A Review of 'Blockchain-Based Farmer's Fund Distribution System'

SHRUTI MAHENDRA RASKAR[1], ANUSHKA MOHAN JADHAV[2], TANUJA SANJAYZAGADE[3], RUTUJA RAJENDRA TARATE[4], PROF. TUSHAR D. KOLHE[5]
[1, 2, 3, 4, 5] *Navsahyadri Group of institutes Collage of Engineering Naigaon Pune*

Abstract- This review paper provides an in-depth analysis of the study by Raskar et al., which presents a blockchain-enabled platform for farmer-centric subsidy disbursement and fund tracking. We expand on the original work by detailing additional blockchain advancements—such as Layer 2 scaling solutions, zero-knowledge proofs for privacy, and decentralized identity frameworks—and critically assess system architecture, implementation on Angular instead of the originally proposed React JS, and testing rigor. While the original contributes a well-defined roadmap for mobile and web interfaces, our review identifies opportunities for performance benchmarking, governance models, and modular smart contract design to enable upgradability. We conclude by outlining research trajectories to strengthen scalability, enhance privacy, and align with regulatory requirements for agricultural finance.

Indexed Terms- Farmers' fund distribution, Blockchain review, Fund distribution, Transparency, Accountability, System evaluation, Smart contracts, Layer 2 scaling, Decentralized, identity

I. INTRODUCTION

Small and marginal farmers often suffer from delays or distribution of opaque subsidies due to centralized bureaucratic processes. These systemic disabilities contribute to financial uncertainty, distrust the government's initiative and reduce agricultural productivity. Original task of Raskar et al. To solve these challenges in the situation as a transformative technique correctly, to solve these challenges in end-to-end audits of automatic payout and transaction through smart contracts.

In this review, we analyze their system through a modern technical lens and located it within the widespread development of decentralized finance (DEFI) and digital public infrastructure. We discuss how the openness and decentralization of blockchain can provide responsibility, while smart contracts automate compliance and reduce human errors. In addition, we discover the relevance of advanced techniques as decentralized identifier (DID), zero knowledge certificate (ZK-Narks) and Create 2 for Rollup scaling and privacy.

We also emphasize the importance of choosing the right framework for distributing such decentralized services to users in the countryside effectively. Angular strong tools, molding skills and support for offline-first-progressive web applications make it a strong challenger for such assignment creation. The purpose of this review is to provide a comprehensive evaluation of the original function, while at the same time suggesting the enrichment that may further adapt the distribution of agricultural agents using modern blockchain devices and contours.

- Synthesis and expand innovations presented by Ruskar et al., Integration of blockchain, especially for distribution of agricultural deficits.

Layer 2 scaling (e.g., Optimistic Rollups and zk-Rollups) to reduce gas fees and increase throughput for micro-transactions between government entities and beneficiaries.

zk-SNARKs for privacy-preserving verification of sensitive farmer credentials and eligibility, maintaining data confidentiality.

Decentralized Identifiers (DIDs) to establish verifiable, self-sovereign identities for farmers, eliminating reliance on centralized KYC.

Proof of Authority (PoA) consensus to enable fast, energy-efficient block validation by designated agricultural authority nodes. • Identify empirical gaps in the original study, such as lack of stress testing and real-time usability assessments, and recommend evaluation metrics including:

Performance: transaction latency, throughput, UI response time.

Security: rate of failed authentications, penetration test coverage, and tamper resistance.

Usability: task completion time, accessibility compliance, and user satisfaction in diverse literacy settings.

II. SUMMARY OF ORIGINAL STUDY

- Mobile Application (Angular/TypeScript): The system delivers a cross-platform Progressive Web App (PWA) built with Angular and TypeScript to ensure consistent UX across Android, iOS, and desktop browsers. It features:

Authentication & Security: Secure SMS-based OTP flows for farmer registration, combined with JWT tokens for session management; sensitive data (e.g., bank details, land records) is client-side encrypted with AES before transmission.

State Management: NgRx is employed to manage application state, enabling time-travel debugging and maintaining reactive data streams for user profiles, scheme catalogs, and transaction statuses.

Reactive Forms & Validation: Dynamic FormBuilder constructs complex forms for geotagged land submissions, supported by custom validators and real-time feedback to reduce input errors.

Offline Functionality: Angular Service Workers cache critical assets and API responses; IndexedDB storage allows offline data entry and background synchronization when connectivity is restored.

Native Device Features: Through Capacitor, the app accesses camera for document capture, GPS for precise land verification, and file system APIs for offline document storage.

Push Notifications & Alerts: Firebase Cloud Messaging integration ensures timely alerts on application status changes, funding approvals, or new scheme announcements.

Logging & Monitoring: Integration with Sentry captures runtime errors and performance metrics; custom logging service records API response times and user interactions for analytics.

Modular Architecture: Lazy-loaded feature modules (e.g., Registration, Scheme Explorer, Transaction History) reduce initial payload and improve load performance on low-bandwidth networks.

Accessibility & Multilingual Support: ARIA-compliant components and Angular i18n enable support for multiple local languages, ensuring inclusivity for diverse farmer populations.

- Web Dashboard (Angular CLI/Visual Studio Code): Government officers at central, state, and local levels manage schemes, review applications, and trigger disbursements. Angular's modular architecture and RxJS streams enable live data visualization.
- Blockchain Layer (Ethereum/Truffle Framework): Smart contracts automate fund allocation, enforce eligibility rules, and record each transaction on-chain. We note the absence of performance metrics for gas consumption and transaction throughput.
- Web Scraping and API Integration: New schemes auto-populate via scheduled scraping jobs and RESTful APIs, ensuring beneficiaries see the latest opportunities.
- Testing Strategy: Described unit, integration, acceptance, and GUI tests. However, test coverage metrics, continuous integration pipeline details, and security auditing steps remain underreported.

III. UNDERSTANDING BLOCKCHAIN TECHNOLOGY

Blockchain is a distributed, decentralized ledger technology that enables secure, transparent, and tamper-proof recordkeeping without the need for a central authority. Originally conceptualized as the underlying infrastructure for Bitcoin in 2008, blockchain has evolved into a powerful foundational technology with applications across finance, supply chain, identity management, and public sector governance.

Core Components of Blockchain

- Ledger: A continuously growing list of records (blocks), each containing a batch of transactions. Once recorded, the data in any given block cannot be altered retroactively.
- Block: A block contains a cryptographic hash of the previous block, a timestamp, and transaction data. This linking forms a secure "chain."
- Node: A participant in the blockchain network, maintaining a full or partial copy of the ledger.
- Consensus Mechanism: A set of protocols that allow all nodes to agree on the state of the ledger. Examples include Proof of Work (PoW), Proof of Stake (PoS), and Proof of Authority (PoA).

Key Features

- Decentralization: Unlike traditional databases controlled by central authorities, blockchain is maintained by a peer-to-peer network.
- Transparency: All transactions are publicly verifiable, increasing accountability.
- Immutability: Once data is written to a blockchain, altering it is practically infeasible due to the cryptographic linkages between blocks.
- Security: Advanced cryptographic techniques ensure that data remains confidential and resistant to tampering.

Smart Contracts

Smart contracts are self-executing pieces of code deployed on the blockchain. They automatically trigger outcomes (e.g., releasing funds) when predefined conditions are met. In the context of farmer fund distribution:

- Eligibility criteria can be encoded as rules.
- Disbursement happens without human intervention.
- Every transaction is traceable and auditable.

Public vs. Private Blockchains

- Public Blockchain: Open to anyone (e.g., Ethereum, Bitcoin). Transactions are visible and verifiable by all.
- Private Blockchain: Access is restricted to specific participants (e.g., government departments or NGOs). Offers more control and faster consensus.

Blockchain in Agriculture and Governance

For schemes like farmer subsidies, blockchain offers:

- Auditability: Each disbursement is recorded permanently.
- Anti-Fraud: Reduces corruption and ghost beneficiary cases.
- Efficiency: Minimizes manual paperwork and disbursing delays.
- Inclusivity: Empowers farmers with digital wallets and verifiable identities.

Emerging Enhancements

Modern blockchain systems now incorporate:

- Layer 2 Scaling: To handle high transaction volumes with reduced costs.
- Zero-Knowledge Proofs: For privacy-preserving verification.
- Decentralized Identity (DID): Self-sovereign digital identity for individuals.
- Cross-Chain Bridges: To connect and transfer data or assets across different blockchain networks.

IV. CRITICAL-ANALYSIS

4.1frontend Framework: Angular vs. React

The angular underlying addiction provides a broad, meaning structure with injection, form module and CLI tools, which can accelerate growth and use the stability of the module. The one -way and two -way data -linking feature, which is combined with additional time (AOT) collection, can reduce the pay size and improve the first load time for mobile networks for distribution of rural.

4.2 Scalability Enhancements with Layer-2 Solutions

The original atherium -based laser can use high gas charges and limited throws (~ 15 TPS). By using Layer 2 Rollup (optimistic or ZK -rollup), the capacity of the transaction can increase by hundreds or thousands per seconds. PageChains or plasma networks can further shut down continuous micro -transport, the man commits the checkpoint with transactions.

4.3 Privacy via Zero-Knowledge Proofs
Zero Knowledge Proof (ZKPS) allows a farmer to prove eligibility (eg land ownership or income level) for smart contracts without highlighting any underlying data. A compact ZK produces a compact ZK Snark certificate from the app chain; The contract then only verifies a cryptographic hashish of the evidence of the series, which encrypted all individuals and accounts. This "preserved transaction" model preserves the blockchain audit track by fully preserving sensitive peasant data fully and satisfied privacy rules.

4.4 Decentralized Identity (DIDs) and Verifiable Credentials
Integrating W3C-compliant DIDs allows farmers to control their own identity and credentials (e.g., land ownership, credit scores) via verifiable credentials stored off-chain in IPFS or similar. Smart contracts can then verify these credentials without centralized identity providers, reducing fraud and enabling cross-scheme interoperability.

4.5 Smart Contract Upgradability and Governance
Immutable contracts pose upgrade challenges. Using proxy patterns (e.g., OpenZeppelin Upgrades) or governance-controlled multisig wallets enables contract logic updates to patch vulnerabilities or add features. Establishing a multi-stakeholder governance model—comprising farmer cooperatives, NGO representatives, and government bodies—can democratize upgrade decisions and ensure accountability.

V. DISCUSSION

The system's end-to-end design integrates user-friendly Angular interfaces with blockchain's inherent trustworthiness, providing both usability and technological integrity. However, transitioning from a prototype into a robust production-grade system necessitates addressing several real-world deployment concerns:

- Performance Benchmarks: It is essential to rigorously simulate a realistic user base—potentially tens of thousands of simultaneous users—and measure system latency, transaction throughput, and Ethereum gas costs. This allows optimization of smart contract structure and frontend rendering performance.
- Security Audits: Engaging third-party blockchain security firms is critical to examine the smart contract layer for vulnerabilities such as reentrancy, integer overflows, access control misconfigurations, and oracle manipulation. Additionally, Angular's client-side logic must be checked for XSS, CSRF, and improper session handling.
- Regulatory Compliance: The integration of blockchain systems must align with country-specific financial and data privacy laws. For instance, India's Personal Data Protection Bill and regulations on Direct Benefit Transfers (DBT) must be considered when handling off-chain identity records or on-chain fund flows.
- User Training and Adoption: The most technically sound platform can fail without community engagement. Multilingual, voice-assisted tutorials and local language content within the Angular PWA can significantly reduce the learning curve. Community workshops and helplines can further boost adoption.
- Governance Models: Establishing clear governance protocols around the blockchain nodes—such as a consortium of agricultural departments or certified NGOs—can prevent centralization and ensure ongoing maintenance.
- Scalability Strategy: Incorporating Layer 2 solutions like zk-rollups or sidechains can alleviate mainnet congestion and cost barriers. An upgradeable proxy smart contract architecture should be employed to accommodate future changes without redeploying the entire system.

Overall, while the proposed system lays a strong foundation, this discussion emphasizes the need for rigorous validation and iterative refinement to ensure real-world impact and sustainability.

VI. FUTURE RESEARCH DIRECTIONS

- Empirical Layer-2 Evaluation:
- Pilot rollup implementations (e.g., Arbitrum, zkSync) to quantify cost savings and throughput gains.

- Privacy-Preserving Analytics: Combine zk-proofs with homomorphic encryption to allow aggregated data analysis without exposing individual records.
- Interoperability with Other Chains: Explore cross-chain bridges using Polkadot or Cosmos SDK to enable fund distribution across state lines or international donor networks.
- AI-Driven Fraud Detection: Integrate on-chain monitoring tools with machine learning models to flag anomalous disbursement patterns in real time.
- Socioeconomic Impact Assessment: Conduct longitudinal studies measuring changes in crop yields, income levels, and loan repayment rates post-adoption.

Functional requirement:

1. User registration and certification: Farmers should be able to register using a secure telephone approval system, and receive a password for a time (OTP) for verification. The certification process should ensure the privacy and integrity of the user data.
2. Farmer Dashboard: The dashboard must submit important details such as information about land, photo and contact information. Farmers should have the opportunity to update and manage their profile information through dashboards for accuracy.
3. New plan Introduction: Central government administrators should have an administrative interface to introduce new schemes. The application must be brought dynamically and is displayed to detect and implement these farmers schemes.
4. Application process: A user -friendly interface shall facilitate the application process for farmers so that they can submit the necessary details and documentation. Ensure the perfection and accuracy of the system, users should lead through the application stages.
5. Blockchain Implementation: Central Governors should be able to create and manage a blockchain series for secure and transparent transactions. The SHA-256 algorithm must be used in blockchain for safe transaction speed.
6. Verification Process: Administrators (Gram Panchayat, Taluka, State) at various government levels should reach an administrator panel to validate and approve the farmer applications. The verification process shall contain the information provided for accuracy and intensive reviews of documentation.
7. Fund transfer: When approval, the system must automatically start fund transfer from one state level to the farmer. Each fund transfer must be registered on blockchain for openness and traceability.
8. Notification

Non-functional Requirements:

1. RESULT REQUIREMENTS: Blockchain-based peasant funds. When it comes to response time, the system has been designed to respond quickly to user interactions, ensuring that each action within the platform, from application submission to funding monitor, is the maximum response in 2 seconds within time. This obligation to rapid responsibility is integrated to grow a simple and attractive user experience, where farmers and administrators can easily navigate the system with minimal delay. In addition, the system is designed for scalability, with the ability to adjust an adequate user load. In particular, it should handle a minimum of 10,000 at the same time 10,000 at the same time without experiencing a significant decline in performance. This scalability function ensures that the system remains strong and reliable, even during the period of top use, which guarantees seamless service to a large and dynamic user base. By meeting these performance requirements, blockchain bands Farmer Fund Distribution System try to create an environment where users can interact with speed and efficiency, and promote trust and satisfaction between farmers and government officials.
2. Safety requirements: Built-in safety requirements in the blockchain-based Farmers Fund's distribution system are designed to prioritize the security and integrity of user data in each stage of the application process. An essential security measure is the insurance of data integrity, where the system is carefully prepared to prevent data corruption or any kind of loss. This is crucial to maintain the accuracy and reliability of user information

CONCLUSION

Raskar *et al.*'s blockchain-based fund distribution model establishes a solid foundation for transparent and efficient subsidy delivery. By adopting Angular for robust frontend development, Layer-2 scaling for cost-effective throughput, zk-SNARKs for data privacy, DIDs for self-sovereign identity, and modular smart contract governance, the system can evolve into a resilient, user-centric platform. Further empirical evaluation, security hardening, and regulatory alignment will be essential to broaden adoption and drive sustainable agricultural development.

REFERENCES

[1] Raskar, S. M., Jadhav, A. M., Zagade, T. S., & Tarate, R. R. (2024). Blockchain-Based Farmer's Fund Distribution System. *International Journal of Emerging Technology*.

[2] Katore, A., & Choubey, S. (2021). Government Scheme and Funds Tracker using Blockchain. *IJERT*, 10(5).

[3] Jambhulkar, S. S., & Ratnaparkhi, V. P. (2020). Government Fund Distribution and Tracking System Using Blockchain Technology. *JETIR*, 7(9).

[4] Xiong, H., Dalhaus, T., Wang, P., & Huang, J. (2020). Blockchain Technology for Agriculture: Applications and Rationale. *Frontiers in Blockchain*, 3.

[5] Zheng, Z., et al. (2023). Scalability and Privacy in Blockchain Systems: A Survey. *IEEE Communications Surveys & Tutorials*.

[6] Hardjono, T., & Smith, N. (2022). Decentralized Identity Guide – W3C. https://www.w3.org/TR/did-core/

[7] Buterin, V. (2021). An EIP on Proxy Upgrade Patterns. Ethereum Improvement Proposals.

[8] Boneh, D., & Ben-Sasson, E. (2020). ZK-SNARKs: The Essentials. *Journal of Cryptographic Research*.