Enhancing Transparency and Accountability:
Blockchain-Based Government Fund Allocation and Tracking System

Abstract

This report presents the design and implementation of a blockchain-based system for government fund allocation and tracking, aimed at enhancing transparency, accountability, and efficiency in public finance management. Utilizing Quorum, a permissioned blockchain platform, the system introduces a structured mechanism for fund requests and review, along with robust smart contract functionalities including authority control, multi-signature approvals, and dynamic redistribution of funds. The smart contracts serve multiple roles, enabling citizens to submit fund requests, government officials to review and approve these requests, and auditors to ensure compliance and detect anomalies. The system's architecture supports real-time tracking of fund distribution and automates many aspects of the allocation process, thereby reducing administrative overhead, minimizing the risk of corruption, and improving response times to fiscal changes. By leveraging blockchain technology, the proposed system addresses the challenges of traditional government funding systems, offering a secure, transparent, and user-centric platform that promises to revolutionize public sector finance management.

The Chinese University of Hong Kong

FTEC5520 Applied Blockchain and Cryptocurrency

Group 6

CEN, Baihui 1155197612
CHEN, Angyu 1155197620
CHEN, Yunfan 1155198241
HUANG, Shiqi 1155204275
JIANG,Tianyun 1155197611
LIANG, Haojin 1155200065
NIU, Xinyan 1155201239

Table of Content

Project Background

Within the public finance framework, governments serve as the managers of the nation's resources, overseeing and implementing fiscal programs through its various departments. Government corruption poses significant barriers to successful infrastructure project delivery and government opacity is a key factor contributing to long-term project failures. (Abodei, Norta, Azogu, Udokwu & Draheim, 2019) The International Monetary Fund (IMF) and the Organization for Economic Cooperation and Development (OECD) have emphasized the importance of fiscal transparency and accountability in public finances for attaining better macroeconomic and fiscal stability, improving credit ratings, and strengthening fiscal discipline. To increase transparency and accountability, nations globally have adopted tailored regulatory measures. For example, the United States enacted the Digital Accountability and Transparency Act (DATA) in 2014, under which federal departments are required to disclose financials quarterly on government websites. However, solely depending on public disclosure to prevent fiscal mismanagement still leaves room for opacity and inefficiency in the allocation and tracking of government funds. Evident in the U.S., the Payment Protection Program (PPP), which initially aimed at supporting small businesses during economic downturns, has reportedly disbursed loans to unrelated entities. In addition, financial transparency and accountability in the face of uncontrollable events remain a great challenge. During COVID-19, while most governments disclosed overall expenditure information during the pandemic, detailed information on beneficiaries, impacts, and performance was often lacking (Xiao, H.Y& Wang, X.H, 2023).

The vast majority of governments still use the traditional government fund tracking system, and most disclose public expenditure data on government websites, including information on central government budgets, expenditures, fund allocations, and debt management. Traditional systems often have multiple layers of bureaucracy and manual processes that can lead to inefficiencies in fund allocation and reporting, as well as human intervention and complex record-keeping processes that can increase the risk of error and fraud. There are also difficulties in sharing data effectively across different departments and agencies, leading to information silos. The traditional system of government fund tracking is no longer able to fulfill the demands of the government as well as the public.

These incidents underscore a pressing need for a paradigm shift in government funding, one that transcends traditional disclosure mechanisms and introduces a more robust, transparent, and traceable system. Blockchain technology presents a promising avenue for realizing this transformation and the blockchain paradigm is capable of well resolving the problems that existed in the traditional model. By leveraging the inherent features of blockchain—immutability and transparency—this project aims to revolutionize government fund allocation and tracking, thereby enhancing the integrity and efficiency of public finance systems.

Goals and Framework

Our project aims to develop a blockchain-based government fund allocation and tracking system that ensures transparency and accountability. Main objectives and blockchain applications include:

1) Develop a system for real-time tracking of government fund allocation: Every transaction on the blockchain is recorded on a ledger distributed across every participant in the network. This feature can create a transparent and immutable record of all government transactions, including contracts, disbursements, and receipts. 2) Streamline government fund allocation process: Smart contracts can be programmed to release funds only when certain conditions are met. This automation can reduce the likelihood of overspending or misallocation of funds. 3) Enhance data security and integrity in government financial transactions: The decentralized nature and cryptographic protection can offer robust security. Each block is linked to its predecessor through cryptographic hashes, ensuring the integrity of the entire chain.

Our proposed system introduces an advanced architecture to enhance the transparency, efficiency, and accountability of government funding by various entities. Here's how the user-oriented and secure platform operates: Upon the initial login, users are required to register in the system, subsequently requesting funds by uploading the necessary certification documents. These requests are then encrypted and stored on the blockchain as individual blocks, with each request assigned a unique ID. This allows users to easily review and track the status of their requests using this ID. Fund allocators, including central and state governments, process and approve or reject requests by accessing the Requests List. Based on specifics like amount and project nature, approval types vary for different requests, including direct state approval, direct central approval, and dual government approval. Post-approval, smart contracts automatically allocate funds to requesters. Auditors are integral to maintaining financial propriety and accountability. The system grants them secure and transparent access to the entire fund allocation history, thereby streamlining their monitoring duties and aiding in the detection of any irregularities or corruption. Through this comprehensive and user-centric approach, our system aims to revolutionize the management and tracking of government funds.

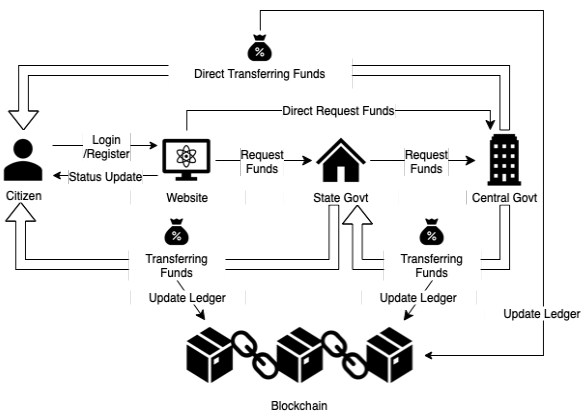


Fig. 1 Flowchart of funding process

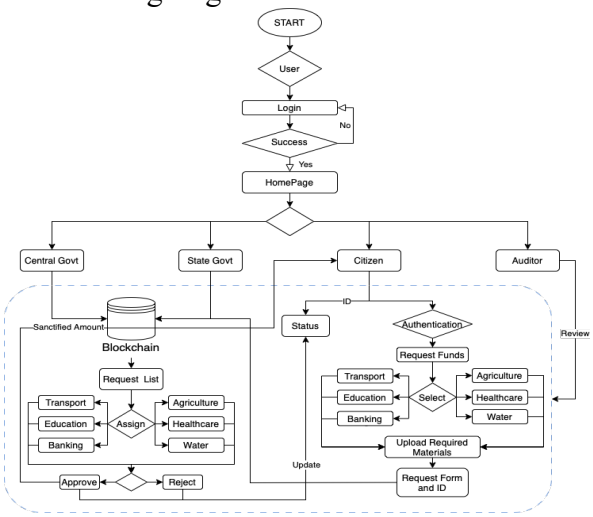


Fig. 2 Architecture diagram

Blockchain Network

We choose Quorum as the network platform for our blockchain-based government fund allocation and tracking system for several strategic advantages. Firstly, Quorum supports private transactions and private contracts. This is crucial for a government setting where information about fund allocation might need to be visible to authorized parties only, ensuring compliance with data protection laws and maintaining confidentiality where necessary. Unlike public blockchains where all transaction details are visible to everyone, Quorum allows for selective transparency. Secondly, Quorum is highly compatible with the Ethereum application. It supports all Ethereum tools and DApps, facilitating easier integration and development. This compatibility allows us to leverage the vast array of tools available in the Ethereum ecosystem, reducing development time and complexity. Lastly, Quorum eliminates the need for gas to perform transactions, which can be a significant cost saving when deploying a large-scale application like a government fund tracking system. This makes it economically viable for extensive governmental use where numerous transactions occur.

Smart Contract Logic

The smart contract logic within the government fund allocation system is engineered with a tiered authority control mechanism to define and enforce distinct permissions for each participating entity:

For all entities: A foundational access control layer is built into the smart contract, providing a structured framework that tailors functionality and access levels specific to different roles. This ensures a secure and orderly interaction within the blockchain environment.

For citizens requesting funds: Citizens are granted the capabilities to initiate fund requests through the smart contract interface, alongside the ability to track the progress and current status of these requests. This design promotes an engaging and transparent process for public participation in government fund allocation.

For governmental approvals: The smart contract incorporates a multi-signature approval workflow, requiring a collective authorization from designated government stakeholders before any fund release. This built-in mechanism is complemented by a dynamic fund redistribution feature that allows for the reallocation of resources across various departments or projects, providing flexibility to adapt to the changing fiscal priorities set by the government.

For oversight by governments and auditors: To uphold the integrity of the funding process, the smart contract is equipped with an anomaly detection system. This system is automated to continuously analyze funding patterns, identifying and flagging irregularities, potential discrepancies, or any deviations from established budgetary guidelines. It serves as a critical tool for both government officials and auditors to supervise and maintain accountability throughout the funding lifecycle.

Smart Contract Design

Authority Control Mechanism

The Authority Control Mechanism is crucial for defining and enforcing the roles and permissions within the system. By categorizing users into distinct roles—Citizen, Government, and Auditor—the mechanism efficiently delineates responsibilities and access rights within the system. Citizens can submit funding requests and view their statuses. Governments are allowed to review submitted requests, make decisions to approve or reject and execute fund transfers for approved projects. Auditors are granted read-only access to requests, enabling them to perform oversight of the funding process. The smart contract includes functions that check a user's role before allowing them to execute certain actions, like submitting funding requests or transferring funds. Role assignments are logged and can only be modified by designated administrators, adding an additional layer of security.

```
// Citizens submit requests
function submitRequest(string memory _description, uint _amount) public onlyRole(Role.Citizen) {
requests.push(Request({
description: _description,
amount: _amount,
requester: payable(msg.sender),
status: RequestStatus.Submitted
}));
emit RequestSubmitted(requests.length - 1, msg.sender);
}

// Government reviews and decides on the requests
function reviewRequest(uint _requestId, RequestStatus _status) public onlyRole(Role.Government) {
Request storage request = requests[_requestId];
request.status = _status;
emit RequestReviewed(_requestId, _status);
}

// Government transfers funds for approved requests
function transferFunds(uint _requestId) public payable onlyRole(Role.Government) {
request.requester.transfer(msg.value);
emit FundsTransferred(_requestId, msg.value);
}

// Function for auditors to view request details
// Auditors can directly view requests through getter functions
```

Additionally, governments have exclusive authority to assign roles, a critical feature that safeguards the system's integrity by preventing unauthorized role assignment.

```
// Government officials can assign roles
function assignRole(address _user, Role _role) public onlyRole(Role.Government) {
roles[_user] = _role;
}
```

Mechanism for Fund Requests and Review

The Mechanism for Fund Requests and Review is a pivotal part of our blockchain-based system, designed to handle the submission, review, and approval of government fund requests transparently and efficiently. This mechanism ensures that all transactions related to fund requests are traceable and immutable, thereby enhancing accountability and reducing the likelihood of fraud.

The structure of a fund request involves company name, contact, industry, project information, request amount, requestor's address, and request status (submitted, in progress, approved, rejected).

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

enum Industry { Transport, Education, Banking, Agriculture, Healthcare, Water }
enum RequestStatus { Submitted, InProgress, Approved, Rejected }

struct FundRequest {
        string companyName; // Company name
        string companyContact; // Contact information
        Industry industry; // Industry of the request, selected from drop-down list
        string projectTitle; // Project information
        string projectDescription;
        uint amount; // Requested amount
        address payable requestor; // Address of the citizen submitting the request
        RequestStatus status; // Request status
        bool stateApproved;
        bool centralApproved;
        bool isFulfilled;
}
```

We define a `createFundRequest` function to accept the parameters required for submitting a request. This function allows a user to submit a new fund request with all the necessary details like the nature of the project, amount requested, and justification.

```
function createFundRequest(
        string memory _companyName,
        string memory _companyContact,
        Industry _industry,
        string memory _projectTitle,
        string memory _projectDescription,
        uint _amount
) external {
        fundRequests.push(FundRequest({
        companyName: _companyName,
        companyContact: _companyContact,
        industry: _industry,
        projectTitle: _projectTitle,
        projectDescription: _projectDescription,
        amount: _amount,
        stateApproved: false,
        centralApproved: false,
        isFulfilled: false
        }));
}
```

We also design a function `viewFundRequest` for citizens to view their request status through each unique request id.

```
function viewFundRequest(uint _requestId) external view returns (FundRequest memory) {
        require(_requestId < fundRequests.length, "Request ID is out of bounds");
        return fundRequests[_requestId];
}
```

Multi-signature approval mechanism

To enhance the security of fund disbursements, the Multi-signature Approval Mechanism requires that high-value transactions receive approvals from multiple authorized government officials before they can be processed. This approach mitigates the risk of fraud or errors because it requires a consensus among multiple stakeholders, making it much harder for a single person to misappropriate funds or make unilateral decisions that could impact financial governance.

First, we define the roles, departments, and basic structure for the fund allocation process. There are in total two governments: state and central, and six departments: transport, education, agriculture, healthcare, and water. Fund requests are assigned to specific government departments based on the industry of the request.

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract GovernmentFundAllocation {

        enum Role { StateGovernment, CentralGovernment }
        enum Department { Transport, Education, Banking, Agriculture, Healthcare, Water }
        struct FundRequest {
        uint amount;
        Department department;
        bool stateApproved;
        bool centralApproved;
        bool isFulfilled;
        }

        address public stateGovernment;
        address public centralGovernment;
        mapping(Department => address) public departmentAddresses;
        FundRequest[] public fundRequests;

        constructor(address _stateGovernment, address _centralGovernment) {
        stateGovernment = _stateGovernment;
        centralGovernment = _centralGovernment;
        }
```

Second, fund allocation requires different levels of approval based on the fund request amount. If the amount is greater than $50,000, both approvals from the departments of state government and central government are required; otherwise, only the state government's approval is needed. Once the request is approved, the fund will be transferred from the government to the

designated department, and finally to the requestor's account. The request status will also be updated accordingly.

```
// Mapping to keep track of each department's balance within the contract
mapping(Industry => uint) public departmentBalances;

// Event declarations for transparency and tracking
event FundsTransferredToDepartment(Industry indexed department, uint amount);
event FundsTransferredToRequestor(address indexed requestor, uint amount);
event DepositReceived(address from, Industry indexed department, uint amount);

function fulfillRequest(uint _requestId) internal {
        FundRequest storage request = fundRequests[_requestId];
        require(request.status == RequestStatus.Approved, "Request must be approved before fulfillment.");

// Check if the approval requirement is met
        bool approvalConditionMet = (request.amount <= 50000 * 1 ether && request.stateApproved) ||
(request.amount > 50000 * 1 ether && request.stateApproved && request.centralApproved);

        if (approvalConditionMet && departmentBalances[request.industry] >= request.amount) {
        // Assuming the contract holds funds for each department
        // Transfer funds from the department's balance to the requestor
        departmentBalances[request.industry] -= request.amount;
        (bool sent, ) = request.requestor.call{value: request.amount}("");
        require(sent, "Failed to send Ether");

        request.isFulfilled = true;
        request.status = RequestStatus.Approved;

        emit FundsTransferredToRequestor(request.requestor, request.amount);
        } else {
        request.status = RequestStatus.Rejected;
        }
}
```

Dynamic Redistribution of Funds Mechanism

The dynamic redistribution of funds mechanism provides a flexible framework that allows for the reallocation of government funds across different projects or departments swiftly and efficiently. This function is especially necessary in scenarios where rapid response to emergencies, evolving public health challenges, or changes in socio-economic conditions demand a flexible approach to budget management. The design of this function involves necessary adjustments to ensure that government funds are optimally deployed by each department. The smart contract includes functions that can adjust allocations based on input from government officials. Redistribution events are recorded, providing an auditable trail of how funds are moved within the system.

```
        // Function to redistribute funds dynamically among departments
        function redistributeFunds() public onlyAdmin {
            uint totalAllocations = 0;
            for (uint i = 0; i < departments.length; i++) {
                totalAllocations += departments[i].allocation;
            }
```

```
// Calculate new allocations based on available funds plus current allocations
 uint newAllocationPerDepartment = (totalFundsAvailable + totalAllocations) / departments.length;

// Update each department's allocation
for (uint i = 0; i < departments.length; i++) {
        departments[i].allocation = newAllocationPerDepartment;
        emit AllocationUpdated(departments[i].name, newAllocationPerDepartment);
    }

// After redistribution, all funds are considered allocated
    totalFundsAvailable = 0;
    emit FundsRedistributed();
}
}
```

Identifying Unusual Fundings

In the domain of government and auditing, the supervision of funding processes requires a proactive system to detect anomalies that may indicate irregularities or deviations from the approved budget norms. To address this need, an Anomaly Detection Mechanism can be integrated into a blockchain-based system, capable of automatically identifying unusual funding patterns and potential discrepancies. Two key strategies can be employed to enhance this mechanism, leveraging both on-chain and off-chain solutions.

Simplified on-chain checks

The simplified on-chain checks mechanism relies on the transparency and immutability of blockchain to monitor transactions and flag any anomalies that occur within the network. Governments can implement smart contracts that define clear rules and parameters for fund allocation and utilization. These contracts can automatically flag transactions that exceed predefined thresholds or deviate from normal patterns. For instance, a smart contract could limit the maximum amount of funds that can be allocated to a specific project or require a multi-signature approval process for transactions exceeding a certain size.

```
# This function checks for unusual funding patterns in on-chain transactions
function government_supervision(on_chain_transactions):
    # Load the approved budget norms for comparison
    approved_budgets = load_approved_budgets()
    # Initialize a list to store detected anomalies
    anomalies = []

    # Iterate through each transaction in the on-chain transaction data
    for transaction in on_chain_transactions:
        # Identify the category and amount of the transaction
        budget_category = transaction.category
        transaction_amount = transaction.amount

        # Check if the transaction amount exceeds the approved budget
        if not approved_budgets[budget_category] or transaction_amount > approved_budgets[budget_category]:
            anomalies.append(transaction)  # Add the transaction to the anomalies list
        # Check if the transaction amount is significantly below the approved budget
        elif transaction_amount < approved_budgets[budget_category] * MIN_DEVIATION_THRESHOLD:
            anomalies.append(transaction)  # Add the transaction to the anomalies list
```

```
    # If anomalies are detected, trigger alerts and initiate an investigation
    if anomalies:
        trigger_alerts(anomalies)  # Notify relevant authorities about the detected anomalies
        initiate_investigation(anomalies)  # Start an investigation process

    # Return the list of detected anomalies
    return anomalies
```

Auditors can be granted access to these smart contracts to monitor transactions in real-time. They can also utilize additional smart contract functions specifically designed for audit purposes, such as functions that compile reports of transactions that have been flagged as anomalies.

```
# This function audits on-chain transactions for any discrepancies from normal patterns
function auditor_supervision(on_chain_transactions, normal_patterns):
    # Initialize a list to store transactions that do not match normal patterns
    discrepancies = []

    # Iterate through each transaction in the on-chain transaction data
    for transaction in on_chain_transactions:
        # Check if the transaction does not match any of the normal patterns
        if not transaction_matches_pattern(transaction, normal_patterns):
            discrepancies.append(transaction)  # Add the transaction to the discrepancies list

    # If discrepancies are found, report to management and recommend an audit
    if discrepancies:
        report_to_management(discrepancies)  # Inform management about the discrepancies
        recommend_audit(discrepancies)  # Suggest an audit to further investigate the discrepancies

    # Return the list of transactions that do not match normal patterns
    return discrepancies
```

Oracles for off-chain anomaly detection

This strategy integrates external data into the blockchain to identify inconsistencies that might not be evident from the on-chain data. Governments can use off-chain data analytics tools to analyze funding patterns and detect anomalies. These tools can handle more complex datasets and analyses than on-chain contracts. When these tools identify an anomaly, they can send a report or alert to the blockchain through an oracle, triggering on-chain actions or investigations.

```
# This function detects anomalies in government funding data by comparing expenditures to budget thresholds
function detect_anomalies(government_funding_data):
    # Load the predefined budget thresholds for different project categories
    budget_thresholds = load_budget_thresholds()
    # Initialize an empty list to store detected anomalies
    anomalies = []

    # Iterate through each project in the government funding data
    for project in government_funding_data:
        # Check if the project's expenditure exceeds the approved budget for its category
        if project.expenditure > budget_thresholds[project.category]:
            # If so, add the project to the list of anomalies
            anomalies.append(project)
        # Check if the project's expenditure is significantly below the approved budget, considering a deviation
limit
```

```
        elif project.expenditure < budget_thresholds[project.category] - project.deviation_limit:
            # If it is, also add the project to the list of anomalies
            anomalies.append(project)

    # If any anomalies are detected
    if anomalies:
        # Notify the relevant authorities about the detected anomalies
        notify_relevant_authorities(anomalies)
        # Initiate an investigation into the anomalies
        initiate_investigation(anomalies)

    # Return the list of detected anomalies
    return anomalies
```

Similarly, auditors can utilize specialized off-chain tools to monitor government fund allocations. They can independently verify the findings reported by oracles or conduct their own analyses using off-chain data, ensuring a dual-layer of oversight.

```
# This function audits funding records to identify discrepancies from normal funding patterns
function audit_funding(funding_records):
    # Load the normal funding patterns based on historical data and standard operating procedures
    normal_patterns = load_normal_patterns()
    # Initialize an empty list to store records that show discrepancies
    discrepancies = []

    # Iterate through each record in the funding records
    for record in funding_records:
        # If the record is flagged as an 'unusual_pattern'
        if record.type == 'unusual_pattern':
            # Check the record against the normal patterns
            check_record(record, normal_patterns)
            # If the record is identified as a discrepancy
            if record.is_discrepancy:
                # Add the record to the list of discrepancies
                discrepancies.append(record)

    # If any discrepancies are found
    if discrepancies:
        # Report the discrepancies to the relevant parties
        report_discrepancies(discrepancies)
        # Recommend necessary corrections based on the discrepancies found
        recommend_corrections(discrepancies)

    # Return the list of identified discrepancies
    return discrepancies
```

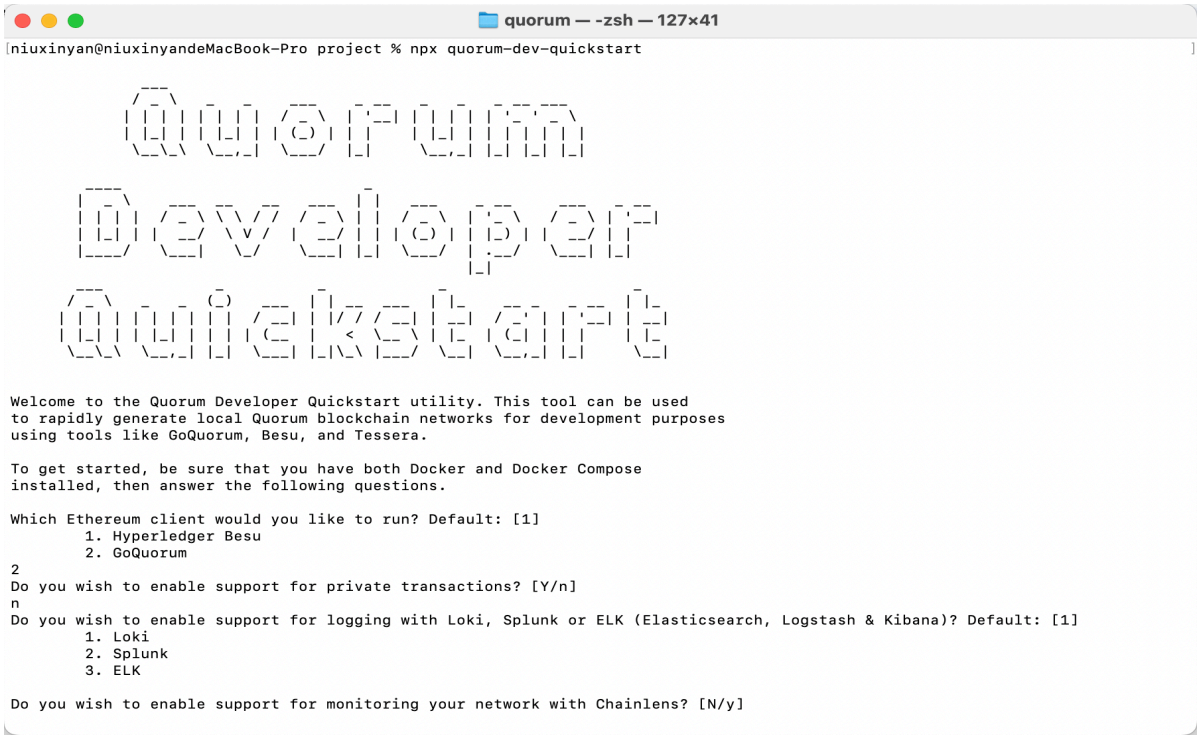
In this system, these two methodologies can be utilized complementarily to provide a more comprehensive security assurance and data monitoring. Under normal circumstances, governments and auditors typically employ Simplified On-Chain Checks to ensure that transactions comply with the rules and protocols of the blockchain, focusing on real-time, on-chain inspections of data within the blockchain itself. Off-Chain Anomaly Detection can be used to identify potential fraudulent activities. Governments and auditors can then invoke data external to the blockchain and base their assessments of anomalies on statistical models or machine learning algorithms.

Implementation

Step 1: Docker provides a consistent environment for our application, ensuring that it runs the same on every machine, which is crucial for testing and deployment. We firstly downloaded docker through the link below：https://www.docker.com/products/docker-desktop/

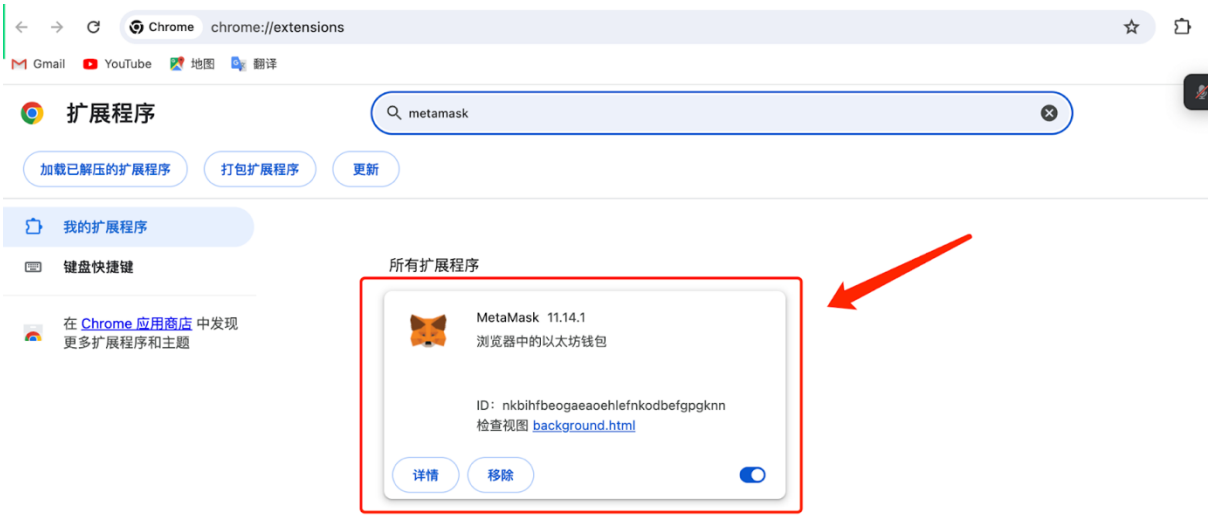
Step 2: Pull the mirror of quorum with command *`docker pull quorumengineering/quorum`*.

Step 3: Use quick start to create docker-compose.yml file.

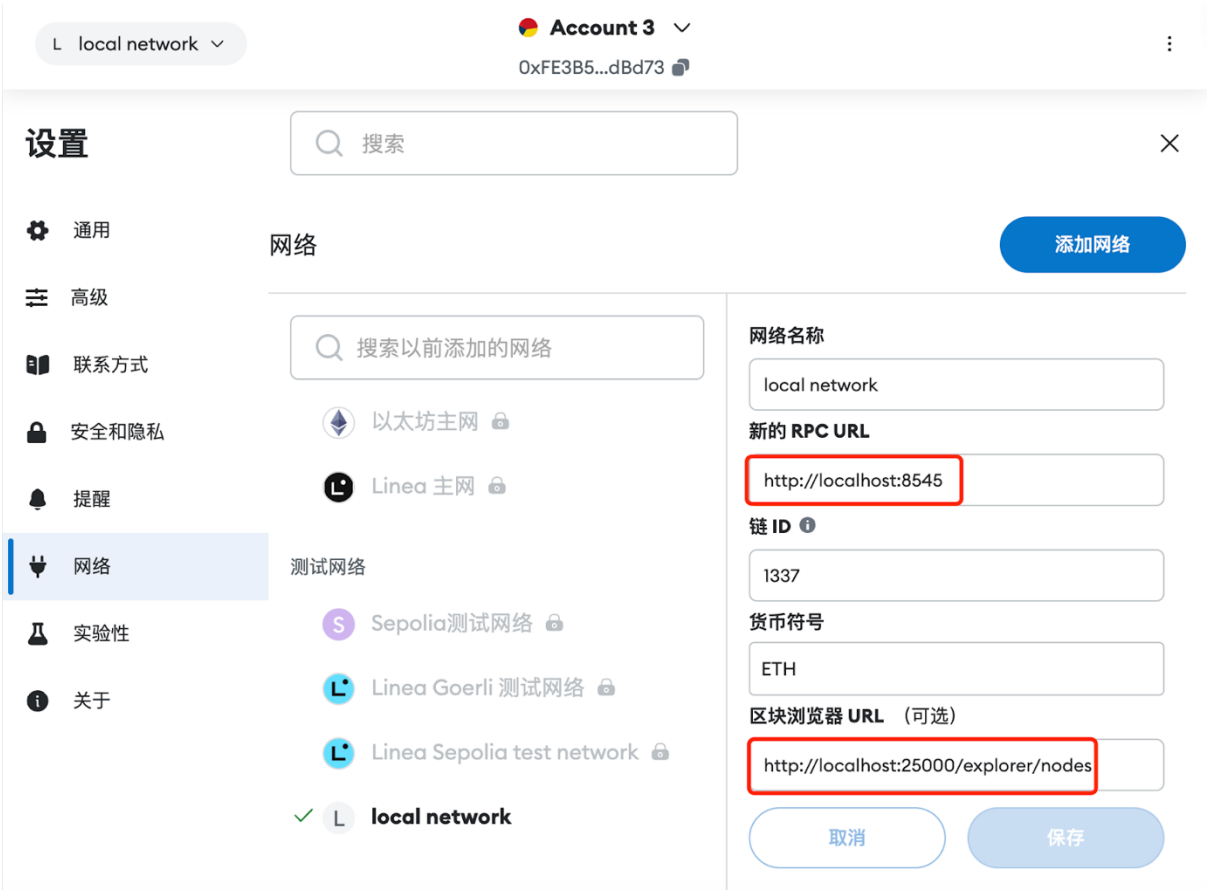


Step 4: Use *`docker-compose.yml`* as the configuration file to build a blockchain environment. This file defines the services, volumes, and networks for the containers.

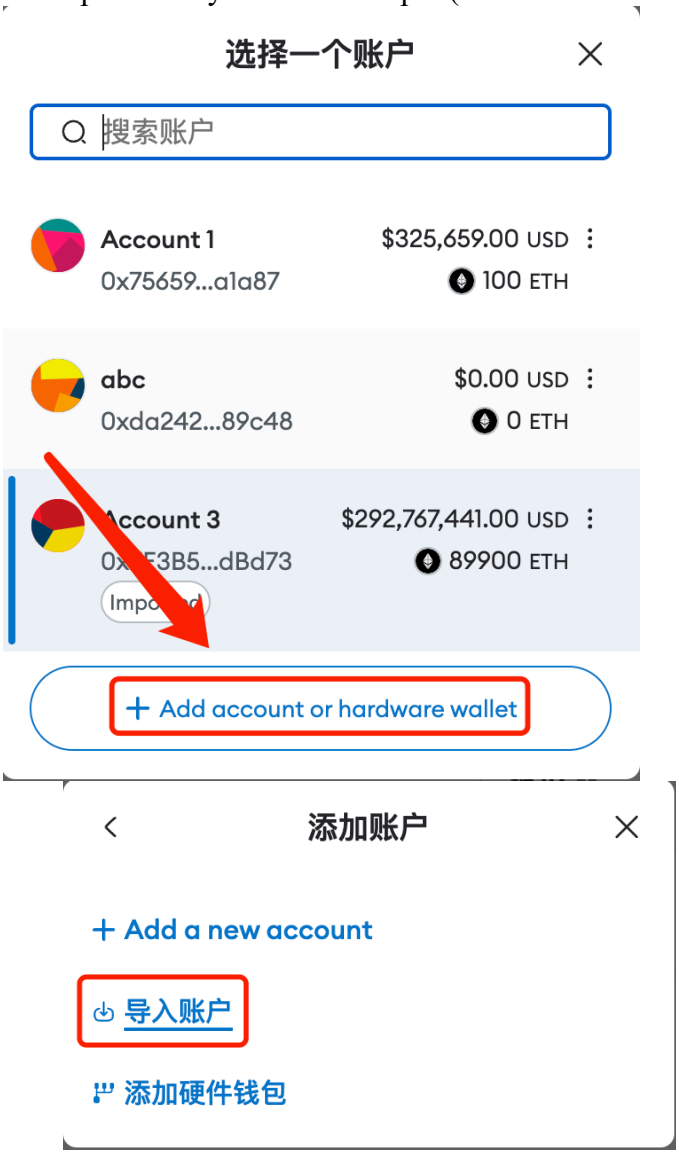
Step 5: Download MetaMask plugin to interact with the deployed smart contracts.



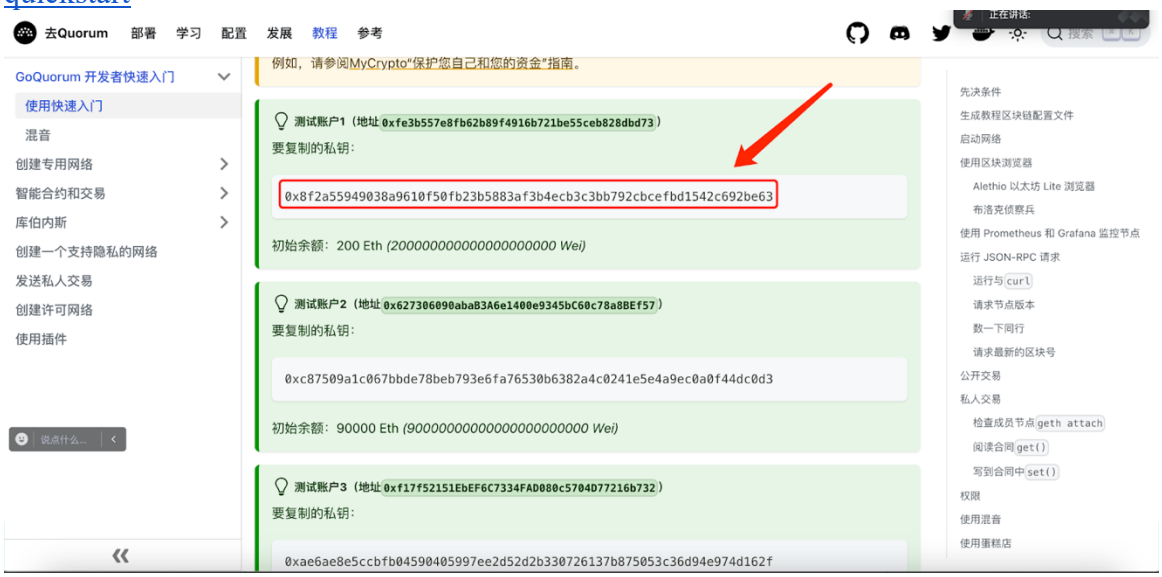
Step 6: Add local network to MetaMask.



Step 7: Use the address and private key created in step 4 (add the account in metamask).

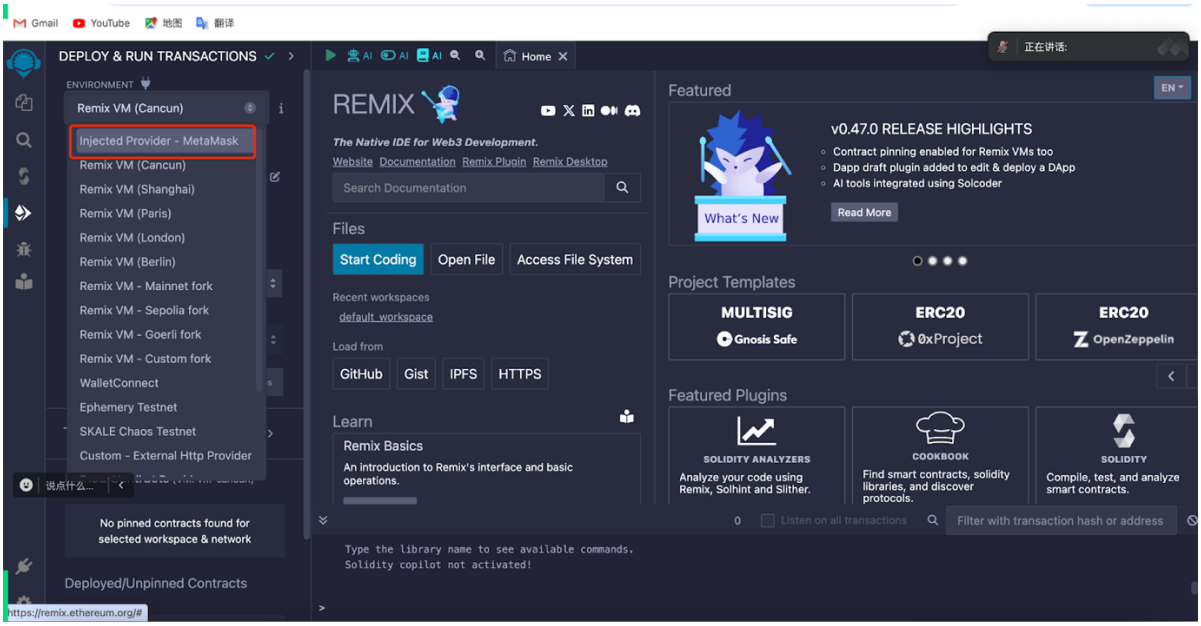


private key：https://docs.goquorum.consensys.io/tutorials/quorum-dev-quickstart/using-the-quickstart

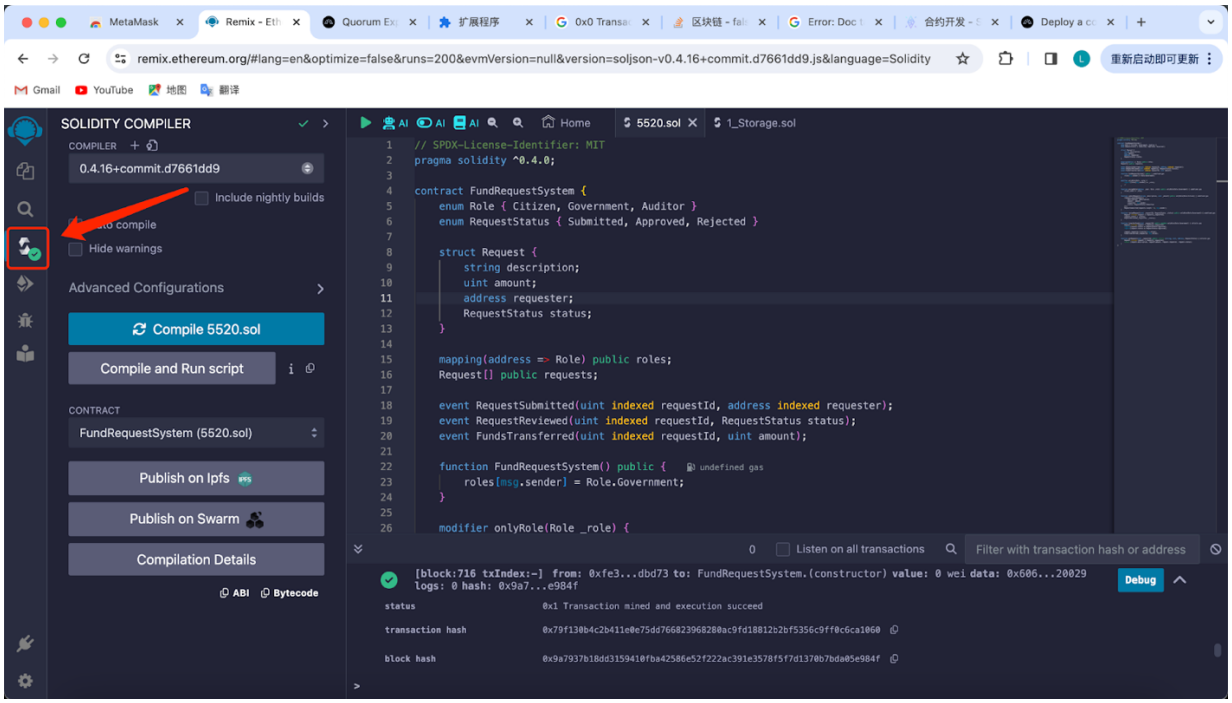


Import accounts using private keys generated during the Quorum setup to test transactions.

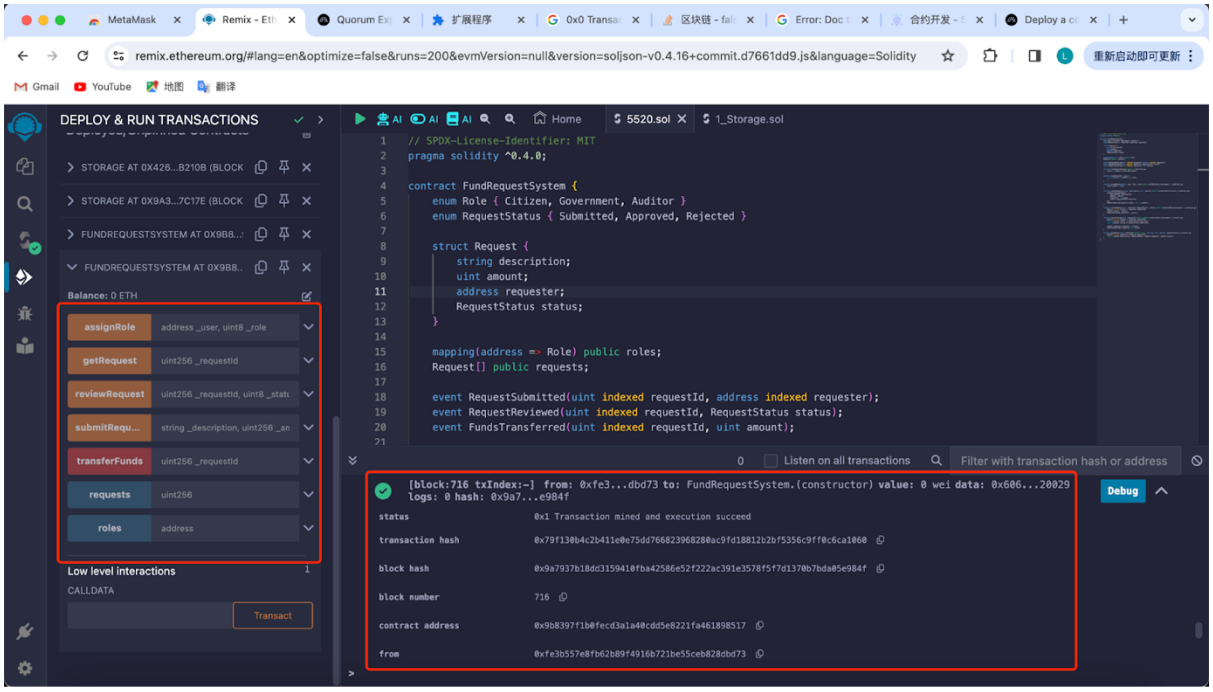
Step 8: Select Injected Provider - Metamask (Environment) to use Remix IDE to compile the smart contract.



Step 9: Compile smart contract



Step 10: Deploy compiled contracts to the Quorum network through Remix, using MetaMask to manage transactions.



Benefits

Transparency

Blockchain technology provides a transparent and tamper-proof ledger that records all transactions and fund allocations. Based on this ledger, all transactions are visible and auditable, enhancing accountability by allowing citizens and stakeholders to track the flow of funds in real time. This transparency reduces the risks of corruption, fraud, and misappropriation of funds. The permanent record of each transaction enables auditors and investigators to promptly detect and trace any suspicious or unauthorized activities, tracking the flow of funds from the source to the ultimate beneficiaries, thereby enhancing financial transparency. In Mechanism for fund requests and review, company name, contact, industry, project information, request amount, requestor's address, and request status are all written into smart contracts. Instead of tracking only the state of currency ownership, we tracks the state transitions of a general-purpose data store.

Security

Advanced encryption technology is employed to safeguard transactions and data. The decentralized nature of blockchain makes it highly resistant to hacking and tampering. By implementing a blockchain-based system, the government can ensure the integrity of its fund allocation process, preventing unauthorized modifications or manipulation. We adopt Authority control mechanism and Multi-signature approval mechanism to make sure the security of our system. We set access rights at the beginning and prevent information tampering and leakage by means of multi-signature encryption.

Efficiency

Blockchain eliminates intermediaries and manual paperwork, streamlining the fund allocation process. Smart contracts stored on the blockchain can automate various tasks, such as verifying criteria, disbursing funds, and enforcing compliance regulations. This automation reduces administrative burdens, minimizes human errors, and ultimately achieves cost savings and improved fund efficiency. Dynamic Redistribution of Funds Mechanism makes timely adjustments to the redistribution of government funds in a flexible manner, ensures maximum utilization of funds, reduces the time required for the approval and declaration of funds, and accelerates the process of fund disbursement.

Social Impact

Government

Government agencies can benefit from the implementation of such a system in two key ways. Firstly, the system allows for enhanced transparency in fund allocation, reducing the occurrence of corruption and facilitating more accurate processes for holding accountable those responsible for unfair distribution of funds. Secondly, the blockchain-based system simplifies the fund allocation process by reducing intermediaries and cumbersome manual paperwork. This, in turn, it will improve the efficiency of government agencies and lead to cost savings.

Citizens

The security of user funds will experience significant improvement, and their trust in government fund management will be enhanced. The increased efficiency of government operations also means that users can expect faster application processing and fund disbursement, thereby improving their financial turnover efficiency and creating greater wealth value.

The Public

The public will have increased opportunities for participation and oversight in government actions and financial decisions. They can track the flow and utilization of funds through the system, question any irregularities in fund usage, and ensure that government decisions and actions align with the public interest. This enhanced transparency empowers the public to actively engage in the governance process and hold the government accountable for its financial practices.

Limitations and Further Improvement

Even though blockchain tracking systems have perfected many issues compared to traditional systems, there is still much room for improvement. Data Privacy and Security have not been fully addressed, blockchain offers data immutability, but ensuring data privacy is not compromised when handling sensitive information remains a challenge Especially when dealing with sensitive financial data from the government. While blockchain technology has shown great potential in the financial sector, its scalability for widespread use at the government level is still a problem. Performance bottlenecks and the risk of system crashes when processing large numbers of transactions are also important considerations. Another issue is the legal and regulatory compliance issue. The legal status and regulatory framework of blockchain technology have not been harmonized globally, which could affect project implementation and government adoption. All of these limitations can hinder the implementation of the tracking fund system and cause the practice to become less efficient or fail. They are very important and therefore it is necessary to take the appropriate measures to deal with them.

Here are the further improvements for the blockchain tracking fund system. A continuous maintenance and upgrade strategy is essential to enable the system to adapt to future technological developments and policy changes, which is crucial to maintaining the continued effectiveness and relevance of the system. Examining how to effectively integrate blockchain systems with existing government information systems to assure seamless data migration and interoperability is capable of improving system utility and minimizing implementation resistance. Apart from that, further development of the technology is quite significant. Learning and developing blockchain technology that is more suitable for the government and able to handle more sophisticated logic and conditions will contribute to better meeting the needs of government funds management.

Q&A

1. What is the complexity of the government fund arrangement? What is the advantage of using blockchain in this area?

The complexity of government fund arrangement lies in the intricate processes involved in allocating funds, ensuring transparency, preventing fraud and corruption, and maintaining accountability. Traditional systems often involve manual paperwork, multiple intermediaries, and opaque processes, leading to inefficiencies and potential mismanagement of funds. The advantage of using blockchain in this area is that it provides a transparent, tamper-proof ledger that records all transactions and fund allocations. This transparency reduces the risks of corruption, fraud, and misappropriation of funds by allowing stakeholders to track the flow of funds in real time. Additionally, blockchain streamlines the fund allocation process by eliminating intermediaries and manual paperwork, enhancing efficiency and reducing administrative burdens.

2. Why does it need to use blockchain? What blockchain technology is used?

Blockchain technology offers several advantages in this context, including transparency, security, efficiency, and accountability. The blockchain technology used in this case is Quorum, a permissioned blockchain platform based on Ethereum. Quorum supports private transactions and contracts, ensuring confidentiality where necessary, while also providing compatibility with Ethereum tools and DApps, facilitating easier integration and development.

3. Why multi-signature is required?

Multi-signature is required to enhance security and prevent unauthorized modifications or manipulation of transactions. In a multi-signature scheme, multiple signatures (authorizations) are required to approve a transaction, typically from different authorized parties. When the government allocates funds, multi-signature ensures that fund disbursements are approved by designated government stakeholders, preventing any single entity from making unilateral decisions. This adds an extra layer of security and accountability to the fund allocation process, reducing the risk of fraud or misuse of funds.

4. What is the use of dynamic re-distribution?

Dynamic re-distribution is used to optimize the allocation of government funds across different projects or departments in a flexible manner. This mechanism allows for timely adjustments to fund redistribution based on changing fiscal priorities or emerging needs. By reallocating funds dynamically, the government ensures maximum utilization of funds, reduces the time required for fund approval and disbursement, and accelerates the process of fund allocation. Dynamic re-distribution enhances efficiency in fund management and enables the government to respond quickly to evolving circumstances or emergencies.

5. How to perform anomaly detection? Based on what method and what flag?

Anomaly detection is performed through two primary methodologies: on-chain and off-chain anomaly detection.

On-Chain Anomaly Detection: An automated system within the smart contract continuously analyzes funding patterns. The system identifies and flags anomalies or deviations from the set budgetary guidelines.

Off-Chain Anomaly Detection: External data is integrated into the blockchain to identify inconsistencies that may not be apparent from on-chain data alone. Governments can use off-chain data analytics tools to analyze funding patterns and detect anomalies. These tools can handle more complex datasets and analyses than on-chain contracts. When an anomaly is detected by these tools, they can send a report or alert to the blockchain via an oracle, which may trigger on-chain actions or investigations.

On-Chain Flags Examples: 1) Budget Cap Exceeded: A smart contract flags any transaction that exceeds the budget cap for a project category, e.g., a transaction of $120,000 is flagged if the cap is $100,000. 2) Sudden Increase in Transaction Volume: If a department typically transacts $10,000 per week and suddenly initiates transactions totaling $50,000 in a single day, this could be flagged. 3) Multi-Signature Approval Failure: A transaction requiring approval from multiple parties proceeds with only one signature, which is flagged.

Off-Chain Flags Examples: 1) Economic Indicator Discrepancy: If economic indicators suggest a downturn, yet a department's spending on certain projects increases, this discrepancy could be flagged. 2) Deviation from Peer Groups: If a department's spending significantly deviates from that of similar departments, this could indicate a potential anomaly. 3) Historical Spending Anomalies: If a department's spending on a particular category is consistently $50,000 per month but suddenly drops to $5,000 without explanation, this could be flagged.

6. There is some privacy issue which is not able to allow the government to transparent the fund allocation, how can you solve it?

Firstly, anonymization techniques can be employed to remove personally identifiable information from the data before disclosing it. Secondly, the government can redact or exclude specific sensitive information from public disclosure. Thirdly, we can enhance our secure data infrastructure. It includes using encryption, access controls, and secure data storage practices to prevent unauthorized access or data breaches.

7. What is the main consideration for choosing Quorum as the network platform?

The primary consideration for choosing Quorum as the network platform for this blockchain-based government fund allocation and tracking system is its enhanced privacy capabilities. Quorum offers private transactions, which are crucial for maintaining confidentiality over

transaction details while still leveraging the transparency and security benefits of blockchain technology. This feature makes Quorum particularly suitable for applications where data sensitivity is paramount, such as in government financial transactions where privacy and security need to be meticulously balanced with transparency.

8. What are the key benefits of using blockchain for government fund allocation and tracking?

1. Transparency: Blockchain provides an immutable and transparent record of all transactions. This transparency allows for real-time tracking of fund flows and ensures that all allocations are visible and auditable, significantly reducing the chances of corruption or misappropriation of funds.

2. Security: The decentralized nature of blockchain makes it highly resistant to tampering and unauthorized changes. This enhances the security of the financial data and fund allocations.

3. Efficiency: By automating processes like fund distribution through smart contracts, blockchain reduces the need for intermediaries, lowers administrative costs, and minimizes human error. This can result in more streamlined operations and faster execution of financial transactions.

9. How can blockchain technology help prevent corruption and misallocation of resources?

Blockchain technology contributes to the prevention of corruption and misallocation of resources in several ways:

1. Immutable record keeping: Once a transaction is recorded on the blockchain, it cannot be altered or deleted. This immutability helps prevent the falsification of transaction records, ensuring that funds are used as intended.

2. Smart contracts: These are self-executing contracts with the terms directly written into code. They can be programmed to release funds only when certain predefined conditions are met, reducing the risk of fraud and ensuring that funds are only used for their intended purposes.

3. Enhanced Oversight: The use of blockchain allows for better oversight of fund distribution and utilization. Auditors and regulators can access a real-time, unalterable audit trail, which helps in quickly identifying and investigating discrepancies or irregularities.

10. Compared with companies, governments tend to be more cautious about dramatic changes such as applying blockchains to the current system. In the past few years, there have already been many cybersecurity accidents such as the collapse of FTX and the scams done by JPEX. Suppose you are negotiating with some influential government officials; how would you adjust your project to motivate them to try this blockchain-based technology since it introduces both improvements and potential financial risks?

To address the concerns of blockchain implementation and motivate officials to adopt blockchain, the emphasis could be on the security, transparency, and efficiency that blockchain can bring to government operations, distinguishing it from the decentralized, less-regulated environments like those of FTX and JPEX. It's important to clarify that the proposed blockchain system—like Quorum—is designed for enterprise use with advanced security features and permissioned network access, which mitigates many of the risks associated with public blockchains. By showcasing successful implementations of blockchain technology in other governmental and financial institutions, we can demonstrate proven stability and security, reassuring officials of its viability and robustness.

Furthermore, the project can further include a detailed risk management plan that outlines steps to mitigate potential financial and operational risks. This plan would highlight the system's design to prevent unauthorized access, fraud, and misallocation of resources, through features such as encrypted transactions, multi-signature approvals, and an immutable ledger. The addition of a pilot project phase can be particularly appealing, allowing the government to test the technology on a smaller scale without significant upfront investment or full-scale implementation risks. This phased approach enables officials to witness firsthand the benefits of blockchain, such as increased transparency in fund allocation and enhanced efficiency through automation, while also providing ample opportunity to adjust the system based on initial results and feedback.

Lastly, the conversation can further focus on the long-term strategic advantages of adopting blockchain technology, such as cost savings, improved citizen trust through enhanced transparency, and the positioning of the government as a modern and technologically advanced institution. Discussing the broader economic and social benefits, including potential improvements in public satisfaction and international competitiveness, can help shift the focus from the perceived risks to the substantial gains. Engaging officials with a vision of a more accountable and efficient government can encourage them to consider blockchain as a transformative tool rather than a disruptive risk. By aligning the technology's capabilities with the government's mission to serve the public effectively, the project can create a compelling case for blockchain's role in modernizing government systems.

References

Abodei, E., Norta, A., Azogu, I., Udokwu, C., & Draheim, D. (2019). Blockchain technology for enabling transparent and traceable government collaboration in public project processes of developing economies. In Digital Transformation for a Sustainable Society in the 21st Century: 18th IFIP WG 6.11 Conference on e-Business, e-Services, and e-Society, I3E 2019, Trondheim, Norway, September 18–20, 2019, Proceedings 18 (pp. 464-475). Springer International Publishing.

Castro, M., & Liskov, B. (1999). Practical Byzantine Fault Tolerance. Proceedings of the Third Symposium on Operating Systems Design and Implementation, 173-186.

Christidis, K., & Devetsikiotis, M. (2016). Blockchains and smart contracts for the Internet of Things. IEEE Access, 4, 2292-2303.

Dai, J., & Vasarhelyi, M. A. (2017). Toward blockchain-based accounting and assurance. Journal of Information Systems, 31(3), 5-21.

Gatteschi, V., Lamberti, F., Demartini, C., Pranteda, C., & Santamaría, V. (2018). Blockchain and smart contracts for insurance: Is the technology mature enough?. Future Internet, 10(2), 20.

Mougayar, W. (2016). The business blockchain: Promise, practice, and application of the next Internet technology. John Wiley & Sons.

Swan, M. (2015). Blockchain: Blueprint for a new economy. O'Reilly Media, Inc.

Trenovski, B. (2018). Fiscal transparency, accountability and institutional performances as a foundation of inclusive and sustainable growth in Macedonia.

Singh, M. S., & Teotia, M. (2018). Tracking Government Fund Allocation Using Blockchain Approach: A review. system, 10.

Xiao, H., & Wang, X. (2024). Fiscal transparency practice, challenges, and possible solutions: lessons from Covid 19. Public Money & Management, 44(3), 196–207.