testing scenario. There are two different points of functionality in the results of functionality testing, namely the results of testing the functionality of the web application system (table 1) and the results of testing the functionality of the smart contract (table 2).

TABLE I. Blackbox testing functionality of the web application system

No.	Test Components	Testing Points	Test result
1.	Login	Input data is blank	Valid
		Incorrect input data	Valid
		Input data is correct	Valid
2.	Register	Input data is blank	Valid
		Same input data	Valid
		Input data is correct	Valid
3.	Project Search page	Displays a collection of projects that have been posted on the Blockchain	Valid
4.	Create Project page	Input data is blank	Valid
		Incorrect input data	Valid
		Input data is correct	Valid
5.	Bid page	Add comments	Valid
		Added offers from Blockchain	Valid
		Show comments	Valid
		Displays offers from Blockchain	Valid
6.	Contracts page	Future time input data	Valid
		Input data costs the amount of the cryptocurrency value Ether	Valid
		Integration on contract confirmation	Valid
		Input data files of work results	Valid
		Output data download the work file	Valid
7.	Contracts & Projects page	Displays a list of projects that have been created	Valid
		Displays offers that have been made	Valid

TABLE II. Blackbox testing functionality of smart contract

No.	Test Components	Testing Points	Testing
1.	Save the Listing Project data	Project form input data	Valid
2.	Save bidding data	Bid input data	Valid
3.	Save worker candidate selection data	Address worker input data	Valid
4.	Save the Project end date data	Future time input data	Valid
5.	Save the deposit value in Ether	Ether value input data	Valid
6.	Integrate with workers	Integrate the contract	Valid
7.	Canceling the Contract Agreement	Returns the deposit amount	Valid
8.	Upload	Input data files of work results	Valid
9.	Withdraw Funds	Make a transfer to the address	Valid
10.	Contract Confirmation Completed	Confirmation input data complete	Valid

Based on the existing test results on the smart contract functionality testing, every data stored can be seen on Etherscan using the sub domain network *rinkeby.etherscan.io*. The following is the test result data stored on the Rinkeby test network can be seen in figure 22 - 28.

Fig.22. Listing project data in blockchain network

Fig.23. Save bidding data in blockchain network

Fig.24. Save freelancer selection data in blockchain network

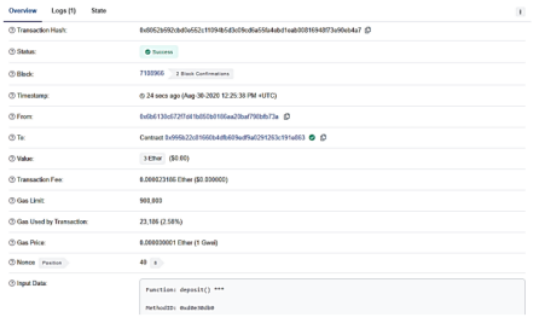

Fig.25. Save deposit data in blockchain network

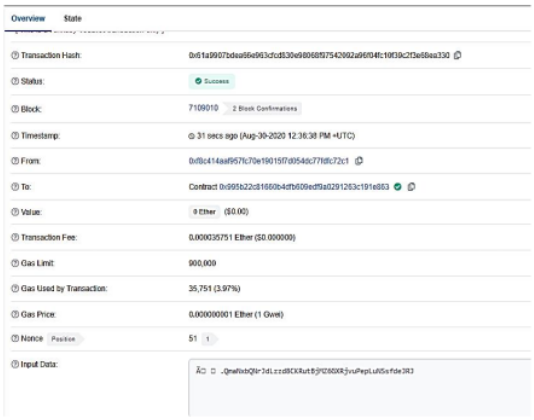

Fig.26. Progress work data in blockchain network

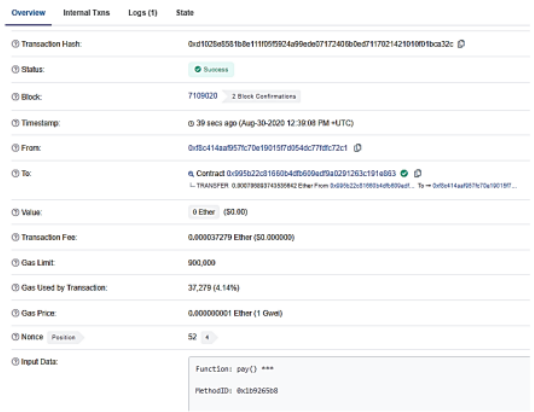

Fig.27. Withdraw payment data in blockchain network

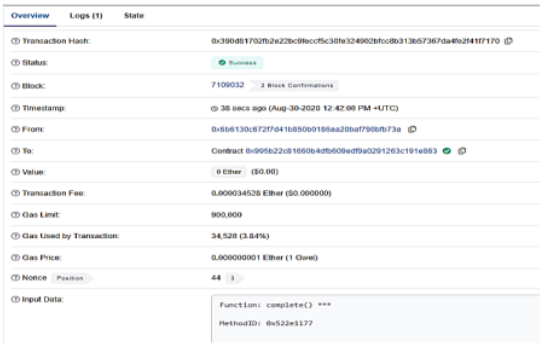

Fig.28 Complete contract confirmation in blockchain network

After testing the functionality it can be observed that the website application system has two test points that were rejected, namely displaying comments in real time and displaying a list of offers that have been made by the user. Meanwhile, there are no rejected test points in the smart contract functionality.. Based on the functionality tests that have been carried out on the application system and smart contract using the blackbox method, it can be concluded that the functionality of the device being built is in accordance with the expected output results. However, the prototype developed is still using the usual web hosting service, where security issues are a concern. The use of distributed cloud services can be used as part of the development solution [22]. In addition, the project search process that is still traditional can be directed into a mechanism that runs automatically according to the data preferences [23][24] of the freelancer so that it will facilitate the link and match between the freelancer and the project owner.

IV. Conclusion

From the results of the tests that have been carried out, there are several conclusions that can be drawn from the prototype development of this freelance marketplace system, namely that the transaction data for the project creation, the project bidding mechanism, the process of making project contracts, to the project completion process have been able to run in a public blockchain environment. The use of smart contracts in these transactions provides an advantage that transactions are more trusted, because smart contracts are on the blockchain which means they are automatically monitored by all devices connected to them. Smart contracts provide autonomy and mechanisms without intermediaries, as smart contracts are managed by the network and executed automatically. Smart contracts also provide speed and certainty of transactions due to their automation mechanism in transactions and payment processing. In addition, smart contracts provide cheaper prices for transactions and are more accurate because they are run by machines. So that the marketplace system using blockchain and smart contracts can be used as an alternative system that is more transparent, ensures the privacy, and is better able to maintain fairness in its user environment. For future research development, this prototype will be developed towards a production model that can run in a real blockchain environment, with the necessary improvements.

References

[1] A. Van den Born and A. Van Witteloostuijn, “Drivers of freelance career success,” J. Organ. Behav., vol. 34, no. 1, pp. 24–46, 2013.

[2] A. S. Gillis, “What is the gig economy? Definition from WhatIs.com,” WhatIs.com, 2020. https://whatis.techtarget.com/definition/gig-economy (accessed May 01, 2021).

[3] H. Yoganarasimhan, “The value of reputation in an online freelance marketplace,” Mark. Sci., vol. 32, no. 6, pp. 860–891, 2013.

[4] M. Gandhi, P. Shah, D. Solanki, and M. Shah, “Decentralized Freelancing System-Trust and Transparency.” International Research Journal of Engineering and Technology (IRJET), 6(09). 2019.

[5] F. Knirsch, A. Unterweger, and D. Engel, “Implementing a blockchain from scratch: why, how, and what we learned,” EURASIP J. Inf. Secur., vol. 2019, no. 1, pp. 1–14, 2019.

[6] I. Afrianto and Y. Heryanto, “Design and Implementation of Work Training Certificate Verification Based On Public Blockchain Platform,” in 2020 Fifth International Conference on Informatics and Computing (ICIC), 2020, pp. 1–8.

[7] I. Afrianto, T. Djatna, Y. Arkeman, I. Hermadi, and I. S. Sitanggang, “Block Chain Technology Architecture For Supply Chain Traceability Of Fisheries Products In Indonesia: Future Challenge,” J. Eng. Sci. Technol. Spec. Issue INCITEST2020, pp. 41–49, 2020.

[8] Steve Cheng et al., “Using blockchain to improve data management in the public sector,” Digit. McKinsey, no. February 2017, pp. 1–10, 2017.

[9] Y. Zhu, C. Lv, Z. Zeng, J. Wang, and B. Pei, “Blockchain-based decentralized storage scheme,” in Journal of Physics: Conference Series, 2019, vol. 1237, no. 4, p. 42008.

[10] H. Khandelwal, K. Mittal, S. Agrawal, and H. Jain, “Certificate verification system using blockchain,” in Lecture Notes in Electrical Engineering, vol. 643, Springer, 2020, pp. 251–257.

[11] I. Afrianto, T. Djatna, Y. Arkeman, I. S. Sitanggang, and I. Hermadi, “Disrupting Agro-industry Supply Chain in Indonesia With Blockchain Technology: Current and Future Challenges,” in 2020

8th International Conference on Cyber and IT Service Management (CITSM), 2020, pp. 1–6.
[12] T. Laurence, Blockchain for dummies. John Wiley & Sons, 2019.
[13] R. O'Shields, "Smart contracts: Legal agreements for the blockchain," NC Bank. Inst., vol. 21, p. 177, 2017.
[14] D. Macrinici, C. Cartofeanu, and S. Gao, "Smart contract applications within blockchain technology: A systematic mapping study," Telemat. Informatics, vol. 35, no. 8, pp. 2337–2354, 2018, doi: https://doi.org/10.1016/j.tele.2018.10.004.
[15] M. Valenta and P. Sandner, "Comparison of ethereum, hyperledger fabric and corda," ebook] Frankfurt Sch. Blockchain Cent., 2017.
[16] E. Androulaki et al., "Hyperledger fabric: a distributed operating system for permissioned blockchains," in Proceedings of the thirteenth EuroSys conference, 2018, pp. 1–15.
[17] M. Radosavljevic, A. Pesic, N. Petrovic, and M. Tosic, "Freelancing blockchain: A practical case-study of trust-driven applications development."
[18] B. Pallam and M. M. Gore, "Boomerang: Blockchain-based Freelance Paradigm on Hyperledger," in 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019, pp. 1–6.
[19] P. Deshmukh, S. Kalwaghe, A. Appa, and A. Pawar, "Decentralised freelancing using Ethereum blockchain," in 2020 International Conference on Communication and Signal Processing (ICCSP), 2020, pp. 881–883.
[20] P. M. Ogedebe and B. P. Jacob, "Software Prototyping: A Strategy to Use When User Lacks Data Processing Experience," ARPN J. Syst. Softw., vol. 2, no. 6, pp. 219–224, 2012.
[21] A. Heryandi and I. Afrianto, "Online Diploma Supplement Information System Modelling for Indonesian Higher Education Institution," in IOP Conference Series: Materials Science and Engineering, 2019, vol. 662, no. 2, doi: 10.1088/1757-899X/662/2/022092.
[22] J. Li, Z. Liu, L. Chen, P. Chen, and J. Wu, "Blockchain-based security architecture for distributed cloud storage," in 2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC), 2017, pp. 408–411.
[23] R. Amin, T. Djatna, Annisa, and I. S. Sitanggang, "Recommendation system based on skyline query: Current and future research," 2020 Int. Conf. Comput. Sci. Its Appl. Agric. ICOSICA 2020, 2020, doi: 10.1109/ICOSICA49951.2020.9243225.
[24] G. Lax, A. Russo, and L. S. Fascì, "A Blockchain-based approach for matching desired and real privacy settings of social network users," Inf. Sci. (Ny)., vol. 557, pp. 220–235, 2021, doi: 10.1016/j.ins.2021.01.004.

Prototype Blockchain Based Smart Contract For Freelance Marketplace System

ORIGINALITY REPORT

14%	14%	5%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.researchgate.net Internet Source	8%
2	ejournal.raharja.ac.id Internet Source	2%
3	ijece.iaescore.com Internet Source	2%
4	Irawan Afrianto, Ikhsan Darmawan, Sufa Atin. "Zoophobia Self-Therapy Application Using Graduated Exposure Therapy Method And Augmented Reality", 2023 International Conference on Informatics Engineering, Science & Technology (INCITEST), 2023 Publication	2%

Exclude quotes	On	Exclude matches	< 2%
Exclude bibliography	On		