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

Group 6

Cen Baihui	*1155197612*	*Jiang Tianyun*	*1155197611*
Chen Angyu	*1155197620*	*Liang Haojin*	*1155200065*
Chen Yunfan	*1155198241*	*Niu Xinyan*	*1155201239*
Huang Shiqi	*1155204275*		

Background

Importance of Transparency & Accountability

Government

- Manager of national resources & Implementer of financial scheme
- Corruption hinders successful project delivery

The International Monetary Fund (IMF) &
The Organization for Economic Cooperation and Development (OECD)

Transparency & Accountability

Better Macroeconomic & Fiscal Stability

Government

States have taken corresponding measures to enhance
(Eg. In 2014, the United States required federal departments to **disclose financial information** on a quarterly basis on a government website.)

Why Blockchain?

Only public disclose → Prevent → Financial mismanage

✖ Not enough

Examples	
US	Payment Protection Program (PPP) lent to unrelated entities
Over 120 countries	Incomplete disclosure of information during the Covid-19 period

Problems in traditional government fund tracking system

- □ **Inefficient allocation of funds and reporting**
- □ **High risk of mistake and fraud**
- □ **High cost and labor-intensive**
- □ **Slow data sharing creates information silos**

Need a more robust, transparent, and traceable system

Project Goals & Framework

NIU Xinyan

Project Goals

- Develop a system for real-time tracking of government fund allocation.
- Streamline government fund allocation process.
- Enhance data security and integrity in government financial transactions.

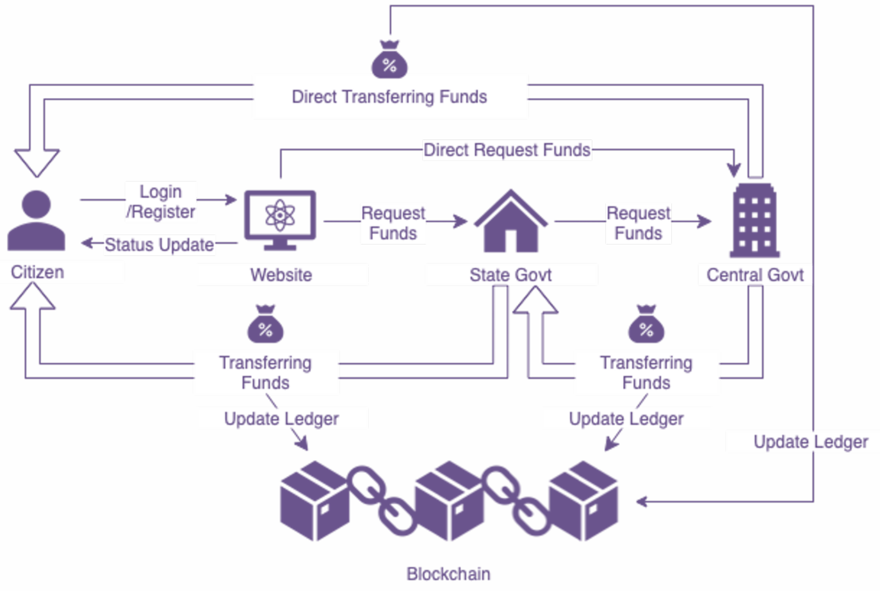
Blockchain

↓

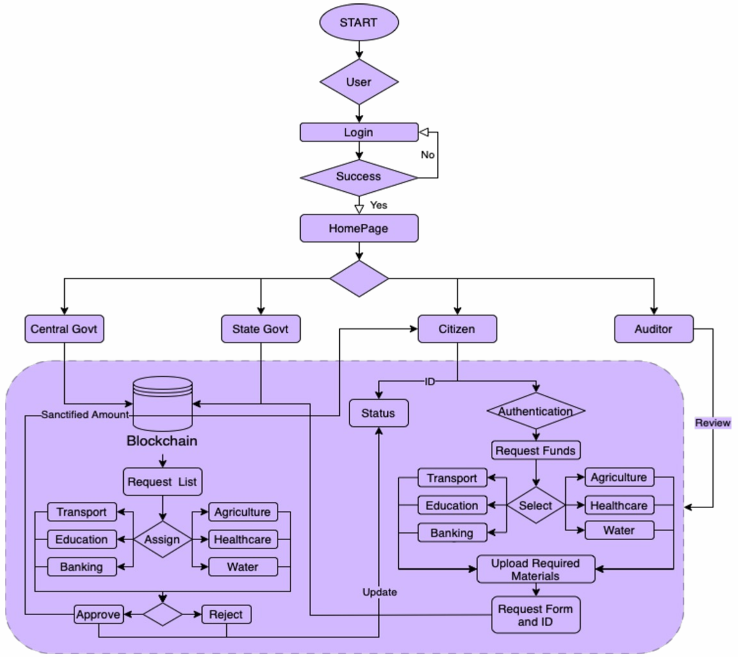
Transparency

Accountability ✓

Flowchart of Funding Process



Architecture Diagram



System Design - Overview

Ethereum Consortium Chain

Platform: Quorum

- Permissioned blockchain
- Multiple consensus mechanisms
- Ethereum ecosystem compatibility

Consensus: IBFT

- Immediate finality
- Energy efficiency
- Permissioned participation

Nodes Configuration

- Decentralized control
- Role specialization
- Security and compliance

Main Functionalities

All Entities

Authority Control

- Categorize users into distinct roles
- Different access and functionality for different roles

Citizens

Request Funds

- Create fund requests
- View fund requests status

Government

Manage Funds

- Automatically release funds after approvals
- Dynamic fund redistribution

Auditors

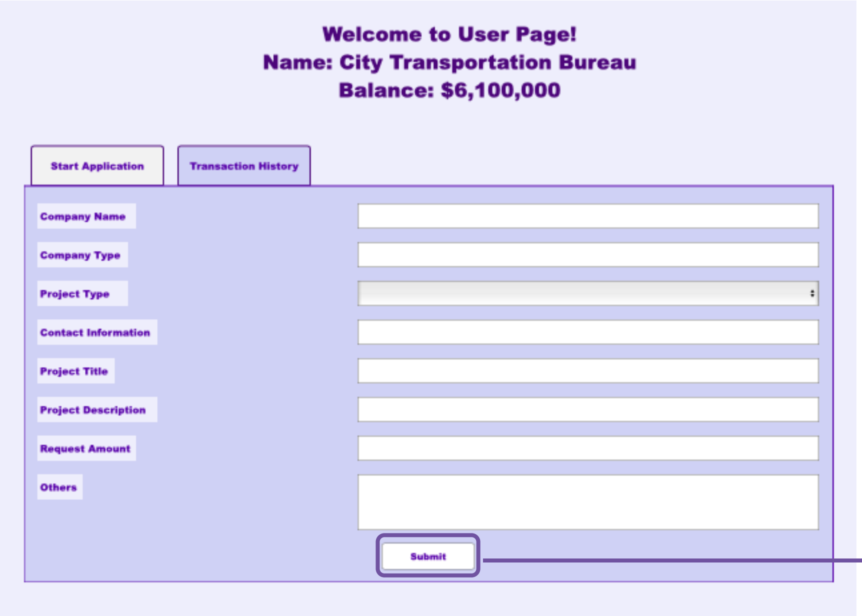
Supervise Funding

- Identify unusual funding patterns, discrepancies, or deviations from approved amount

Prototyping

User Page

Start Application List



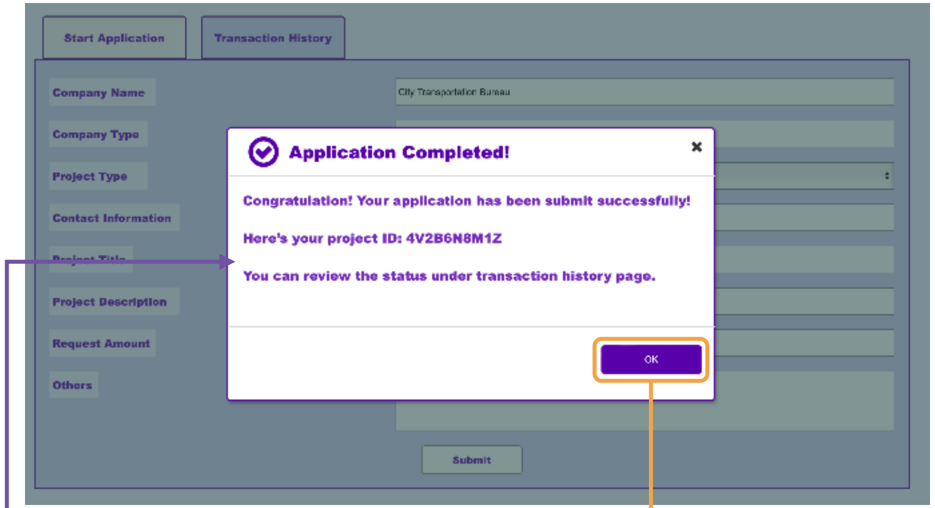
Welcome to User Page!
Name: City Transportation Bureau
Balance: $6,100,000

Key Consideration

1 **Normative Inspections:**
Each item cannot be empty
Company Name = Applicant Name ≠ Name(User Name)

2 **Pushed to government department accounts based on ‘Project Type’:**
Applicants can only apply within the optional

Application Completed Pop-up Window



City Transportation Bureau

Application Completed!

Congratulation! Your application has been submit successfully!

Here’s your project ID: 4V2B6N8M1Z

You can review the status under transaction history page.

Key Consideration

1 **Return Project ID:**
The randomly generated project id is an important indicator for retrieving this application

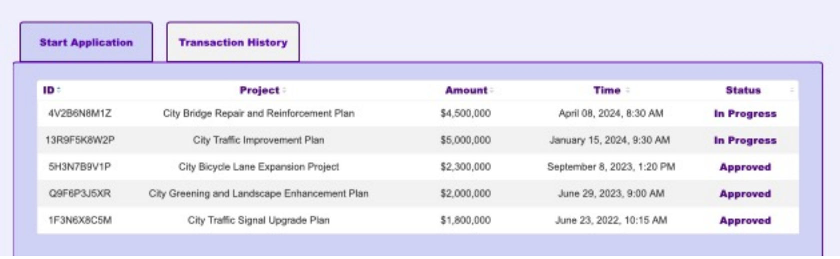
2 **Jump Logic:**
Click on the ’OK’ button to be redirected to the trade history module.

Key Consideration

1 **Time:**
‘Time’ is the time of the last status update of the application form

2 **Status:**
‘In Progress’ means that the application has not yet been approved, i.e. it may be under review, pending revision, etc. ‘Approved’ means that the application has been approved and government funds have been remitted to the wallet address and the amount is shown in ‘Balance’

Transaction History



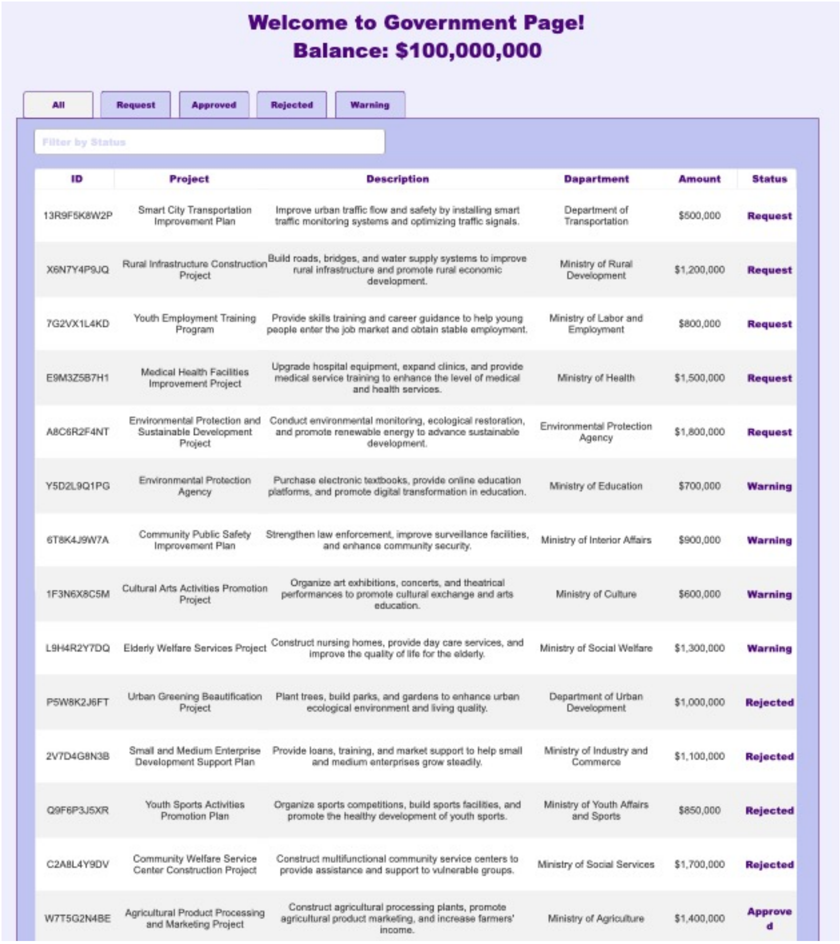
ID	Project	Amount	Time	Status
4V2B6N8M1Z	City Bridge Repair and Reinforcement Plan	$4,500,000	April 08, 2024, 8:30 AM	In Progress
13R9F5K8W2P	City Traffic Improvement Plan	$5,000,000	January 15, 2024, 9:30 AM	In Progress
5H3N7B9V1P	City Bicycle Lane Expansion Project	$2,300,000	September 8, 2023, 1:20 PM	Approved
Q9F6P3J5XR	City Greening and Landscape Enhancement Plan	$2,000,000	June 29, 2023, 9:00 AM	Approved
1F3N6X8C5M	City Traffic Signal Upgrade Plan	$1,800,000	June 23, 2022, 10:15 AM	Approved

Prototyping

NIU Xinyan

Government Page

All Applications List



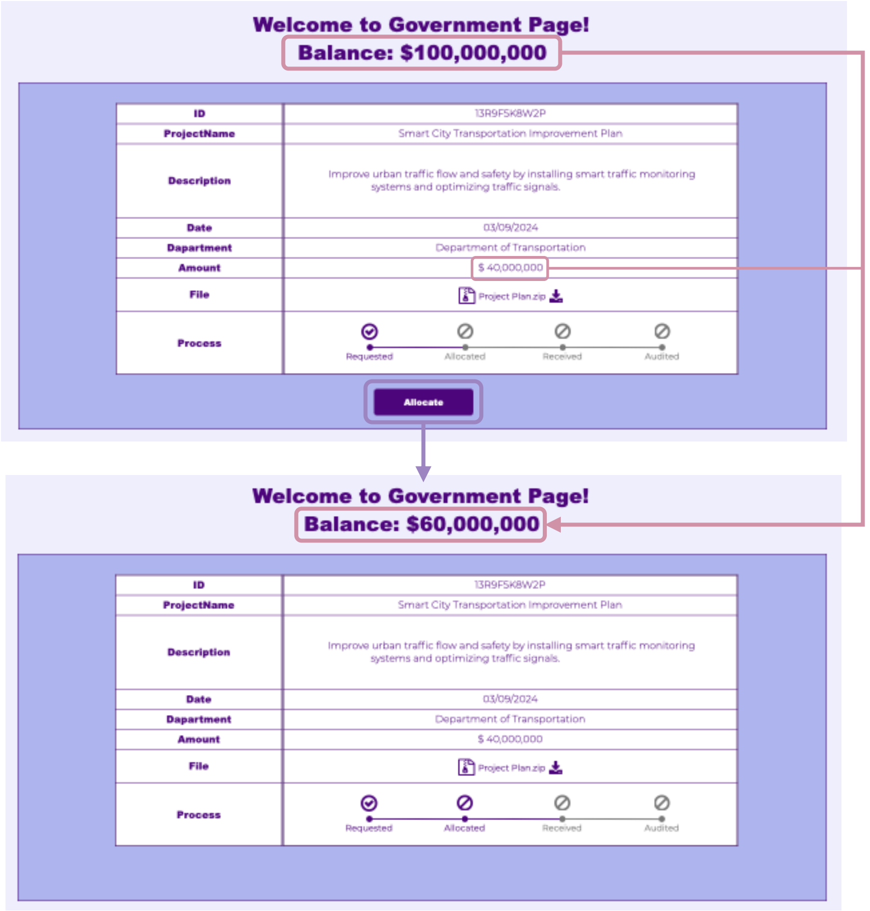
Welcome to Government Page!
Balance: $100,000,000

ID	Project	Description	Dapartment	Amount	Status
13R9F5K8W2P	Smart City Transportation Improvement Plan	Improve urban traffic flow and safety by installing smart traffic monitoring systems and optimizing traffic signals.	Department of Transportation	$500,000	Request
X6N7Y4P9JQ	Rural Infrastructure Construction Project	Build roads, bridges, and water supply systems to improve rural infrastructure and promote rural economic development.	Ministry of Rural Development	$1,200,000	Request
7G2VX1L4KD	Youth Employment Training Program	Provide skills training and career guidance to help young people enter the job market and obtain stable employment.	Ministry of Labor and Employment	$800,000	Request
E9M3Z5B7H1	Medical Health Facilities Improvement Project	Upgrade hospital equipment, expand clinics, and provide medical service training to enhance the level of medical and health services.	Ministry of Health	$1,500,000	Request
A8C6R2F4NT	Environmental Protection and Sustainable Development Project	Conduct environmental monitoring, ecological restoration, and promote renewable energy to advance sustainable development.	Environmental Protection Agency	$1,800,000	Request
Y5D2L9Q1PG	Environmental Protection Agency	Purchase electronic textbooks, provide online education platforms, and promote digital transformation in education.	Ministry of Education	$700,000	Warning
6T8K4J9W7A	Community Public Safety Improvement Plan	Strengthen law enforcement, improve surveillance facilities, and enhance community security.	Ministry of Interior Affairs	$900,000	Warning
1F3N6X8C5M	Cultural Arts Activities Promotion Project	Organize art exhibitions, concerts, and theatrical performances to promote cultural exchange and arts education.	Ministry of Culture	$600,000	Warning
L9H4R2Y7DQ	Elderly Welfare Services Project	Construct nursing homes, provide day care services, and improve the quality of life for the elderly.	Ministry of Social Welfare	$1,300,000	Warning
P5W8K2J6FT	Urban Greening Beautification Project	Plant trees, build parks, and gardens to enhance urban ecological environment and living quality.	Department of Urban Development	$1,000,000	Rejected
2V7D4G8N3B	Small and Medium Enterprise Development Support Plan	Provide loans, training, and market support to help small and medium enterprises grow steadily.	Ministry of Industry and Commerce	$1,100,000	Rejected
Q9F6P3J5XR	Youth Sports Activities Promotion Plan	Organize sports competitions, build sports facilities, and promote the healthy development of youth sports.	Ministry of Youth Affairs and Sports	$850,000	Rejected
C2A8L4Y9DV	Community Welfare Service Center Construction Project	Construct multifunctional community service centers to provide assistance and support to vulnerable groups.	Ministry of Social Services	$1,700,000	Rejected
W7T5G2N4BE	Agricultural Product Processing and Marketing Project	Construct agricultural processing plants, promote agricultural product marketing, and increase farmers' income.	Ministry of Agriculture	$1,400,000	Approved

Key Consideration

- □ List all application forms in **tabular form**
- □ Can be **filtered** by ‘Status’
- □ Tap into a row to view the **details** of the application
- □ The details page allows you to **download** the project proposal submitted by the applicant (multiple file formats are supported) to your local area.
- □ Agreeing to the appropriation will result in a change in the government's **balance**

Application Detail



Welcome to Government Page!
Balance: $100,000,000

ID	13R9F5K8W2P
ProjectName	Smart City Transportation Improvement Plan
Description	Improve urban traffic flow and safety by installing smart traffic monitoring systems and optimizing traffic signals.
Date	03/09/2024
Dapartment	Department of Transportation
Amount	$ 40,000,000
File	Project Plan.zip
Process	✓ Requested ⊘ Allocated ⊘ Received ⊘ Audited

Welcome to Government Page!
Balance: $60,000,000

ID	13R9F5K8W2P
ProjectName	Smart City Transportation Improvement Plan
Description	Improve urban traffic flow and safety by installing smart traffic monitoring systems and optimizing traffic signals.
Date	03/09/2024
Dapartment	Department of Transportation
Amount	$ 40,000,000
File	Project Plan.zip
Process	✓ Requested ⊘ Allocated ⊘ Received ⊘ Audited

Smart Contract

CEN Baihui & JIANG Tianyun

Authority Control Mechanism

Citizens: submit and review

```
// 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: review, decide, transfer funds, assign roles

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

Auditors: read-only

```
// Auditors can directly view requests through getter functions
```

Fund Request and Review

`createFundRequest` function to accept the parameters required for submitting a request

```
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
}));
}
```

`viewFundRequest` for citizens to view their request status through each unique request id

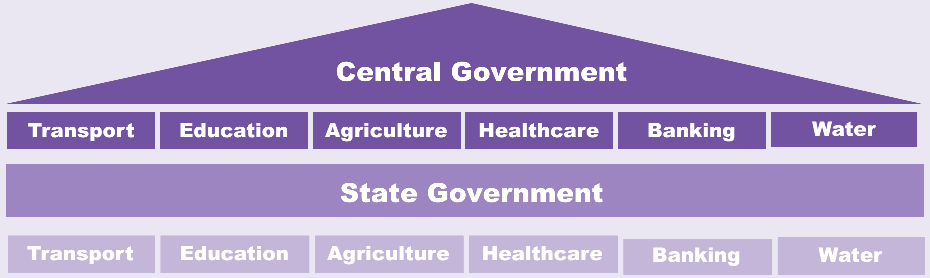
```
function viewFundRequest(uint _requestId) external view returns (FundRequest memory) {
require(_requestId < fundRequests.length, "Request ID is out of bounds");
return fundRequests[_requestId];
}
```

Smart Contract

Multi-signature Approval Mechanism

```
// Define Roles
enum Role { StateGovernment, CentralGovernment }
enum Department { Transport, Education, Banking, Agriculture, Healthcare, Water }

// 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;
}
}
```

1 Department Structure



2 Approval Regulations

Amount >= $50,000:

- approvals from state government and central government are both required

Amount < $50,000:

- only the state government's approval is needed

3 Fund Transfer

Transfer route:

- Government => Designated department => Requestor's account

Request status:

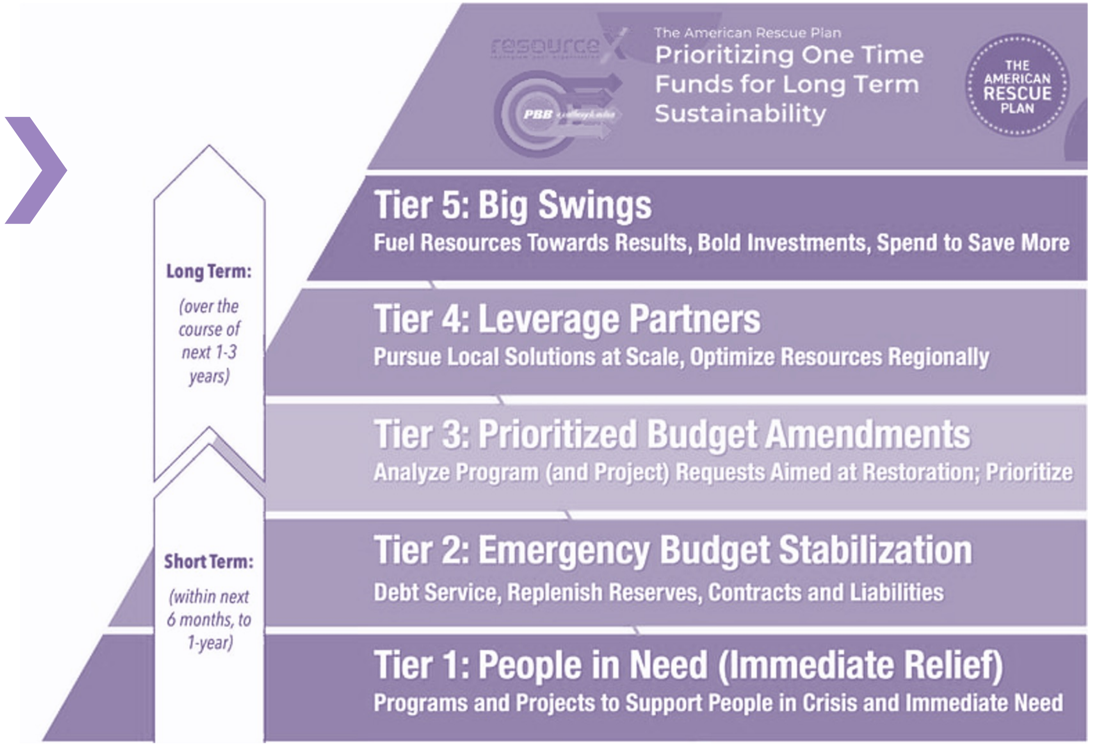
- Update to "Approved"

Smart Contract

Dynamic Redistribution of Funds

```
// 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();
}
```

Provides a flexible framework that allows for funds reallocation across different projects or departments swiftly


resource X
PBB
The American Rescue Plan
Prioritizing One Time Funds for Long Term Sustainability
THE AMERICAN RESCUE PLAN
Tier 5: Big Swings
Fuel Resources Towards Results, Bold Investments, Spend to Save More
Tier 4: Leverage Partners
Pursue Local Solutions at Scale, Optimize Resources Regionally
Tier 3: Prioritized Budget Amendments
Analyze Program (and Project) Requests Aimed at Restoration; Prioritize
Tier 2: Emergency Budget Stabilization
Debt Service, Replenish Reserves, Contracts and Liabilities
Tier 1: People in Need (Immediate Relief)
Programs and Projects to Support People in Crisis and Immediate Need
Long Term:
(over the course of next 1-3 years)
Short Term:
(within next 6 months, to 1-year)

Smart Contract

CEN Baihui & JIANG Tianyun

To detect anomalies that may indicate irregularities or deviations from the approved budget norms

Strategy One

Simplified On-Chain Checks

- □ Real-time on-chain monitoring
- □ Automatic flagging of suspicious transactions
- □ Dual oversight by governments and auditors

Strategy Two

Oracles for Off-Chain Anomaly Detection

- □ Integration of external data for sophisticated analysis
- □ Detection of inconsistencies not evident from on-chain data alone

Simplified On-Chain Checks for Anomaly Detection

Government Supervision

```
# 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
```

Auditor Supervision

```
# 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
```

Smart Contract

CEN Baihui & JIANG Tianyun

Oracles for Off-Chain Anomaly Detection

Government Supervision

```
# 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
```

Auditor Supervision

```
# 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
```

Supervision Review

Government Supervision

- □ **Use off-chain data analytics tools to analyze funding patterns**
- □ **Send reports/alerts to the blockchain through oracles**

Auditor Supervision

- □ **Independently verify oracle findings or conduct analyses using off-chain data**

Key Conclusion

1 **Complementary Methodologies:** Simplified On-Chain Checks + Off-Chain Anomaly Detection work together

2 **Routine On-Chain Surveillance:** On-chain checks for real-time compliance monitoring

3 **Advanced Off-Chain Analysis:** Leverage external data and advanced analytics (statistical models, machine learning) to detect potential fraud

Benefits and Impact(1/2)

Transparency

1. Records all transactions and fund allocations.

2. All transactions are visible and immutable.

3. Reduces the risks of corruption, fraud, and misappropriation

Mechanism for fund requests and review

Security

1. Advanced encryption technology.

2. Decentralized nature of blockchain.

3. Ensure the integrity of its fund allocation process.

Authority control mechanism

Efficiency

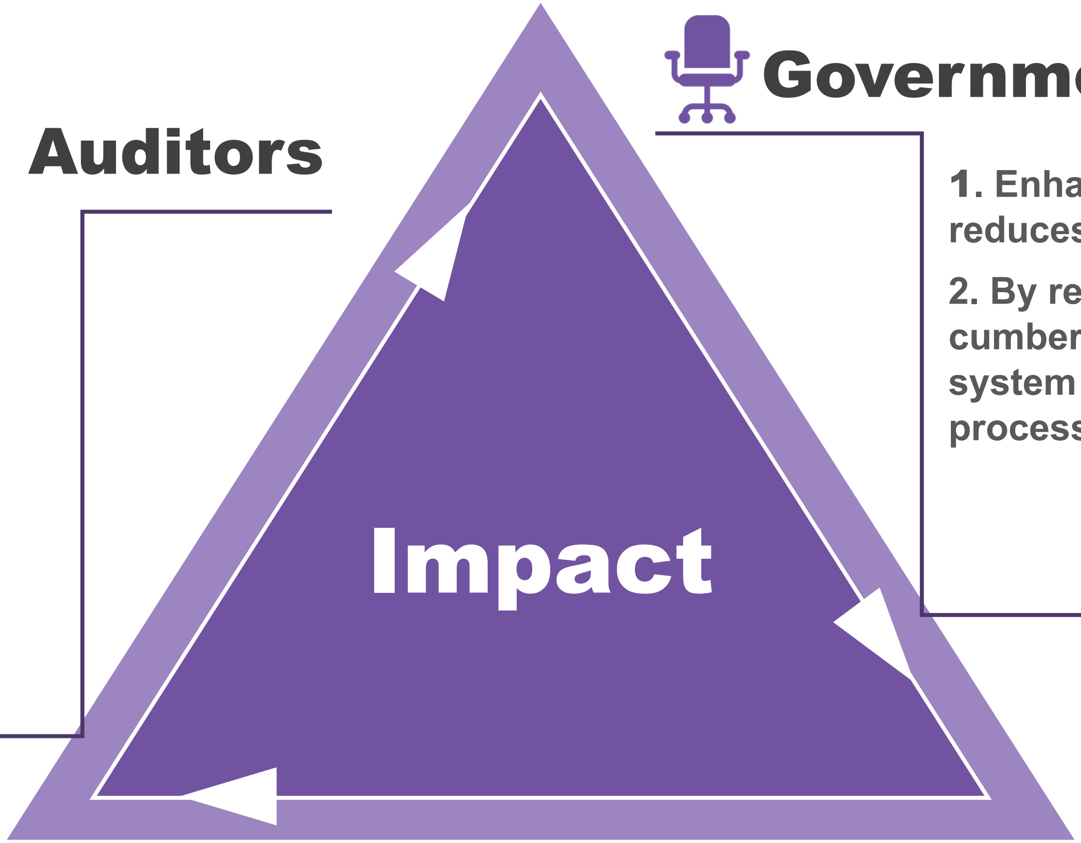
1. Automation of various tasks. e.g. verifying eligibility criteria and distributing funds.

2. Reduces administrative burdens.

3. Minimizes human errors.

Multi-signature approval mechanism

Benefits and Impact(2/2)



Auditors

1. Track the flow and utilization of funds through the system.

2. Question any irregularities in fund usage

3. Ensure that government decisions and actions align with the public interest.

Government

1. Enhanced transparency in fund allocation reduces the occurrence of corruption.

2. By reducing intermediaries and cumbersome manual paperwork, the system simplifies the fund allocation process.

Citizens

1. Improve security of their funds and enhance their trust in government fund management.

2. Expect faster application processing and fund disbursement.

Limitations & Further Improvement

Limitations

Data Privacy and Security

Blockchain offers data immutability, but ensuring data privacy is not compromised when handling sensitive information remains a challenge

Scalability Issues

Performance bottlenecks and risk of system crashes when processing large numbers of transactions

Legal and Regulatory Compliance

Legal and regulatory frameworks for blockchain have not been harmonized globally, potentially affecting project implementation and government adoption

Further Improvement

Continuous Maintenance and Upgrade Strategy

Ensure that the system can adapt to future technological developments and policy changes

System Integration and Interoperability

Effective integration of blockchain systems with government information systems to ensure seamless data migration and interoperability

Improvements in Technology

Further development of more government-friendly blockchain technology capable of handling more complex logic and conditions

Thank You!

Group 6

Cen Baihui	1155197612	Jiang Tianyun	1155197611
Chen Angyu	1155197620	Liang Haojin	1155200065
Chen Yunfan	1155198241	Niu Xinyan	1155201239
Huang Shiqi	1155204275		