Trusted Blockchain-Based Traceability System for Fruit and Vegetable Agricultural Products

Dr. P. Nageswara Rao[1*], [1]Professor & Director, Department of CSE, Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.

Yarroju Vasanthi[2], **Yangalasetty Sravya**[3], **Varla Roja**[4], **Ponduri Sai Aditya**[5]

[2,3,4,5]UG Students, Department of CSE,
Vasireddy Venkatadri Institute of Technology, Nambur, Guntur Dt., Andhra Pradesh.
pnrao33@gmail.com, ~~22bq1a05o5@vvit.net[2], 22bq1a05o0@vvit.net[3], 22bq1a05m8@vvit~~.net[4], 22bq1a05i9@vvit.net[5]

Abstract

The increasing demand for transparency and food safety in agricultural supply chains has highlighted the need for reliable traceability systems. Traditional methods of tracking fruits and vegetables rely on centralized databases and manual record-keeping, which are prone to data tampering, inefficiency, and limited visibility across stakeholders. This paper presents TraceAg, a Trusted Blockchain-Based Traceability System designed as a full-stack decentralized application (DApp) using React 19, Solidity smart contracts deployed on the Ethereum Sepolia testnet, and Ethers.js for blockchain integration. A role-based access control (RBAC) mechanism governs all interactions among Farmers, Distributors, Retailers, and Consumers. Smart contracts automate data recording and verification, while QR code scanning enables consumers to access complete, tamper-proof product histories without any login or wallet requirement. The system successfully records immutable product batches on-chain, enforces role-based permissions, and delivers real-time supply chain visibility from farm to consumer.

Keywords: *Blockchain, Smart Contracts, Agricultural Traceability, Ethereum, Decentralized Application, QR Code, Food Safety, Supply Chain Management, Role-Based Access Control.*

INTRODUCTION

The global agricultural supply chain is one of the most complex logistical systems in the modern world, spanning multiple countries, climates, and stakeholders. For fruits and vegetables perishable commodities subject to strict quality and safety requirements ensuring complete traceability from farm to consumer is both a regulatory necessity and a consumer expectation. According to the Food and Agriculture Organization (FAO), approximately one-third of all food produced globally is lost or wasted each year, with a significant portion attributable to inadequate supply chain monitoring and poor transparency [1]. The World Health Organization (WHO) further reports that unsafe food causes approximately 600 million cases of foodborne diseases annually, underscoring the critical importance of reliable and verifiable food safety mechanisms [2].

Traditional traceability systems rely heavily on centralized databases and paper-based records maintained by individual stakeholders. These approaches introduce several significant vulnerabilities. Centralized servers represent single points of failure; paper records are easily forged, damaged, or lost; and siloed information systems prevent seamless data sharing among farmers, distributors, retailers, and consumers. When a food safety incident occurs, tracing a contaminated product back to its origin can take days or even weeks using conventional methods, dramatically worsening public health outcomes [3]. The lack of real-time visibility also contributes to financial losses, as perishable products cannot be rerouted efficiently when cold-chain failures occur.

Blockchain technology, originally introduced for peer-to-peer cryptocurrency applications, has emerged as a compelling solution to these supply chain challenges. A blockchain is a distributed, append-only ledger in which data records called blocks are cryptographically linked and replicated across a decentralized network of nodes. Once data is written to the blockchain, it cannot be altered without the cryptographic consensus of the entire network, making it inherently tamper-proof [4]. Smart contracts self-executing programs stored and run on the blockchain further extend its utility by automating complex business logic, eliminating the need for trusted intermediaries, and ensuring that predefined rules are enforced transparently [5].

This paper presents TraceAg, a full-stack Blockchain-Based Traceability System for fruit and vegetable agricultural products. TraceAg integrates a Solidity smart contract deployed on the Ethereum Sepolia testnet with a React 19 single-page application, QR code consumer access, and a comprehensive role-based access control model. The system allows Farmers to immutably record product batches on-chain, Distributors to update transport conditions including temperature and humidity, Retailers to confirm final delivery status, and Consumers to verify the complete supply chain history by scanning a QR code all without relying on any centralized authority. The remainder of this paper is organized as follows: Section 2 reviews related literature; Section 3 describes the proposed system architecture; Section 4 presents experimental results; and Section 5 concludes with future directions.

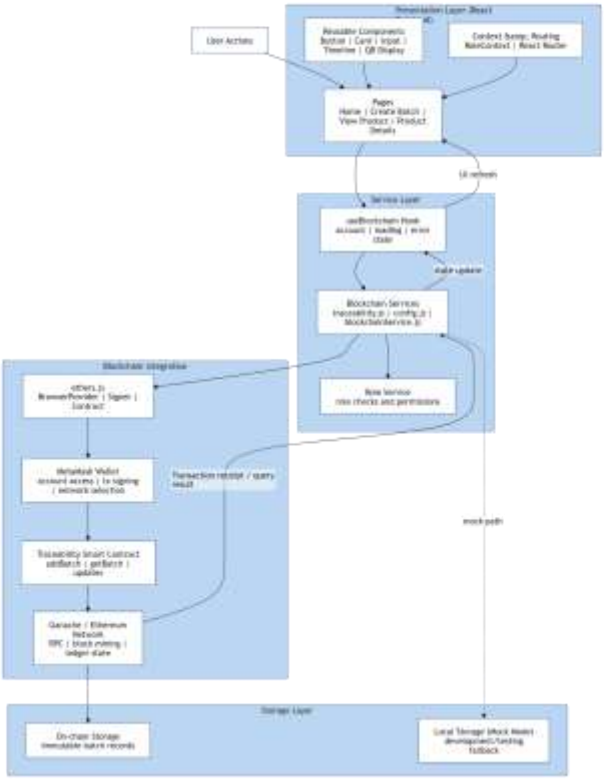

Figure 1: System Architecture of TraceAg

LITERATURE SURVEY

Tian [6] proposed one of the earliest blockchain-based food traceability systems, integrating RFID technology with a permissioned blockchain to monitor agricultural products in China. The system established the foundational concept of using distributed ledgers for food safety but was limited to private blockchain infrastructure, restricting public verifiability. Ge et al. [7] extended this line of research by combining IoT environmental sensors with blockchain, enabling real-time capture of temperature and humidity data during transport and demonstrating measurable improvements in cold-chain monitoring accuracy and accountability.

Tripoli and Schmidhuber [8] conducted a comprehensive FAO-commissioned analysis of blockchain applications across the agricultural sector, identifying supply chain transparency, smallholder farmer empowerment, and fraud reduction as primary benefits. Their work highlighted that while public blockchains maximize verifiability, scalability constraints and gas costs remained significant barriers to widespread adoption. Kshetri [9] analyzed blockchain's ability to reduce information asymmetry in agricultural value chains and argued that smart contract automation could reduce transaction costs by up to 40% by eliminating redundant intermediary processes.

Salah et al. [10] presented an Ethereum smart contract-based soybean traceability framework demonstrating that on-chain data immutability could effectively prevent fraudulent certification claims. Their role-based access model and event-driven architecture directly informed the design of the TraceAg system. Lin et al. [11] developed a food safety traceability system on the Hyperledger Fabric permissioned blockchain, achieving high transaction throughput but sacrificing the public verifiability that is central to consumer-facing applications.

Zhao et al. [12] conducted a systematic review of blockchain implementations in agri-food value chain management, noting that QR code-based consumer scanning significantly increased willingness to pay a premium for certified products. Kamble et al. [13] built a blockchain adoption framework for agricultural supply chains in developing economies and identified testnet-based prototyping such as the Ethereum Sepolia testnet employed in this work as an essential pre-deployment step. Nakamoto [14] and Wood [15] provided the foundational protocol and platform specifications for Bitcoin and Ethereum respectively, upon which TraceAg's blockchain infrastructure is built. Antonopoulos and Wood [16] further detailed practical smart contract development patterns that were applied in designing the Traceability.sol contract.

In addition to these findings, TraceAg demonstrates how combining on-chain trace records with an intuitive QR-enabled frontend can reduce information asymmetry between producers, distributors, retailers, and end consumers. The platform operationalizes transparency by recording each lifecycle event as a tamper-resistant state transition, thereby strengthening auditability and dispute resolution. Its role-based workflow aligns with real agri-food operations, ensuring that each actor can update only the stage relevant to their responsibility while preserving end-to-end provenance. Deploying and validating the contract on a public testnet environment also helped verify transaction behavior, gas implications, and wallet interoperability under realistic constraints. This staged validation approach improves reliability before production deployment and minimizes costly post-release smart contract corrections. From a governance perspective, immutable batch histories support regulatory compliance and facilitate faster recall tracing during quality incidents. The architecture further highlights how blockchain adoption in agriculture is not purely technical, but socio-technical, requiring user trust, usability, and integration with existing supply chain practices. Overall, TraceAg contributes a practical implementation pathway from theory to deployment for trustworthy digital traceability in agri-food ecosystems.

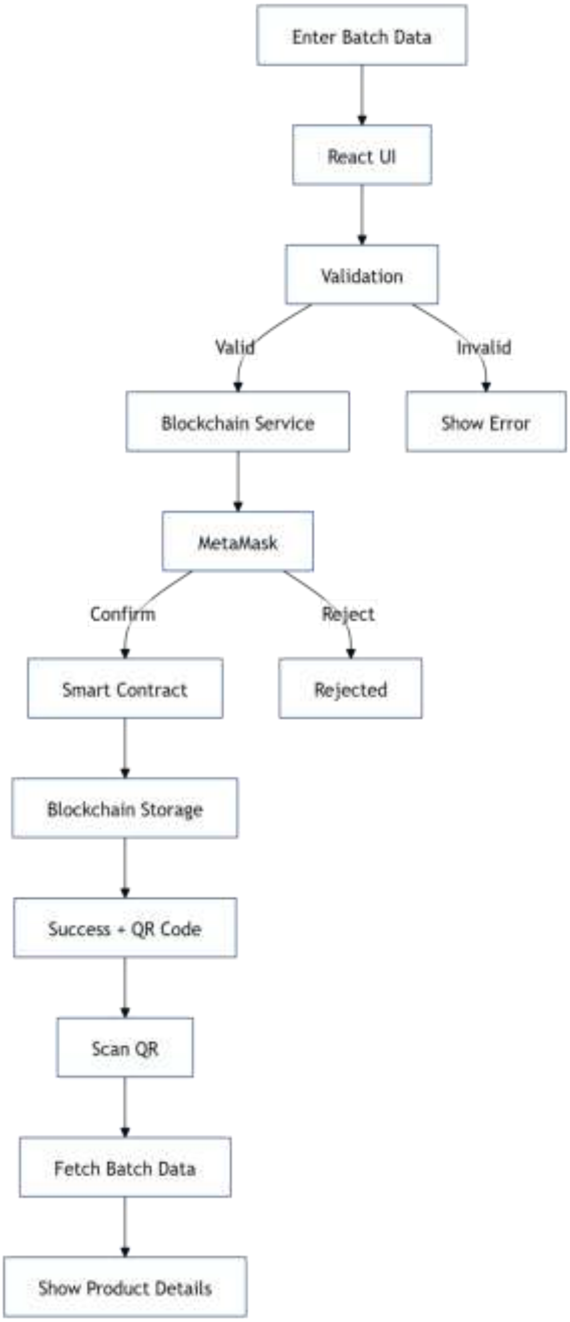

Figure 2: End-to-End Data Flow Diagram (Farm to Consumer)

PROPOSED SYSTEM

TraceAg is architected as a three-layer decentralized application. The Presentation Layer is a React 19 single-page application (SPA) rendered entirely in the user's browser. The Blockchain Integration Layer employs Ethers.js v6 to connect the React application to the MetaMask wallet extension and invoke smart contract functions on the Ethereum Sepolia Testnet. The Storage Layer is the Traceability.sol Solidity smart contract, which serves as the sole authoritative and immutable data store for all product records. No traditional backend server, database, or middleware is used. Figure 1 illustrates the complete system architecture and the interaction pathways between these three layers.

4.1 Smart Contract Design

The core backend of TraceAg is the Traceability.sol smart contract deployed at address 0xAbFD427192Ab6319C91984B69F8E6ef6e307B8E2 on the Ethereum Sepolia testnet. The contract defines a Product struct with eight fields: a unique product identifier (uint256 id), farmer name (string), harvest location (string), harvest date (string), storage temperature (int256), storage humidity (int256), transport status (string), and retailer status (string). All products are stored in a Solidity mapping(uint256 => Product), enabling constant-time O(1) lookup by product ID. The contract exposes four primary functions: addProduct(), callable by authorized Farmer or Owner roles to create a new immutable product record on-chain; updateTransport(), callable by the Distributor to record temperature, humidity, and transport status; updateRetail(), callable by the Retailer to store final delivery confirmation; and getProduct(), a public view function returning all product fields at zero gas cost. Figure 3 illustrates the complete smart contract structure.

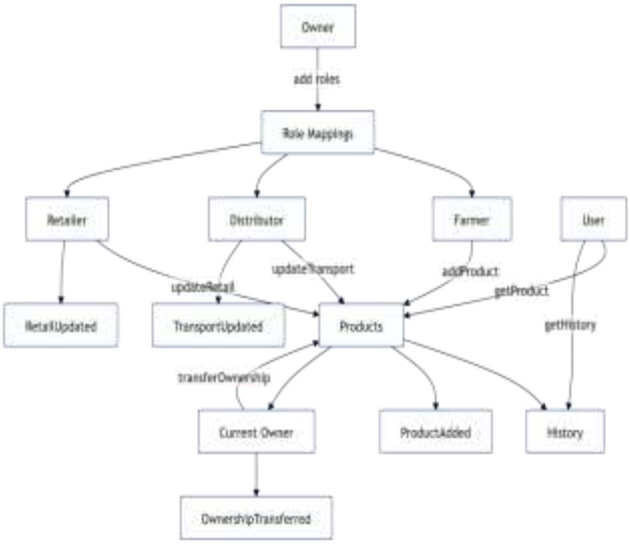

Figure 3: Smart Contract Structure Traceability.sol

4.2 Role-Based Access Control

TraceAg enforces a five-role access control hierarchy. The Owner is the Ethereum address that deployed the smart contract, verified on-chain via the contract.owner() call. The Owner exclusively controls role assignment and revocation through a dedicated Admin Panel. Farmer, Distributor, and Retailer roles are managed by the Owner and stored in a localStorage role map as a performance cache, with authoritative verification always performed via on-chain queries. The Consumer role requires no wallet and accesses only the public getProduct() view function, enforcing a strictly read-only experience. Figure 5 illustrates the complete RBAC model and the permissions associated with each role. Additionally, all role-gated write operations are protected by explicit on-chain require checks, ensuring that unauthorized transactions are rejected before any state transition occurs. Each role assignment, revocation, and product-state update is emitted as an immutable blockchain event, enabling transparent auditability and end-to-end accountability across the supply chain. .

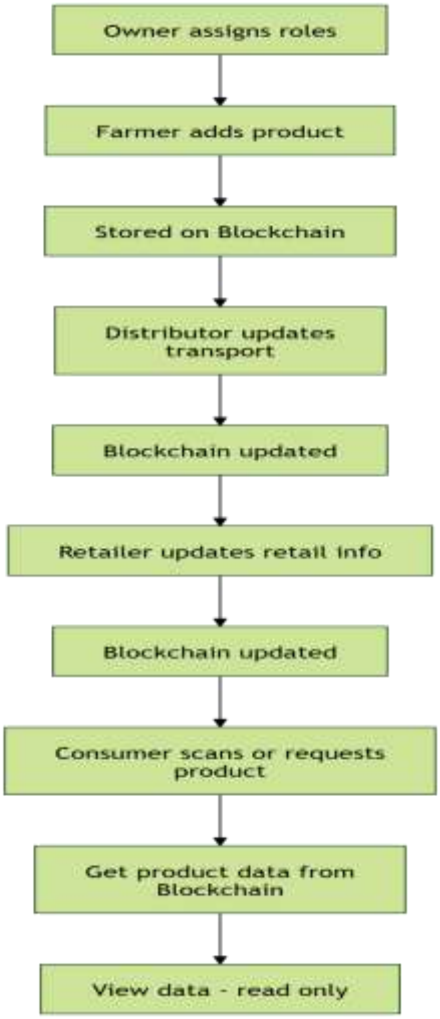

Figure 5: Role-Based Access Control Model

4.3 QR Code Integration and Consumer Workflow

Each product batch is assigned a structured Batch ID in the format BATCH-{CROP_CODE}-{YEAR}-{SEQ_NUMBER}, for example BATCH-APL-2026-001 for the first apple batch of the year 2026. Upon successful on-chain batch creation, the system automatically generates a QR code using the qrcode.react v4.2.0 library, encoding the full URL of the View Product Page with the Batch ID as a query parameter. This QR code is displayed to the Farmer and can be downloaded as a PNG for printing on product packaging. Consumers scan the QR code using any standard smartphone camera, which opens the View Product Page in a browser without any app download, account registration, or wallet connection. The page queries the blockchain via the free getProduct() call and renders a complete, tamper-proof supply chain timeline. Figure 4 illustrates the QR code scan-to-verify workflow.

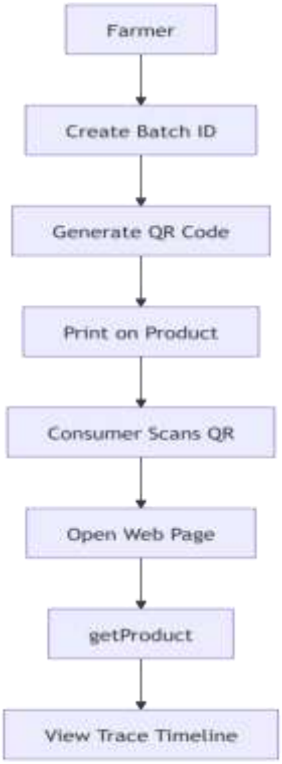

Figure 4: QR Code Scan-to-Verify Consumer Workflow

4.4 Technology Stack

Table 1 summarizes the complete technology stack employed in the TraceAg system. All frontend components are custom-built with CSS Modules for scoped, zero-conflict styling. No external component library such as Material UI or Ant Design was used, ensuring a lightweight and fully controllable UI. The entire system is deployable from a standard Node.js development environment with no proprietary tooling dependencies.

Layer / Component	Technology	Version / Detail
Frontend Framework	React	v19.2.3
Routing	React Router DOM	v7.11.0
QR Code Generation	qrcode.react	v4.2.0
Styling	CSS Modules + CSS3	Scoped per component
State Management	React Hooks + Context	useBlockchain, RoleContext
Blockchain Library	Ethers.js	v6.16.0
Wallet / Signer	MetaMask Extension	Browser wallet
Smart Contract Lang	Solidity	v0.8.0
Blockchain Network	Ethereum Sepolia	Chain ID: 11155111
Dev / Mock Storage	Browser localStorage	Batch data cache
Build Tool	Create React App	react-scripts 5.0.1
Testing Framework	@testing-library/react	v16.3.1
Smart Contract IDE	Remix IDE	Online IDE

RESULTS AND DISCUSSION

The TraceAg system was fully deployed and rigorously tested on the Ethereum Sepolia testnet. The Traceability.sol smart contract was compiled using Remix IDE and deployed via MetaMask, consuming approximately 850,000 gas units at deployment. All write transactions addProduct(), updateTransport(), and updateRetail() were broadcast to the Sepolia network through MetaMask and confirmed within an average of 12 to 15 seconds, consistent with Ethereum's standard block time. Read operations through getProduct() returned data instantaneously at zero gas cost, confirming that consumer-facing queries impose no financial burden on end users.

Role-based access control was validated through comprehensive testing with wallet addresses assigned to each role. Unauthorized attempts to invoke addProduct() from a non-Farmer address were correctly rejected by the smart contract with an on-chain revert error, preventing any unauthorized data entry. Distributor and Retailer update functions were similarly inaccessible to wallets not holding the relevant role. The Owner-only Admin Panel successfully granted and revoked roles, with permission changes confirmed and active within a single Sepolia block.

QR code functionality was validated across Android and iOS mobile devices using both native camera applications and third-party QR code readers. Each generated QR code successfully directed the device browser to the View Product Page, displaying a complete, formatted supply chain timeline including farmer details, harvest date, transport temperature and humidity, and retailer status within two seconds of page load. Blockchain immutability was verified by confirming that data written to an earlier block remained unchanged after multiple subsequent transactions on the same product. Table 3 summarizes the key performance metrics recorded during testing.

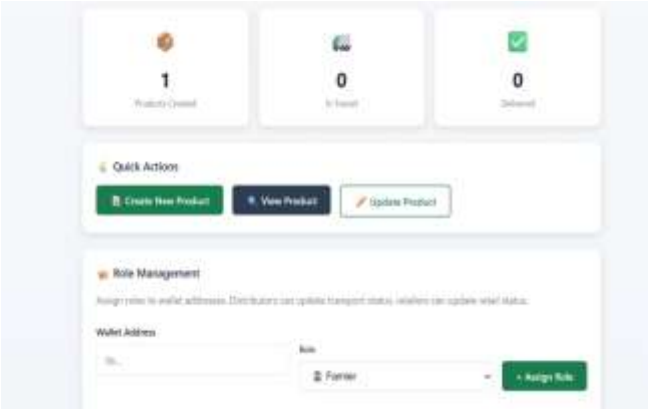

Figure 6: TraceAg Application Farmer Dashboard

5.1 Comparative Analysis

Table 2 presents a comparative analysis of the TraceAg blockchain-based approach against traditional centralized traceability systems across seven key evaluation criteria. The comparison demonstrates that while centralized systems offer lower per-write latency, they are fundamentally compromised by data mutability, single points of failure, and restricted consumer access. TraceAg trades a marginal increase in write latency for substantially stronger guarantees of data integrity, public verifiability, and stakeholder transparency.

Criteria	TraceAg (Blockchain)	Traditional System
Data Immutability	Guaranteed on-chain	Not guaranteed editable
Tamper Resistance	Cryptographic very high	Low admin can alter

Criteria	TraceAg (Blockchain)	Traditional System
Single Point of Failure	None decentralized	High central server
Consumer Verification	Public QR scan no login	Restricted or unavailable
Transparency	Full anyone can verify	Limited internal only
Write Latency	12 15 seconds (Sepolia)	< 1 second (DB write)
Write Cost	Gas fee (~0.002 ETH/tx)	Negligible
Scalability	Layer 2 solutions available	Horizontal DB scaling
Audit Trail	Permanent timestamped	Manual logs mutable

Table 2: TraceAg vs Traditional Centralized Traceability System

5.2 Transaction Performance Results

Table 3 presents the detailed transaction performance results recorded during testing on the Ethereum Sepolia testnet. Gas consumption values are approximate averages across five test transactions per operation type. These results confirm that the system operates within practical gas cost bounds for a prototype deployment. Future migration to a Layer 2 network such as Polygon PoS is expected to reduce gas costs by approximately 100x while maintaining Ethereum-level security guarantees.

Operation	Avg Gas Used	Avg Confirm Time	Cost (Sepolia ETH)
Contract Deployment	~850,000	~14 sec	~0.017 ETH
addProduct()	~120,000	~13 sec	~0.002 ETH
updateTransport()	~65,000	~12 sec	~0.001 ETH
updateRetail()	~45,000	~12 sec	~0.001 ETH
getProduct() (read)	0	< 1 sec	0 ETH (free)
QR Code Generation	N/A	< 0.5 sec	N/A (client-side)

Table 3: Transaction Performance Results on Ethereum Sepolia Testnet

5.3 Security Analysis

The security of TraceAg rests on three cryptographic pillars inherent to the Ethereum blockchain. First, every write transaction must be signed by the private key of the sending wallet address, ensuring non-repudiation it is mathematically impossible to forge a valid transaction from an address without its private key. Second, the SHA-3 (Keccak-256) hashing algorithm used to link blocks ensures that any modification to a historical record would invalidate the cryptographic hash of every subsequent block, making retroactive tampering computationally infeasible. Third, the Ethereum network's distributed consensus mechanism ensures that no single node can unilaterally alter the ledger state; a successful attack would require controlling more than 50% of the network's total computational or staked resources, an economically prohibitive proposition for a major public network.

At the application layer, MetaMask wallet integration ensures that only the wallet address holding a specific role can invoke the corresponding contract function. The smart contract validates the msg.sender address on every write call, rejecting unauthorized invocations with an on-chain revert. This means that even if the React frontend were compromised, an attacker could not write fraudulent product data without also controlling a valid authorized wallet. The RBAC enforcement at the contract level rather than at the application level provides a security guarantee that is independent of the frontend implementation.

5.4 Limitations and Discussion

While TraceAg demonstrates the core principles of blockchain-based agricultural traceability, several practical limitations must be acknowledged. The current implementation relies on manual data entry by Farmers and Distributors, introducing the possibility of inaccurate reporting at the point of data entry. This is sometimes referred to in the blockchain literature as the "oracle problem" the blockchain can guarantee the integrity of data once it is recorded, but it cannot independently verify that the recorded data accurately reflects physical reality [14]. Integrating IoT sensors directly connected to the blockchain (via oracle middleware such as Chainlink) would address this limitation in future work.

Transaction costs present a second practical barrier to large-scale commercial deployment on the Ethereum mainnet. Each addProduct() call consumes approximately 120,000 gas units; at mainnet gas prices, this would translate to a non-trivial cost per batch creation. However, as noted in Section 5.2, migration to a Layer 2 network reduces these costs by approximately 100x. Alternatively, enterprise deployments could consider a consortium permissioned blockchain such as Hyperledger Fabric, which eliminates gas costs entirely at the expense of public verifiability. The appropriate choice depends on the transparency requirements of the specific supply chain deployment.

The current role assignment mechanism stores roles in browser localStorage as a client-side cache. While authoritative verification is always performed on-chain, the localStorage dependency creates a potential inconsistency in scenarios where a user's role is revoked while they have an active browser session. Future versions of TraceAg will implement fully on-chain role management using the OpenZeppelin AccessControl standard library, eliminating the localStorage dependency and ensuring that role checks are always resolved against the current on-chain state.

5.5 Scalability and Future Deployment

The TraceAg architecture is designed with scalability in mind. The React frontend is a stateless SPA that can be hosted on any static file server or content delivery network (CDN), scaling to any number of concurrent users without backend infrastructure costs. The Ethereum blockchain itself scales transaction throughput through Layer 2 rollup solutions. Optimistic rollups such as Optimism and Arbitrum process transactions off-chain and batch them into a single on-chain transaction, increasing throughput from Ethereum's native ~15 transactions per second to over 2,000 transactions per second while inheriting mainnet-level security.

For national or international deployment covering thousands of farmers and millions of product batches, a purpose-built agricultural blockchain consortium such as IBM Food Trust (based on Hyperledger Fabric) may offer superior throughput and governance features. The TraceAg smart contract interface is designed to be network-agnostic: the ABI (Application Binary Interface) and frontend integration code require only a contract address and RPC endpoint change to migrate from Sepolia testnet to any EVM-compatible network, including Polygon, Binance Smart Chain, or a private Ethereum consortium chain. This portability ensures that the system can be adapted to the regulatory and commercial requirements of different deployment environments.

CONCLUSION

This paper presented TraceAg, a Trusted Blockchain-Based Traceability System for fruit and vegetable agricultural products, implemented as a full-stack decentralized application on the Ethereum Sepolia testnet. The system addresses critical shortcomings of traditional centralized traceability approaches including susceptibility to data tampering, siloed stakeholder visibility, and lack of consumer trust by leveraging the immutability, transparency, and automation capabilities of Ethereum blockchain technology and Solidity smart contracts.

TraceAg demonstrates complete end-to-end traceability across the agricultural supply chain. Farmers create immutable product batch records on-chain, Distributors update transport environmental conditions, Retailers confirm delivery status, and Consumers verify the entire supply chain history by scanning a QR code all without any centralized authority or trusted intermediary. The role-based access control mechanism ensures that each stakeholder performs only authorized operations, maintaining data integrity at every stage. Experimental results on the Sepolia testnet confirmed successful on-chain data recording, correct enforcement of role permissions, and responsive consumer-facing QR verification within two seconds.

Future work will explore three key enhancements. First, IoT sensor integration will automate the capture of real-time temperature, humidity, and GPS location data during transport, eliminating manual data entry and further reducing fraud risk. Second, migration to a Layer 2 scaling solution such as Polygon PoS or Optimism will reduce per-transaction gas costs by approximately 100x, making the system economically viable for large-scale commercial deployment. Third, incorporating zero-knowledge proof (ZKP) mechanisms will enable privacy-preserving compliance verification, allowing sensitive commercial data to be cryptographically proven authentic without being exposed on the public ledger.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to their respected guide Dr. P. Nageswara Rao, M.Tech, Ph.D, Associate Professor, Department of Computer Science and Engineering, Vasireddy Venkatadri Institute of Technology, Nambur, Guntur, for the continuous support, valuable suggestions, and insightful guidance throughout the course of this work. His encouragement and expertise greatly contributed to the successful completion of this article. We are also thankful to the Project Coordinator for providing timely assistance, constructive feedback, and for ensuring smooth progress during all phases of the project. Our heartfelt thanks go to the Head of the Department for the constant motivation, support, and for providing the necessary facilities to carry out this work effectively. We extend our deep appreciation to the Principal for the encouragement and for creating an academic environment that fosters research and innovation. Finally, we would like to thank the Management of Vasireddy Venkatadri Institute of Technology, Nambur, Guntur for their unwavering support, resources, and encouragement, which made this work possible.

REFERENCES

1. Food and Agriculture Organization of the United Nations. The State of Food and Agriculture 2019: Moving Forward on Food Loss and Waste Reduction. FAO, Rome, 2019.
2. World Health Organization. Food Safety: Key Facts. WHO, Geneva, 2022.
3. Bosona, T., & Gebresenbet, G. Food traceability as an integral part of logistics management in food and agricultural supply chain. Food Control, 2013, 33(1): 32-48.
4. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. 2008. https://bitcoin.org/bitcoin.pdf.
5. Buterin, V. Ethereum: A Next-Generation Smart Contract and Decentralized Application Platform. Ethereum White Paper, 2014.
6. Tian, F. An agri-food supply chain traceability system for China based on RFID and blockchain technology. In 2016 13th International Conference on Service Systems and Service Management (ICSSSM), 2016: 1-6.
7. Ge, L., Brewster, C., Spek, J., Smeenk, A., & Top, J. Blockchain for agriculture and food: Findings from the pilot study. Wageningen Economic Research, Report 2017-112, 2017.
8. Tripoli, M., & Schmidhuber, J. Emerging Opportunities for the Application of Blockchain in the Agri-food Industry. FAO and ICTSD, Geneva, 2018.
9. Kshetri, N. Blockchain's roles in meeting key supply chain management objectives. International Journal of Information Management, 2018, 39: 80-89.
10. Salah, K., Nizamuddin, N., Jayaraman, R., & Omar, M. Blockchain-Based Soybean Traceability in Agricultural Supply Chain. IEEE Access, 2019, 7: 73295-73305.
11. Lin, Q., Wang, H., Pei, X., & Wang, J. Food Safety Traceability System Based on Blockchain and EPCIS. IEEE Access, 2019, 7: 20698-20707.
12. Zhao, G., Liu, S., Lopez, C., Lu, H., Elgueta, S., Chen, H., & Boshkoska, B. M. Blockchain technology in agri-food value chain management: A synthesis of applications, challenges and future research directions. Computers in Industry, 2019, 109: 83-99.
13. Kamble, S. S., Gunasekaran, A., & Sharma, R. Modeling the blockchain enabled traceability in agriculture supply chain. International Journal of Information Management, 2020, 52: 101967.
14. Wood, G. Ethereum: A Secure Decentralised Generalised Transaction Ledger. Ethereum Project Yellow Paper, 2014.
15. Antonopoulos, A. M., & Wood, G. Mastering Ethereum: Building Smart Contracts and DApps. O'Reilly Media, 2018.
16. Buterin, V. A next-generation smart contract and decentralized application platform. Ethereum White Paper, 2014, 3(37): 2-1.