



An Implementation of Project Investment Management on Blockchain-based Crowdfunding System

Yan Watequlis Syaifudin^{1*}, Salies Apriliyanto², Hein Htet³, Ferdy Hanan Pradana⁴, Yuri Ariyanto⁵, Indrazno Siradjuddin⁶, Pramana Yoga Saputra⁷, Triana Fatmawati⁸

^{1,2,4,5,7,8} Department of Information Technology, State Polytechnic of Malang, Indonesia

³ Institutes of Innovation for Future Society Designated, Nagoya University, Japan

⁶ Department of Electrical Engineering, State Polytechnic of Malang, Indonesia

 yanwatequilissyaifudin1@gmail.com

Abstract

This study explores the transformative impact of blockchain technology on corporate investment management by developing an application that enhances transparency, security, and efficiency while mitigating fraud risks. By utilizing the Proof of Authority (PoA) methodology on a private blockchain network, the application enables secure, immutable transaction records and facilitates the management of investment projects along with token purchase transactions. It is structured using the Waterfall model, including essential phases such as requirements analysis and rigorous testing, which demonstrate the application's ability to improve operational efficiency and foster accountability among stakeholders. The research highlights the role of tokenization in crowdfunding, wherein rights or assets are converted into digital tokens, enhancing transparency and liquidity. The system also integrates smart contracts to automate critical investment processes while ensuring secure user authentication and authorization. Evaluations reveal high success rates in functionality and user acceptance, though challenges such as scalability and regulatory compliance remain. Addressing these aspects is vital for maintaining the system's effectiveness and user confidence in decentralized investment management.

Keywords: Blockchain, Investment Management, Smart Contracts, Private Blockchain, Equity Crowdfunding

ARTICLE INFO

Article history:

Received

April 02, 2024

Revised

May 20, 2024

Accepted

June 29, 2024

Published by

ISSN

Website

This is an open access article under the CC BY SA license

CV. Creative Tugu Pena

2963-6752

<https://attractivejournal.com/index.php/ajse>

<https://creativecommons.org/licenses/by-sa/4.0/>

<https://creativecommons.org/licenses/by-sa/4.0/>



INTRODUCTION

In the digital age, blockchain technology has revolutionized various sectors, particularly corporate investment management, which often faces challenges related to transparency, security, and efficiency, leading to a lack of trust among stakeholders and increased fraud risks[1]. This study aims to develop an investment management application that leverages blockchain technology to enhance transaction transparency and security by utilizing the Proof of Authority

(PoA) methodology on a private blockchain network, ensuring secure and immutable transaction records. The application features management of investment projects, token purchase transactions, and decentralized storage of transaction history, developed using the waterfall model that includes stages such as requirements analysis, system design, implementation, testing, and maintenance. Rigorous testing demonstrates the application's capability to improve efficiency and foster trust in investment management by recording every transaction transparently and irreversibly, significantly reducing fraud risk and enhancing accountability among stakeholders.

Crowdfunding Investment Management encompasses the professional management of various assets to achieve financial goals[2], involving activities like asset selection, diversification, and portfolio monitoring for clients such as individuals, institutions, and pension funds. Blockchain technology plays a crucial role in providing a secure and transparent framework for data storage through a decentralized network[3], allowing for enhanced accountability and integrity via consensus mechanisms like Proof of Authority[4]. Additionally, Investment Management Systems (IMS) are comprehensive software platforms that integrate portfolio management, risk analytics, and order management functions[5], streamlining workflows and enabling asset managers to make informed decisions, access real-time market data, and improve overall investment performance while reducing operational inefficiencies[6].

In the context of crowdfunding, "tokenization" involves converting rights or assets into digital tokens that can be exchanged[7], facilitating capital raising through Initial Coin Offerings (ICOs) by providing tokens with specific functions and values[8]. Tokenization utilizes blockchain technology to ensure the legitimacy of tokens and secure transactions, enhancing transparency, liquidity, and global reach within fundraising strategies[9]. The investment process through a crowdfunding platform involves several key stages, including project creation by owners, administrative review, investor participation, fundraising, project execution, closure, and profit distribution. Additionally, tokens serve as digital representations of assets or rights within the project, with types including utility tokens for future products and security tokens for ownership rights. Smart contracts play a critical role in managing tokens by automating the execution of rules related to their creation and ownership transfer[10], ensuring transparency and security while addressing design elements such as total supply and divisibility to meet project needs and contributor expectations[11].

This paper presents the development of decentralized system for token-based project investment management that is designed with robust functional and technical requirements to ensure secure and efficient operations. Key components include secure user authentication and authorization for investors, project managers, and administrators, facilitated by role-based access mechanisms. The

Proof of Authority system integrates smart contracts to automate critical processes such as investment handling and asset ownership recording, while leveraging a Proof of Authority (PoA) consensus protocol to validate transactions efficiently[12], [13]. Technical specifications encompass adequate hardware and software support, involving platforms like Go Ethereum[14] and Solidity[15] for smart contract development. The system's workflow is carefully structured to guide user interactions from input to data management, ensuring seamless communication between the client, the blockchain network, and databases for transaction integrity[16]. Furthermore, the smart contracts manage both transaction and token data, supporting transparent and automated processes within the investment landscape. Overall, the implementation emphasizes security, reliability, and user engagement through a comprehensive approach to managing investments in a decentralized environment.

The evaluation focuses on its functionality, smart contract performance, transaction validation, and user acceptance, revealing a strong overall performance across various testing metrics. Functionality tests indicate successful outcomes in user login, registration, project management, and transaction handling, while smart contract evaluations for both TransactionContract and TokenContract confirm effective management of transactions and tokens. User Acceptance Testing (UAT) results demonstrate high satisfaction levels among application members and admins, especially regarding ease of use and clarity of investment information, although some respondents highlighted the need for improvements in the speed of the investment process. Future challenges for the system include the necessity for scalable infrastructure to accommodate growth, the importance of regulatory compliance in an evolving digital landscape, and the need for ongoing user education to ensure confidence in navigating tokenized investments. Addressing these challenges will be essential for the system's long-term effectiveness and user trust.

CROWDFUNDING INVESTMENT MANAGEMENT

A. Investment Management

Investment Management is the professional process of managing different types of assets and financial instruments with the aim of achieving predetermined financial goals. The process involves various activities, such as financial planning, appropriate asset selection, diversification, and continuous monitoring of the investment portfolio. Clients using this service can be individuals with a certain wealth, institutions, companies, or pension funds.

B. Blockchain

Blockchain is an innovative technology that enables distributed and secure data storage without the need for intermediaries[17]. In its basic structure, a

blockchain consists of blocks that contain transaction data, a hash of the previous block, and a timestamp. Each block is connected chronologically, creating a chain that guarantees data integrity[16]. With this design, blockchain provides an effective solution for creating a transparent and trustworthy system. One of the key features of blockchain is security and transparency. Data stored on a blockchain is encrypted and recorded across many nodes on the network, making it extremely difficult to hack. Also, since all transactions can be viewed by authorized parties, it increases accountability. As such, blockchain can be used in a variety of applications that require trust and transparency[18].

C. Investment Management System

An Investment Management System (IMS) is a comprehensive software platform designed to streamline and enhance the management of investment portfolios and workflows. It integrates various essential functionalities for investment professionals, including Portfolio Management (PMS), Risk Analytics, Order Management (OMS), and Execution Management (EMS). By consolidating these capabilities into a single system, an IMS facilitates improved operational efficiency, reduced costs, and better decision making for asset managers[19]. Key features include real-time monitoring of investment performance, tools for assessing risks through methodologies such as Value-at-Risk (VaR) and a robust Investment Book of Record (IBOR) that maintains a comprehensive view of all investment positions and transactions.

Figure 1. Crowdfunding and Tokenization



The implementation of an investment management system provides several advantages, such as streamlined workflows that reduce operational inefficiencies and the risk of data discrepancies. Users benefit from immediate access to up-to-date market data and analytics, enabling informed decision-making and proactive risk management strategies[20]. Several organizations use this system, including asset managers in charge of numerous accounts, institutional investors such as pension funds in charge of sizable portfolios, and wealth management companies offering customized strategies for individual clients. In general, an IMS serves as a vital tool for investment professionals, enabling them to navigate complex portfolios while enhancing efficiency and improving overall investment performance. Figure ure 1 illustrates this.

SYSTEM CONCEPT AND OVERVIEW

In this context, "tokenization" refers to the act of transforming rights or assets into a digital form that can be exchanged, such as tokens[21]. Tokenization is used in the context of Initial Coin Offerings (ICOs) to raise capital by selling digital tokens with predetermined functionality and value[22]. These tokens can be used as exchangeable assets, grant access to goods and services, or perform other duties. Tokenization makes use of blockchain technology to guarantee the legitimacy of tokens and secure transactions. In general, the integration of tokenization into crowdfunding frameworks represents a significant advance, offering greater transparency, liquidity, and global reach. This evolution in fundraising strategies aligns with the growing demand for innovative and efficient financial mechanisms.

A. Investment Process

The investors must go through in the investment process through a crowdfunding platform. Each step in the diagram reflects a systematic flow from registration to project funding. The following is an explanation of each of these steps:

1) Project Owner Creates Project:

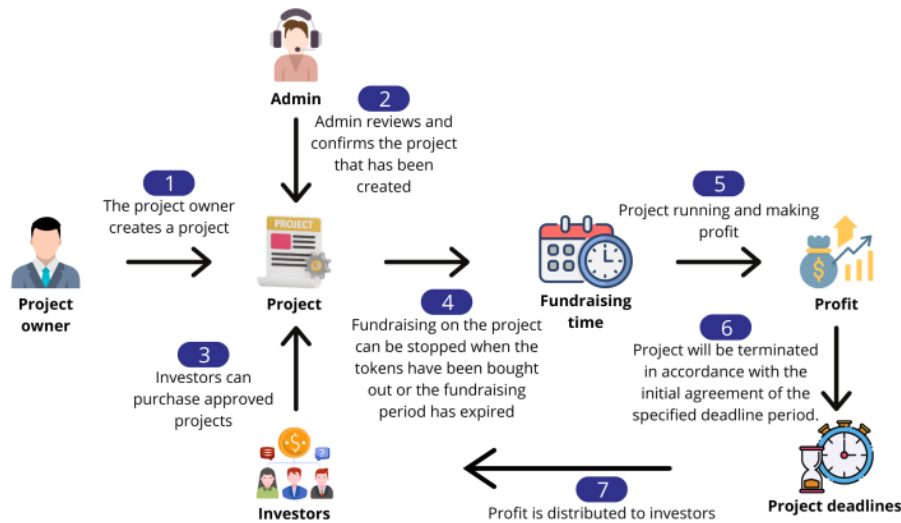
- The process starts with a project owner (can be an individual or an organization) who has an idea or ideas that they want to bring to life.
- The project owner then creates a complete project proposal, including a description of the project, goals, funding targets, and planned use of funds.

2) Admin Reviews and Confirms Project:

- Once the project is created, the admin (usually from the crowdfunding platform) will review the project proposal.
- This review aims to ensure that the project meets the terms and conditions that apply on the platform.

- If it passes the review, the project will be confirmed and can begin to be offered to investors.
- 3) *Investors Purchase the Project:*
- Once the project is confirmed, it will be displayed on the crowdfunding platform and made accessible to investors.
 - Investors interested in the project can purchase “tokens” or “shares” of the project as a form of investment.
 - The amount invested by each investor will vary depending on the size of the project and the interest of the investor.
- 4) *Fundraising:*
- The fundraising process will take place over a certain predetermined period of time.
 - During this period, the project owner will continue to promote his project to attract more investors to participate.
 - Fundraising can be stopped if the fund target has been reached or if the fundraising period has ended.
- 5) *Project Runs and Generates Profits:*
- Once the funding target is reached, the project owner will start running the project.
 - If the project runs successfully, it will generate profits.
- 6) *Project Closes:*
- *The project will be closed according to the time period specified in the initial agreement.*
 - *Project closure can occur for several reasons, for example, the project has been completed or the project fails to achieve the expected targets.*
- 7) *Profits Distributed to Investors:*
- After the project is closed, the profits generated from the project will be distributed to the investors in accordance with their respective investment portions.
 - This profit sharing is usually done periodically, for example every year or every quarter.

Figure 2. Investment Stages in Crowdfunding



B. Tokenization Model

Tokens in blockchain-based crowdfunding act as digital representations of assets, contributions, or specific rights within a project. This section explains the different types of tokens that can be created, such as utility tokens, which provide access to future services or products, and security tokens, which represent ownership or a share of the project's profits. The role of tokens as tradable digital assets, controlled automatically by smart contracts, is also discussed. Smart contracts are self-executing programs running on the blockchain that automatically implement, manage, and verify agreements or transactions. In the context of token creation and management, smart contracts ensure that all predefined rules, such as the total number of tokens that can be created, token distribution, and ownership transfers, are executed transparently and securely.

The design and specifications of tokens are crucial to ensure that they meet the needs of the project and the expectations of contributors. This section describes aspects of token design, including the total supply, divisibility, and special features such as burnability or mintability. These design elements determine how the tokens will interact with the blockchain system and contributors.

DECENTRALIZED SYSTEM FOR TOKEN-BASED PROJECT INVESTMENT MANAGEMENT

A. System Requirements

The design of the system hinges on a robust set of functional requirements essential for efficient and secure operations. A key priority is the implementation of secure user authentication and authorization, ensuring that investors, project managers, and administrators access the system through role-based authorization mechanisms[23]. This foundational security measure ensures that users can only perform actions pertinent to their roles. Furthermore, the integration of smart

contracts is critical, as it automates pivotal processes such as investment handling, dividend distribution, and asset ownership recording. Through this automation, the system not only increases accuracy but also reduces the need for manual intervention. Additionally, the incorporation of a Proof of Authority (PoA) consensus protocol is vital for transaction processing, enabling transactions to be recorded and validated by trusted validators, thus ensuring greater speed and efficiency compared to traditional public consensus systems. The system also empowers project managers with tools for managing fundraising campaigns by facilitating the creation, monitoring, and transparent reporting of these campaigns, thereby enhancing investor trust. Lastly, the adoption of decentralized validation ensures that trusted validators within the PoA framework verify and approve all transactions, obviating the need for open miners and reinforcing system reliability.

Data integrity is a cornerstone of the system's design, ensuring that all transaction data, once approved by validators via the PoA mechanism[24], is permanently recorded and immutable. This immutability is critical in establishing an environment of trust and reliability, assuring users that transaction records are accurate and tamper-proof[25]. By addressing these non-functional requirements alongside the functional ones, the system is geared to deliver a highly effective and reliable platform that meets the needs of its diverse users while maintaining operational excellence.

B. Technical Specification

The decentralized Investment Management System requires the following technical specifications:

- **Hardware:** The Investment Management System implementation also requires adequate hardware support. The hardware used must be able to optimally support system performance, including transaction processing, data storage, and maintaining network stability. In this subchapter, we will discuss the hardware specifications needed to run the system, ranging from servers, storage, to the network infrastructure needed for the system to function efficiently and safely.
- **Software:** In addition to reliable hardware, the development of an Investment Management System requires reliable software to support the functionality required by users, both investors and project managers, as shown in Table I. In this system, software plays an important role to ensure transaction security, efficient portfolio management, and transparency in the investment flow. In this subchapter, we will explain the software used in the implementation of the system, including programming languages, frameworks, and tools that support optimal investment management.

Table 1. Software Specifications

Software Component	Details
--------------------	---------

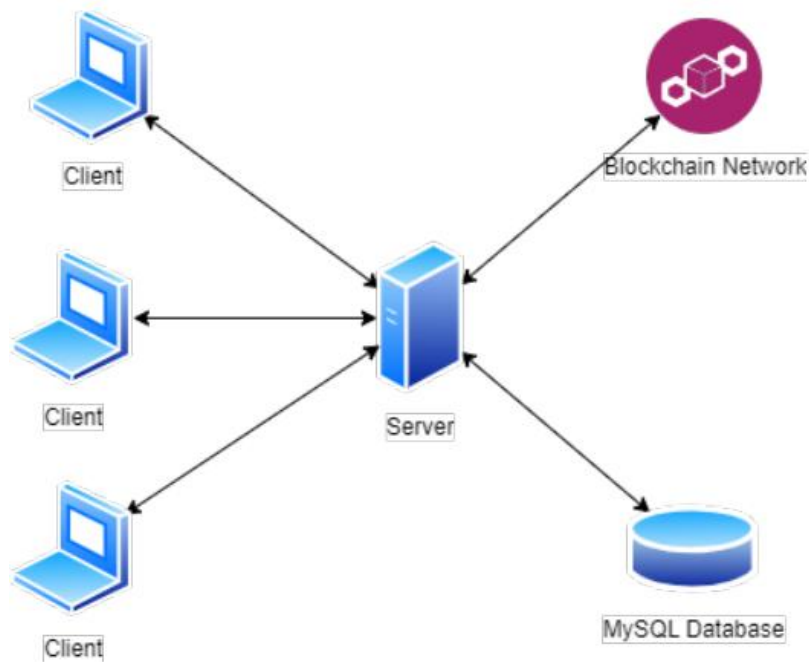
Blockchain Platform	Go Ethereum (Geth)
Smart Contract Language	Solidity
Development SDK	Thirdweb SDK
Operating System	Linux Ubuntu 22.04.03 LTS

C. System Workflow

The system workflow applied in the development of the blockchain-based investment management application outlines the sequence of steps and processes involved in the operation of the platform, from user interaction to transaction execution and data handling. This section describes the data flow within the system from user input, investment validation, token issuance, to data storage and performance monitoring. In addition, it also explains how these workflows are structured to ensure optimal efficiency, strong security, and data integrity. The purpose of this discussion is to provide a comprehensive understanding of how the various components in the system collaborate seamlessly to deliver the desired functionality and performance.

The client sends requests to the server, which then processes and manages the data before sending a response back to the client. The server acts as the main processing hub, bridging communication between the client, the blockchain network, and the MySQL database. The blockchain network is used to store token purchase transaction data and token ownership data through transaction validation performed by smart contracts, ensuring that each transaction is permanently recorded and immutable. On the other hand, the MySQL database is used to store information not required by the application but not stored on the blockchain network.

Figure 3. Investment Management System Architecture



D. Design of Smart Contracts

Smart contracts written using Solidity and implemented on the Ethereum network are designed to automate and execute commands without the need for a third party. Here are the scripts of the `TransactionContract` and `TokenContract` smart contracts:

Listing 1: TransactionContract

```

// SPDX-License-Identifier: MIT
pragma solidity 0.8.16;
contract TransactionContract
{
    struct Transaction
    {
        string transactionCode;
        string campaignId;
        string fromToUserId;
        string orderType;
        string paymentStatus;
        string status;
        uint quantity;
        uint256 totalPrice;
        string paymentMethodDetailId;
        string paymentProof; // Added
        paymentProof attribute
        uint createdAt;
    } mapping(string => Transaction) public transactions;
    string[] public transactionCodes

    function addTransaction;
    function getAllTransactions();
    function updateStatus;
    function updatePaymentStatus;
  
```

```
function updatePaymentProof;  
function getTransactionByFromToUserId;  
function getTransactionByCode;  
function getTransactionByCampaignId;  
function getCountTransaction();  
}
```

1) *TransactionContract Overview:*

- **Purpose:** Manages transactions including campaign details, order types, and payment statuses.
- **Key Functions:**
 - `addTransaction`: This function is used to add new transactions to the system. This is the starting point in transaction management, where every transaction made will be inputted into the smart contract.
 - `updateStatus`: This function serves to update the status of a particular transaction. This allows flexibility in managing transaction statuses that may change over time, such as from “processed” to “completed”.
 - `updatePaymentStatus`: This function allows users to update the payment status of a particular transaction, so that the system can reflect the latest condition of the transaction, whether it has been paid or not.
 - `updatePaymentProof`: This function is used to update the proof of payment associated with a particular transaction. This is important to validate that the payment has been made correctly and provide transparency to all parties involved.
 - `getAllTransactions`: This function is designed to collect and return all transactions that have been added into the system. It provides a way for other users or contracts to view all existing transactions.
 - `getTransactionByFromToUserId`: This function is designed to collect all transactions made by a particular user, based on that user ID. This allows users to review their transaction history easily.
 - `getTransactionByCode`: This function returns complete information regarding a particular transaction based on the given transaction code. It allows quick and direct access to transaction details.
 - `getTransactionByCampaignId`: This function is used to get all transactions related to a particular campaign, based on the campaign ID. This helps in analyzing the performance of a particular campaign by collecting all related transactions.
 - `getCountTransaction`: This function calculates the total number of transactions that have been added to the contract. This is useful for getting concise information on how many transactions have been made in the system.

2) *TokenContract Overview:*

- **Purpose:** Manages tokens associated with campaigns and transactions, including their status and ownership.
- **Key Functions:**
 - `addToken`: Adds a new token to the contract with detailed information such as campaign ID and transaction code. This function allows the creation of new tokens that can be tracked by the contract.
 - `getAllToken`: Retrieves all tokens in the contract, allowing anyone to see the entire list of tokens stored in the contract. This function helps to get complete data about all tokens.
 - `getTokensByCampaignId`: Retrieve all tokens associated with a specific campaign ID. This is useful when we want to see the tokens registered in one particular campaign, for example to track token inventory in that campaign.
 - `getTokensBySoldTo`: Retrieve all tokens that have been sold to a particular user. With this function, we can see all the tokens owned by a particular user, helping to record or manage token ownership.
 - `getTokensByCampaignIdAndSoldTo`: Retrieves all tokens in a specific campaign that were sold to a specific user. This function helps to view the tokens that a user has purchased in a specific campaign, such as verifying purchases in a custom campaign.
 - `getSoldTokensByCampaignId`: Retrieve all tokens that have the status “sold” for a specific campaign ID and for a specific user. This function is useful for getting data on tokens that have been successfully sold in a campaign and for knowing specific purchases by users.
 - `getTokensByTransactionCode`: Retrieve tokens based on specific transaction codes. This helps track or verify specific transactions if we have the transaction code, such as for auditing or status checking.

IMPLEMENTATION OF WEB APPLICATION

A. Private Blockchain Network Configuration

The configuration file for a private blockchain network is crucial for defining the network parameters and initial settings. This file outlines the blockchain’s rules, including the consensus mechanism, chain ID, and various block settings. It specifies the difficulty level, gas limits, and initial account allocations, ensuring that the blockchain operates as intended from its inception. Below is an example configuration that includes key settings for the blockchain network.

Listing 2: Private Blockchain Network Configuration

```

{
  "conFigure ": {
    "chainId": 1234567,
    "homesteadBlock": 0,
    "eip150Block": 0,
    "eip155Block": 0,
    "eip158Block": 0,
    "byzantiumBlock": 0,
    "constantinopleBlock": 0,
    "petersburgBlock": 0,
    "istanbulBlock": 0,
    "berlinBlock": 0,
    "clique": {
      "period": 5,
      "epoch": 30000
    }
  },
  "difficulty": "1",
  "gasLimit": "8000000",
  "extraDataSignerAddress": "0x0000SignerAddress00000",
  "alloc": {
    "node1Address": {
      "balance": "{{balance}}"
    }
  }
}

```

The conFigure uration file details the following aspects:

- **conFigure** : Includes blockchain parameters such as `chainId` and activation blocks for different Ethereum Improvement Proposals (EIPs). The `clique` section sets consensus-related parameters like `period` and `epoch`.
- **difficulty**: Defines the mining difficulty. For private networks, this is often set to a low value to ensure faster block creation.
- **gasLimit**: Sets the maximum gas limit per block, which affects transaction complexity and block size.
- **extraDataSignerAddress**: Contains additional data related to the network's signer address.
- **alloc**: Specifies initial balances for accounts. In this example, `node1Address` is assigned a balance indicated by `balance`.

B. Blockchain Integration

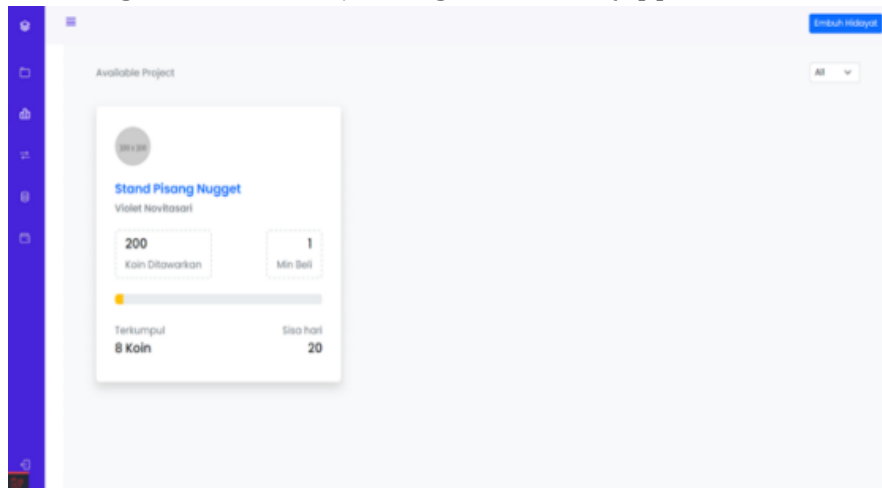
The implementation of storage on the blockchain network begins when a user initiates a token purchase transaction through the application's user interface. The user inputs the details of the desired token purchase. Once the transaction data is entered, the system performs validation to ensure the transaction's legitimacy. This validation includes verifying the number of tokens being purchased and ensuring that the user has sufficient balance to complete the purchase.

After validating the transaction data, the system creates tokens based on the quantity purchased by the user. This token creation process is conducted according to the specifications defined within the system. Subsequently, the system sends a request to store the transaction data and token information on the blockchain. This storage request is validated by the blockchain using the Proof of Authority (PoA) consensus algorithm. Once the transaction and token creation processes are completed, the user receives a confirmation that their transaction has been successfully processed and that the tokens have been added to their account. This integration ensures that all transactions are securely recorded on the blockchain, providing an immutable and transparent record of token ownership and transaction history.

C. Web Implementation

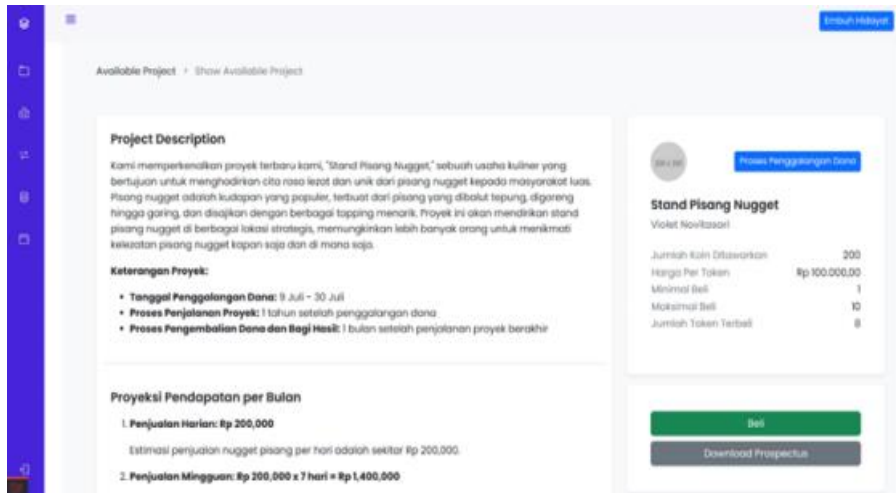
The main interface of the project page available to application members is shown in Figure 4. This page displays a list of all published projects with tokens available for purchase.

Figure 4. Main Project Page Available (Application Member)



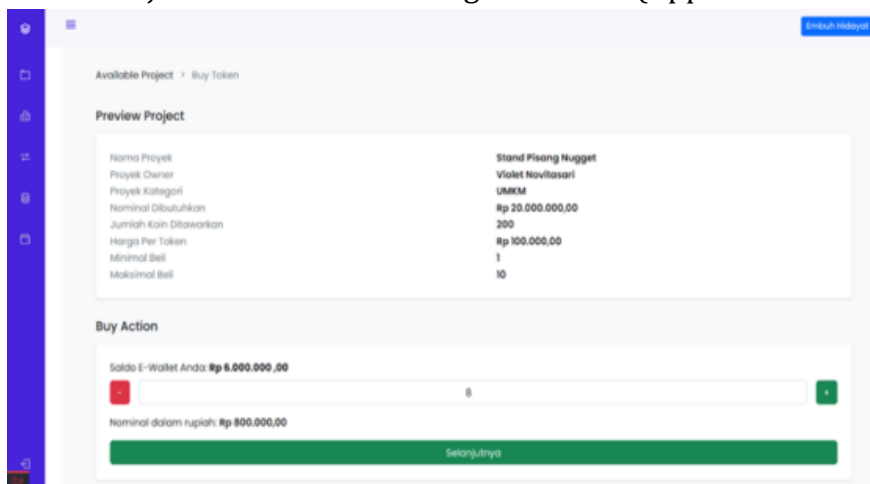
To view the details of each project, Figure 5 illustrates the project detail page.

Figure 5. Project Detail Page Available (Application Member)



Token purchase for a project can be completed through the interface shown in Figure ure 6, which presents the project token purchase page.

Figure . 6: Project Token Purchase Page Available (Application Member)



EVALUATION

This section evaluates the blockchain-based investment management application, focusing on functionality, smart contract performance, transaction validation, and user acceptance. The evaluation results are summarized, followed by a discussion and recommendations for future improvements.

A. Evaluation Results

The results are based on the testing described in Section 5.6, including functionality testing, smart contract evaluation, blockchain transaction validation, and user acceptance. Here is a summary of the testing outcomes.

1) *Functionality Testing Results:* The functionality testing results are summarized in Table 2.

Table 2. Functionality Testing Results

Role	Feature	Result
All Roles	Login	Successful
All Roles	Register	Successful
Admin	User Management	Successful
Admin	Payment Method Management	Successful
Admin	Project Management	Successful
Admin	Token Purchase Transactions Management	Successful
Admin	Application Wallet Transactions Management	Successful
Application Member	Project Management	Successful
Application Member	Available Project	Successful
Application Member	Token Purchase Transaction History	Successful
Application Member	My Tokens	Successful
Application Member	Application Wallet Transactions	Successful

2) *Smart Contract Testing Results:* The smart contract testing results are divided into two parts, TransactionContract and TokenContract. The details are summarized in Tables 3 and 4. Table 5 presents the results for the *TransactionContract* testing.

Table 3. TransactionContract Testing Results

Feature	Result
Add Transaction	Successful
Update Transaction Status	Successful
Update Payment Status	Successful
Update Payment Proof	Successful
Get Transactions by fromToUserId	Successful
Get Transactions by campaignId	Successful
Get Total Transactions	Successful
Project Management	Successful

Table 4. TokenContract Testing Results

Feature	Result
Add New Token	Successful
Get Token by campaignId	Successful
Get Token by campaignId and soldTo	Successful
Get Sold Tokens by campaignId	Successful
Get Token by soldTo	Successful
Get Token by transactionCode	Successful
Update Status by transactionCode	Successful
Delete Token by transactionCode	Successful
Delete Token by campaignId and soldTo	Successful
Delete Token by campaignId	Successful

3) *Blockchain Transaction Validation Testing Results:* The blockchain transaction validation testing results are summarized in Table 7.

Table 5. Blockchain Transaction Validation Testing Results

Feature	Result
Validate Token Purchase Transaction Storage	Successful
Validate Token Ownership Data Storage	Successful

a) *User Acceptance Testing (UAT) Results:* The User Acceptance Testing (UAT) results for both Application Members and Admins are shown in Table 6 that summarizes the UAT results for Application Members.

Table 6. User Acceptance Testing (UAT) Results (10 respondents)

Feature	Percentage	Agreement Level
Ease of Use for Investment and Navigation	80.00%	Strongly Agree
Clarity of Investment Project Information	82.00%	Strongly Agree
Smooth Investment Process	92.00%	Strongly Agree
Speed and Efficiency of Investment Process	52.00%	Strongly Agree
Profit-Sharing Process in Accordance with Agreement	88.00%	Strongly Agree
Smart Contract Functionality	96.00%	Strongly Agree
Blockchain Integration Security	86.00%	Strongly Agree

b) *Admin:* Table 9 summarizes the UAT results for Admins.

Table 7. User Acceptance Testing (UAT) Results for Admin (10 respondents)

Feature	Percentage	Agreement Level
Admin Interface Navigation	70.00%	Agree
Project Approval Management Mechanism	84.00%	Strongly Agree
Efficiency of Token Issuance and Management	88.00%	Strongly Agree
Access to Transaction Monitoring and User Activity	78.00%	Agree
Accuracy of Transaction Verification and Validation	86.00%	Strongly Agree

DISCUSSIONS

The proposal for a blockchain-based crowdfunding investment management system outlines a comprehensive framework designed to enhance transparency, efficiency, and security in the investment process. The findings indicate that the integration of blockchain technology facilitates the tokenization of assets, which allows for streamlined transaction processes and enhances the overall user experience. The Investment Management System (IMS) effectively automates key functions such as portfolio management, risk assessment, and transaction handling through smart contracts. The testing results showcase its functionalities, confirming high success rates in user authentication, project management, and transaction processing, highlighting the platform's potential to effectively manage investments, reduce operational inefficiencies, and support investor engagement.

The evaluations of the system highlight the successful implementation of smart contracts in managing transactions and tokens, demonstrating secure and transparent interactions among users. Smart contracts ensure that the terms of investment agreements are honored while providing automated updates on transaction statuses and ownership transfers. User Acceptance Testing (UAT) results show high levels of satisfaction among both application members and admins, particularly concerning ease of use and the clarity of investment information, with most respondents expressing strong agreement with the system's speed and efficiency. However, areas such as the speed of the investment process indicated a need for further enhancements to meet user expectations fully, emphasizing the importance of constant iterative improvements based on user feedback.

Despite the promising results, future challenges remain that could impact the system's effectiveness and adoption. The first challenge lies in scaling the infrastructure to handle increased transaction volumes as the user base grows, which necessitates robust hardware and software support. Additionally, regulatory compliance will be critical, especially given the evolving landscape of digital assets and investment platforms. Ensuring that the system remains adaptable to changing regulations will be essential for its long-term viability. Finally, user education and support are vital to bolster confidence in using the platform and navigating the complexities of tokenized investments. Addressing these challenges will be crucial for sustaining growth and enhancing user trust in the crowdfunding investment management system.

CONCLUSION

The proposal for a blockchain-based crowdfunding investment management system provides a framework aimed at improving transparency, efficiency, and security in investment processes through the tokenization of assets. The findings reveal that the system effectively automates critical functions such as

portfolio management, risk assessment, and transaction handling using smart contracts, which have demonstrated high success rates in areas like user authentication and project management. Evaluations of the system indicate secure and transparent interactions facilitated by smart contracts, with user acceptance testing reflecting strong satisfaction regarding ease of use and clarity of information. However, challenges such as infrastructure scalability, regulatory compliance, and the need for user education persist and must be addressed to ensure the system's long-term viability and foster user trust in navigating tokenized investments.

REFERENCES

- [1] R. Ramadoss, "Blockchain technology: An overview," *IEEE Potentials*, vol. 41, no. 6, 2022, doi: 10.1109/MPOT.2022.3208395.
- [2] D. Cumming, M. Meoli, and S. Vismara, "Does equity crowdfunding democratize entrepreneurial finance?," *Small Business Economics*, vol. 56, no. 2, 2021, doi: 10.1007/s11187-019-00188-z.
- [3] A. S. Yadav, S. Agrawal, and D. S. Kushwaha, "Distributed Ledger Technology-based land transaction system with trusted nodes consensus mechanism," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 8, 2022, doi: 10.1016/j.jksuci.2021.02.002.
- [4] L. Chacko, P. Rajan, V. Anilkumar, and V. Pathari, "Proof-of-Variable-Authority: A Blockchain Consensus Mechanism for Securing IoT Networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2023. doi: 10.1007/978-3-031-49099-6_24.
- [5] D. Gozman and W. Currie, "The role of investment management systems in regulatory compliance: A post-financial crisis study of displacement mechanisms," *Journal of Information Technology*, vol. 29, no. 1, 2014, doi: 10.1057/jit.2013.16.
- [6] A. Rühr, D. Streich, B. Berger, and T. Hess, "A classification of decision automation and delegation in digital investment management systems," in *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2019. doi: 10.24251/hicss.2019.174.
- [7] R. M. Garcia-Teruel and H. Simón-Moreno, "The digital tokenization of property rights. A comparative perspective," *Computer Law and Security Review*, vol. 41, 2021, doi: 10.1016/j.clsr.2021.105543.
- [8] L. Hornuf, T. Kück, and A. Schwienbacher, "Initial coin offerings, information disclosure, and fraud," *Small Business Economics*, vol. 58, no. 4, 2022, doi: 10.1007/s11187-021-00471-y.
- [9] A. Ferreira, "Emerging regulatory approaches to blockchain based token economy," *The Journal of The British Blockchain Association*, vol. 3, no. 1, 2020, doi: 10.31585/jbba-3-1-(6)2020.
- [10] Z. Zheng *et al.*, "An overview on smart contracts: Challenges, advances and platforms," *Future Generation Computer Systems*, vol. 105, 2020, doi: 10.1016/j.future.2019.12.019.
- [11] P. Karhula, V. Vallivaara, N. Lehto, V. Pentikäinen, and J. Aikio, "Asset Tracking With Smart Contracts," *IEEE Wireless Communications and Networking Conference*, no. May, 2019.
- [12] J. Yang, J. Dai, H. B. Gooi, H. D. Nguyen, and A. Paudel, "A Proof-of-Authority Blockchain-Based Distributed Control System for Islanded Microgrids," *IEEE Trans Industr Inform*, vol. 18, no. 11, 2022, doi: 10.1109/TII.2022.3142755.

- [13] M. M. Islam and H. P. In, "Decentralized Global Copyright System Based on Consortium Blockchain With Proof of Authority," *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3270627.
- [14] Getch.Ethereum, "Home | go-ethereum," Getch official website.
- [15] Solidity, "Solidity — Solidity 0.8.19 documentation," Documentation Web Solidity.
- [16] K. M. Khan, A. Fahim, D. M. Saleem, and M. Jokhio, "Empirical Investigation on Blockchain Interoperability," *VFAST Transactions on Software Engineering*, vol. 11, no. 1, 2023, doi: 10.21015/vtse.v10i4.1315.
- [17] X. Xu, I. Weber, and M. Staples, "Existing Blockchain Platforms," in *Architecture for Blockchain Applications*, 2019. doi: 10.1007/978-3-030-03035-3_2.
- [18] P. Centobelli, R. Cerchione, P. Del Vecchio, E. Oropallo, and G. Secundo, "Blockchain technology for bridging trust, traceability and transparency in circular supply chain," *Information and Management*, vol. 59, no. 7, 2022, doi: 10.1016/j.im.2021.103508.
- [19] J. Q. Li and S. McNeil, "Data envelopment analysis for highway asset investment assessment," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 8, no. 1, 2021, doi: 10.1016/j.jtte.2019.06.001.
- [20] S. Sood, "Leveraging Web Analytics for Optimizing Digital Marketing Strategies," in *Big Data Analytics: Digital Marketing and Decision-Making*, 2022. doi: 10.1201/9781003307761-7.
- [21] J. Gan, G. Tsoukalas, and S. Netessine, "Initial coin offerings, speculation, and asset tokenization," *Manage Sci*, vol. 67, no. 2, 2021, doi: 10.1287/mnsc.2020.3796.
- [22] Y. Tian, R. E. Minchin, K. Chung, J. Woo, and P. Adriaens, "Towards Inclusive and Sustainable Infrastructure Development through Blockchain-enabled Asset Tokenization: An Exploratory Case Study," *IOP Conf Ser Mater Sci Eng*, vol. 1218, no. 1, 2022, doi: 10.1088/1757-899x/1218/1/012040.
- [23] S. Peng *et al.*, "A peer-to-peer file storage and sharing system based on consortium blockchain," *Future Generation Computer Systems*, vol. 141, 2023, doi: 10.1016/j.future.2022.11.010.
- [24] H. Han, S. Fei, Z. Yan, and X. Zhou, "A survey on blockchain-based integrity auditing for cloud data," *Digital Communications and Networks*, vol. 8, no. 5, 2022, doi: 10.1016/j.dcan.2022.04.036.
- [25] M. Altulyan, L. Yao, S. Kanhere, and C. Huang, "A blockchain framework data integrity enhanced recommender system," *Comput Intell*, vol. 39, no. 1, 2023, doi: 10.1111/coin.12548.

Copyright Holder :

© Yan Watequlis Syaifudin, Salies Apriliyanto, Hein Htet, Ferdy Hanan Pradana, Yuri Ariyanto, Indrazno Siradjuddin, Pramana Yoga Saputra, Triana Fatmawati. (2024).

First Publication Right :

© Asian Journal Science and Engineering

This article is under:

CC BY SA