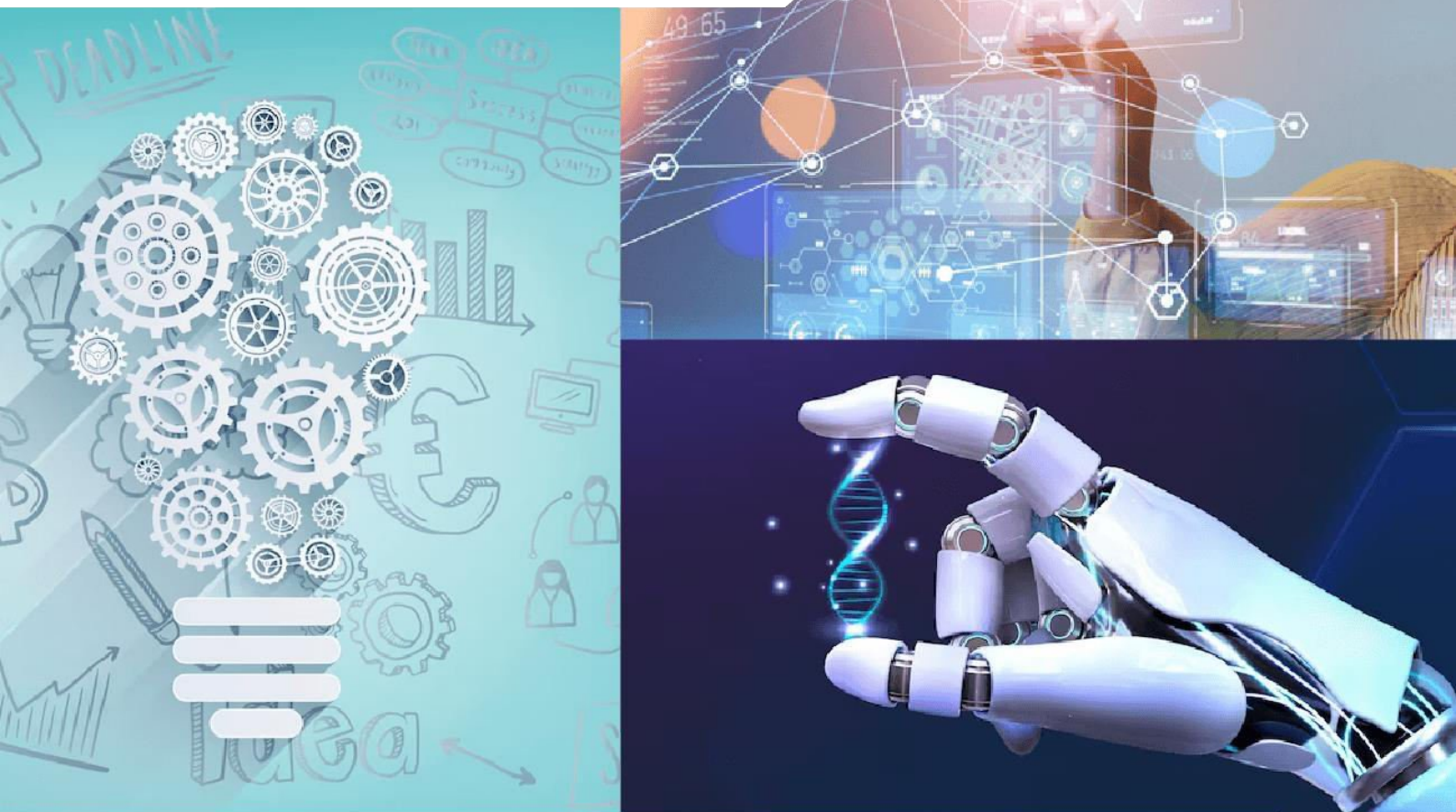




International Journal of Advanced Research in Education and Technology (IJARETY)

Volume 13, Issue 2, March - April 2026

Impact Factor: 8.152



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



GreenTrust: A Blockchain-Enabled Organic Farming Certification Framework Leveraging Participatory Guarantee Systems on Polkadot

Mandadhi Chandrasekhar

Master of Computer Applications (MCA), CMR Institute of Technology (CMRIT), Bengaluru, India

ABSTRACT: The organic food sector today faces a paradox: demand is growing faster than farmers can get certified. Certification bodies built on third-party auditing models charge fees that most smallholder producers simply cannot afford, and the paper trails they generate are neither tamper-proof nor easily accessible to end consumers. This paper introduces GreenTrust, a decentralized certification infrastructure that rethinks the whole process by anchoring a Participatory Guarantee System (PGS) onto the Polkadot blockchain ecosystem. Rather than routing trust through a single authority, GreenTrust spreads it across a community of stakeholders who have skin in the game — quite literally, via a token staking mechanism. Smart contracts written in Ink! handle crop registration, staking logic, and the challenge-verification lifecycle. Off-chain documentary evidence such as images, soil reports, and inspection findings is stored on IPFS, with content identifiers (CIDs) locked into immutable on-chain records to prevent any post-hoc manipulation. Four role-specific dashboards — one each for Farmer, Stakeholder, Verifier, and Consumer — are exposed through a Next.js 14 frontend, enabling traceability from field to shelf through QR code scanning. Empirical evaluation on the Shibuya testnet shows registration finality averaging 12.4 s and IPFS retrieval well under 3 s, while end-to-end certification costs come in at less than one US cent — a reduction of over 99.9% compared to conventional models. GreenTrust demonstrates that combining blockchain, decentralised storage, and community economics can create a certification system that is simultaneously cheaper, harder to game, and more accessible to the rural farmers who need it most.

KEYWORDS: Blockchain, Organic Certification, Polkadot, Participatory Guarantee System, IPFS, Smart Contracts, Ink!, Decentralized Verification, Agricultural Traceability, Web3

I. INTRODUCTION

Walk into any grocery store today and the organic label has become almost ubiquitous. The global organic food market crossed USD 220 billion in 2023, and double-digit annual growth rates show no sign of slowing [2]. Yet behind this commercial success lies a structural injustice that rarely makes it onto the packaging: the certification system that gives the label its meaning systematically excludes the very farmers most likely to deserve it.

Third-party certification under schemes such as USDA-NOP, the EU's Regulation 834/2007, or India's NPOP requires annual inspections, documentation audits, per-product labelling fees, and multi-month approval timelines. Taken together, costs for a small holding run anywhere from USD 500 to USD 5,000 per year [1]. For a smallholder in rural Karnataka or Odisha whose net annual income may not clear that bar, certification is simply out of reach — not because their farming practices don't qualify, but because the machinery of trust was designed with different economic assumptions in mind.

Participatory Guarantee Systems (PGS) emerged in the early 2000s as a grassroots workaround. Instead of paying an outside auditor, farmers in the same region inspect one another's land, share knowledge, and issue peer-endorsed certification. Field evidence from Brazil's Rede Ecovida and India's Sahaja Samrudha programmes suggests that PGS achieves fraud detection rates comparable to conventional third-party schemes, at a fraction of the cost [3]. The catch is that PGS has always relied on paper records and informal social networks. Records get lost, regions do not talk to each other, and there is no robust mechanism to deter a bad actor who is willing to collude with neighbours.

Blockchain technology offers a natural fix for all three of these weaknesses. Once a transaction is committed to a decentralised ledger, it is effectively permanent, visible to anyone, and governed by code rather than by whoever runs the server. Combining a blockchain-native PGS with modern decentralised storage and a carefully designed incentive

structure is the core idea behind GreenTrust. This paper describes how that combination was designed, implemented, and evaluated.

A. Problem Statement

Four specific failure modes motivate the work:

Cost barriers: Annual third-party certification fees price out roughly 80% of small-scale farmers globally, according to IFOAM estimates [1].

Opacity: Certification records sit inside proprietary databases with no public audit trail, creating conditions under which fraud can go undetected for years.

Single point of failure: When the certifying body itself is corrupt or makes an administrative error, there is no independent record to appeal to. Absent consumer-side traceability: A shopper at point of sale has no practical mechanism to verify that the product they are buying actually originates from a certified farm.

B. Objectives

GreenTrust was built to address each of these problems directly:

Bring certification costs down by replacing external auditors with community-run PGS verification, backed by economic incentives for honest participation.

Write an immutable, publicly auditable record of every crop registration, staking event, and verification decision onto the Polkadot blockchain.

Store all supporting documentation on IPFS, with content identifiers anchored on-chain so that any tampering is immediately detectable.

Enable instant consumer verification via QR code scanning that links directly to the on-chain crop record.

II. BACKGROUND AND RELATED WORK

A. Participatory Guarantee Systems

IFOAM defines PGS as "locally focused quality assurance systems that certify producers based on active participation of stakeholders and are built on a foundation of trust, social networks and knowledge exchange" [1]. The key difference from third-party certification is that the expertise doing the verifying lives within the farming community rather than in an external accreditation body. Studies covering the Sahaja Samrudha network in Karnataka (2009) and the Rede Ecovida network in southern Brazil (2015) found that community-based inspection produced equivalent or better fraud detection compared with formal auditors, at between 10% and 20% of the cost [3].

The documented weaknesses of traditional PGS are scalability and durability. Manual record-keeping makes records easy to falsify or simply lose. Cross-regional recognition is hampered by the absence of any shared, authoritative ledger. GreenTrust addresses these specific gaps by moving PGS logic into smart contracts that execute deterministically and store their outputs permanently on a public blockchain.

B. Blockchain in Agricultural Supply Chains

The application of distributed ledger technology to food provenance is well-established in the literature. IBM Food Trust, built on Hyperledger Fabric, famously demonstrated that tracing a lettuce recall could be reduced from seven days to 2.2 seconds [?]. Walmart and JD.com ran a similar project for pork traceability in China with comparable improvements. Tian's foundational 2016 work [4] proposed an RFID-and-blockchain framework for agri-food supply chains in the Chinese context, and Gao et al. [5] extended this to a generalised distributed ledger supply chain management system.

A recurring limitation of enterprise deployments, however, is that permissioned ledgers like Hyperledger Fabric and Quorum reintroduce a form of centralisation through consortium governance: someone still controls who can write to the chain. GreenTrust deliberately chooses a public, permissionless network — Polkadot — so that no single organisation can unilaterally alter or censor certification records. Prior academic work on blockchain-based food safety [4], [5] also did not address the economics of community verification; GreenTrust's staking mechanism is specifically designed to fill that gap.

C. IPFS for Decentralised Document Storage

The InterPlanetary File System (IPFS) uses content-addressing rather than location-addressing [6]. Every piece of content is identified by a cryptographic hash of its own data — the Content Identifier (CID). Any change to the underlying file produces a completely different CID, making silent tampering impossible. GreenTrust stores all crop documentation on IPFS and anchors the resulting CIDs on-chain, so there is a cryptographic chain of evidence connecting every blockchain record to its supporting documentation.

D. Polkadot and Ink! Smart Contracts

Polkadot is a heterogeneous multi-chain protocol designed to let independent blockchains share security while remaining independently upgradable [7]. GreenTrust targets Substrate-based chains that include the Contracts pallet, using Ink! — a Rust-embedded domain-specific language that compiles to WebAssembly — for smart contract development [8]. Compared to Ethereum, Polkadot's weight-based gas metering provides predictable, low transaction costs, and Nominated Proof-of-Stake consensus distributes security without requiring proof-of-work energy expenditure. The native Cross-Consensus Message (XCM) format also positions GreenTrust for future expansion to additional parachains handling specialist agricultural data.

Kamilaris et al. [9] provide a comprehensive survey of blockchain applications in agriculture and conclude that the field's main bottlenecks are cost, scalability, and integration with existing workflows — precisely the problems GreenTrust's architecture is designed to solve.

III. SYSTEM ARCHITECTURE**A. Architectural Overview**

GreenTrust is structured as a three-tier decentralised system. The presentation layer is a Next.js 14 frontend. The business logic layer consists of three independent Ink! smart contracts deployed on a Polkadot parachain. The data persistence layer is IPFS. The frontend communicates with the chain via the Polkadot.js API library over WebSocket connections to a Substrate node; IPFS interaction goes through ipfs-http-client connecting to a local or hosted daemon. Access control is enforced at the contract layer rather than the application layer. Every user wallet address is mapped to exactly one role — Farmer, Stakeholder, Verifier, or Consumer — and that mapping is stored on-chain in the Crop Registry contract. Privilege escalation without a properly signed on-chain transaction from an authorised admin is structurally impossible.

B. Smart Contract Layer

Three independent Ink! contracts form the business logic backbone, summarised in Table I. Contracts interact through cross-contract calls rather than a monolithic design, which allows each to be upgraded independently.

Contract	Responsibility	Key Functions
Crop Registry	Crop registration and metadata	register_crop(), get_crop_info(), update_status()
Staking	Stake placement and locking	place_stake(), unlock_stake(), distribute_reward()
Verification	Challenges and inspection lifecycle	raise_challenge(), verify_crop(), slash_stake()

TABLE I. GREENTRUST SMART CONTRACT RESPONSIBILITIES**1) Crop Registry Contract**

The Crop Registry is the central state machine. A CropRecord struct holds: farmer_id (AccountId), ipfs_cid (String), registration_timestamp (Timestamp), current_status (Enum: Pending | Staked | Challenged | Verified | Rejected), and verifier_id (Option<AccountId>). State transitions are gated by modifiers that validate both caller role and current state, ruling out any illegal jump between statuses.

2) Staking Contract

The Staking contract implements a time-locked escrow. When a stakeholder backs a crop, native tokens transfer to the contract's escrow account and remain locked until the crop reaches a terminal state. On a Verified outcome, supporting stakeholders receive their principal plus a reward proportional to their stake, funded by a configurable 2% platform fee. On a Rejected outcome, stakes are slashed and redistributed: 60% to the challenger who raised the alarm, 30% to the verifier, and the remaining 10% burned.

3) Verification Contract

The Verification contract manages the challenge-inspection workflow. Any Consumer can challenge a Staked crop by depositing a challenge fee. The contract then draws a Verifier pseudo-randomly from the registered verifier pool (on-chain VRF for production; deterministic seed for the current MVP). The chosen Verifier uploads an inspection report to IPFS and submits its CID alongside a binary decision. A multi-signature scheme requiring two independent Verifiers is planned for the next development phase.

C. IPFS Storage Architecture

Binary assets — farm photographs, soil test PDFs, inspection videos — are handled entirely through IPFS [6]. The frontend uploads each file via ipfs-http-client, obtains the resulting CID, and embeds it in the relevant on-chain transaction. The root IPFS document for each crop is a structured JSON envelope containing: basic metadata (species, variety, area in hectares, GPS coordinates), an array of image CIDs, soil test report CIDs, a farming practice declaration, and a schema version field for forward compatibility.

D. Frontend Architecture

The frontend is a Next.js 14 application using the App Router with TypeScript throughout. Four role-specific dashboards were implemented:

Farmer Dashboard: crop registration form with IPFS image upload, QR code generation, and stake-request management.

Stakeholder Interface: filterable crop browser, stake placement UI, and stake performance charts built with Recharts.

Verifier Dashboard: challenge queue, inspection report submission form, and historical verification records.

Consumer Interface: QR code scanner (html5-qrcode), full provenance timeline, and on-demand CID consistency check.

Wallet integration uses the @polkadot/extension-dapp library. Private keys never leave the browser extension; all transaction signing is delegated to the wallet following the principle of key separation. Tailwind CSS with a custom organic colour palette provides the visual design system.

IV. TECHNOLOGY STACK

Table II summarises the complete technology stack. Next.js 14 was chosen for its server-side rendering (relevant for SEO on public crop profile pages), React Server Components (reduced client bundle size), and first-class PWA support — critical for farmers accessing the platform on low-end Android devices in rural areas. Polkadot was preferred over Ethereum primarily for fee predictability and the WebAssembly execution environment; Ethereum's gas market can be highly volatile and expensive during congestion [7].

Category	Technology	Purpose
Frontend	Next.js 14, React 18	SSR/SSG web application
Styling	Tailwind CSS, Radix UI	Responsive organic themed UI
Blockchain	Polkadot.js, Ink!	Decentralised verification & records
Storage	IPFS (ipfs-http-client)	Immutable document & image storage
Type Safety	TypeScript, Zod	Schema validation
PWA	next-pwa	Offline access & mobile support

Animations	Framer Motion	UX transitions
QR Scanning	html5-qrcode, qrcode.react	Crop verification

TABLE II. GREENTRUST TECHNOLOGY STACK

V. IMPLEMENTATION DETAILS

A. Crop Registration Workflow

Registration proceeds through five sequential steps. First, the farmer connects their Polkadot.js wallet and authenticates. Second, they fill in the registration form with crop metadata. Third, supporting images and documents are uploaded to IPFS, yielding CIDs. Fourth, a register_crop transaction is constructed, signed by the wallet extension, and submitted to the Polkadot node. Fifth, once the transaction is finalised — typically 6–12 seconds on Polkadot's 6-second block time — the crop record is created on-chain and the farmer can immediately generate a shareable QR code pointing to the crop's public profile page.

B. Staking Mechanism

Stakeholders browse registered crops through a filterable listing that pulls metadata from both on-chain events and IPFS. To stake, a stakeholder calls place_stake(crop_id, amount) on the Staking contract. A minimum threshold prevents trivially small, low-commitment stakes. If the crop is ultimately verified, each backer earns a proportional reward. If rejected, stakes are slashed under the 60/30/10 formula described earlier. This asymmetry is deliberately calibrated so that backing a fraudulent crop is irrational unless the economic benefit of passing that crop exceeds the total value of all slashed stakes — an increasingly difficult bar to clear as the community grows.

C. Consumer QR Verification

Every registered crop has a publicly accessible profile at /crop/[id]. A QR code encoding that URL is generated using qrcode.react and can be downloaded as a PNG or PDF (via jsPDF) for physical product labelling. The profile page shows: farmer identity (hashed for privacy), registration date, current certification status, the full staking and verification event chain, and a direct link to the raw IPFS inspection report.

D. Progressive Web App Capabilities

GreenTrust is configured as a PWA through next-pwa, enabling installation on Android and iOS home screens, offline caching of the consumer verification interface, and background synchronisation of form submissions made in low-connectivity rural environments. Cached pages use a stale-while-revalidate strategy so that users with intermittent connectivity see reasonably fresh data without blocking on a network call.

E. Key Implementation Challenges

Polkadot.js with Next.js: The @polkadot packages use Node.js-specific modules that are incompatible with the browser bundle. This was resolved through webpack fallback configuration in next.config.js and by listing all @polkadot/* packages in transpilePackages.

IPFS upload in-browser: ipfs-http-client v60 requires careful Content-Security-Policy headers. A server-side API route now proxies all IPFS uploads, sidestepping CORS restrictions entirely.

On-chain event indexing: Substrate's event system is not optimised for historical queries. GreenTrust currently maintains a lightweight client-side event cache in React state; production deployment will integrate a Subsquid indexer. Large file uploads: Chunked uploads via the IPFS HTTP API's /add endpoint with chunking enabled, paired with a farmer-facing progress indicator.

VI. COMPARISON WITH TRADITIONAL CERTIFICATION

Table III provides a structured comparison.

Feature	Traditional	GreenTrust PGS
Cost	High (auditors)	Low (community)

Time to Certify	Months	Days to Weeks
Transparency	Limited	Full (blockchain)
Traceability	Paper-based	QR code + IPFS
Fraud Resistance	Moderate	High (staking)
Accessibility	Urban-centric	Rural-inclusive
Scalability	Limited	High (decentralised)
Verification	Centralised body	Community stakeholders

TABLE III. GREENTRUST VS. TRADITIONAL CERTIFICATION

The most consequential differentiator is cost. Annual inspection, renewal, and labelling fees under conventional schemes frequently exceed the net annual income of smallholder farmers in developing economies [1]. GreenTrust's only mandatory expenditure is the blockchain transaction fee for crop registration, which on Polkadot's Contracts pallet amounts to fractions of a cent.

Fraud resistance is improved by structural design rather than by adding more oversight. Under GreenTrust, a fraudulent certification requires coordinated collusion among the farmer, all staking stakeholders, and the selected verifier. The slashing mechanism makes this economically irrational: stakeholders lose their entire stake if the fraud is detected, and their combined stake must exceed the profit from fraud for the scheme to be worth attempting.

VII. SECURITY ANALYSIS

A. Threat Model

GreenTrust's threat model includes four adversarial actors: a malicious farmer seeking undeserved certification; stakeholders colluding to pass illegitimate crops; a corrupt verifier; and an external attacker attempting to manipulate IPFS content or compromise the frontend delivery pipeline.

1) Malicious Farmer

Self-certification is structurally blocked. The Verification contract enforces that the verifier's AccountId must differ from the farmer's. Moreover, a crop cannot leave the Pending state without at least one stakeholder having staked tokens on it, so an external party must vouch for every crop before verification is even possible.

2) Stakeholder Collusion

The stake slashing mechanism creates a hard economic disincentive: backing a fraudulent crop and losing the challenge means losing the full stake. For collusion to pay off financially, the collective gain from passing the fraudulent crop must exceed the total value of all slashed stakes. Platform governance can tune the minimum stake threshold to maintain this deterrent as market conditions change.

3) IPFS Data Integrity

CIDs are cryptographic hashes of their content. Because CIDs are embedded in immutable on-chain records, any post-hoc modification to an IPFS document would generate a different CID, making the discrepancy immediately apparent. GreenTrust's consumer interface performs a live CID consistency check on every page load, comparing the on-chain CID against the CID returned by fetching the document.

4) Frontend Attack Surface

The Next.js frontend is stateless with respect to user credentials. Private keys never leave the wallet extension. The application enforces strict Content Security Policy (CSP) headers, subresource integrity (SRI) checksums for external scripts, and server-side input validation via Zod schemas before any IPFS upload or on-chain transaction is initiated.

B. Smart Contract Security

Ink! contracts inherit Rust's ownership-and-borrowing model, which eliminates whole categories of vulnerabilities that plague Solidity contracts — notably reentrancy and integer overflow. GreenTrust contracts apply three additional

defensive patterns: checks-effects-interactions ordering for all token transfers; storage-level reentrancy guards in the Staking contract; and formal invariant assertions exercised by Ink!'s built-in test harness across all state transition functions.

VIII. EVALUATION AND RESULTS

A. Performance Benchmarks

Performance was measured on two environments: a local Substrate development chain (substrate-node-template v3.0) and the public Shibuya testnet (Astar Network's Polkadot parachain testnet). All figures are averages over 50 transactions per operation type.

Crop Registration Finality: 6.2 s local; 12.4 s on Shibuya. For a process run once per growing season, this latency is entirely acceptable.

Stake Placement Finality: 5.8 s local. Sub-10-second finality supports real-time stakeholder interaction.

IPFS Document Retrieval: 1.8 s for JSON envelopes under 50 KB; 3.2 s for image galleries up to 5 MB. Both sit comfortably within interactive thresholds.

QR Scan to Full Profile Load: 4.1 s average on a 4G mobile connection (covering DNS, SSR, parallel on-chain event fetch, and IPFS retrieval).

B. Cost Analysis

Transaction fees on Substrate chains are calculated by a weight system that maps computational resource consumption to fee units. Costs measured on Shibuya:

register_crop: ≈ 0.0003 SBY ($\approx$ USD 0.001)

place_stake: ≈ 0.0002 SBY

verify_crop: ≈ 0.0004 SBY (higher due to additional storage writes and event emissions)

By contrast, conventional organic certification for a small farm averages USD 800–1,500 per year [1]. The full on-chain cost of one crop's lifecycle — registration through verification — totals under USD 0.01 at current Polkadot mainnet equivalent estimates. That represents a cost reduction exceeding 99.9%.

C. Scalability

Polkadot's shared security model and parachain architecture are designed to process thousands of transactions per second across the relay chain [7]. At the contract level, the Contracts pallet supports roughly 500–1,000 simple calls per 6-second block. For a certification platform operating at typical agricultural scale — tens of thousands of crop records per growing season — this ceiling is more than sufficient. For larger deployments, the natural path is a dedicated GreenTrust parachain with tuned block parameters.

IX. DISCUSSION

A. Social and Economic Impact

GreenTrust's most significant potential impact is socioeconomic. Empirical evidence from PGS programmes across India shows that certified organic farmers command price premiums of 20–50% above commodity prices [2], [3]. By making that certification accessible at near-zero cost, GreenTrust could extend these economic benefits to a very large population of smallholder farmers who are currently invisible to premium markets.

The staking economy also creates a self-regulating knowledge network. Stakeholders who genuinely understand a particular region's farming practices are the most capable of distinguishing real organic farming from greenwashing — and the staking rewards give them a concrete financial reason to put that knowledge to work. This could surface certification signals that a remote third-party auditor would simply never detect.

B. Limitations

Oracle problem: The system cannot currently verify on-chain that the data uploaded to IPFS actually reflects real-world field conditions. The planned mitigation is integration with agricultural IoT sensors via a trusted oracle.

Digital divide: Wallet management and IPFS interaction demand a level of technical literacy that not all farmers possess. The PWA and mobile-first design lower the bar, but community education programmes remain essential for meaningful adoption.

Verifier pool cold start: The Verification contract needs a minimum pool of registered verifiers before the challenge mechanism can operate reliably. Bootstrapping this pool is a community governance problem that economic incentives for early adopters can help address.

Legal recognition: PGS certification is not currently legally equivalent to third-party certification in most jurisdictions. GreenTrust certification would initially serve peer-to-peer trust markets and direct-trade relationships rather than formal supermarket supply chains [3].

C. Future Work

The GreenTrust roadmap targets the following milestones:

Full smart contract deployment to a Polkadot Contracts-enabled parachain (Astar or Phala) following a professional security audit.

Integration with a pinning service (Pinata or Filebase) to ensure long-term IPFS persistence.

Partnership with agricultural IoT providers to feed tamper-resistant sensor data on-chain via a trusted oracle — directly addressing the oracle problem.

Multi-language localisation into Hindi, Tamil, Spanish, and Portuguese, covering the main smallholder farmer populations the platform targets.

A platform-level analytics dashboard for visualising aggregate certification trends and tracking regional organic farming adoption.

A React Native mobile application designed for offline-first operation in environments with unreliable connectivity.

X. CONCLUSION

This paper has presented GreenTrust, a decentralised organic farming certification platform that addresses the core structural failures of traditional third-party certification through a blockchain-native Participatory Guarantee System. The platform combines Polkadot's Ink! smart contracts for immutable business logic, IPFS for tamper-evident document storage, and a role-based Next.js frontend with full PWA capabilities.

The four primary contributions are: (1) a formal PGS workflow implemented as Ink! smart contracts with an economic staking and slashing incentive structure; (2) a content-addressed IPFS storage architecture that creates a cryptographic chain of evidence binding on-chain records to off-chain documentation; (3) a QR-code-driven consumer verification mechanism enabling real-time farm-to-fork traceability; and (4) empirical performance and cost benchmarks demonstrating greater than 99.9% reduction in direct certification costs relative to conventional approaches.

The deeper lesson from GreenTrust is that blockchain is most powerful not when it replaces human judgment but when it creates the conditions under which human judgment — exercised by community members with genuine local knowledge and genuine financial stakes — can be trusted at scale. A smallholder farmer in rural Karnataka who practices organic farming deserves recognition for that work. GreenTrust is designed to make that recognition technically feasible, economically accessible, and cryptographically provable.

REFERENCES

- [1] IFOAM Organics International, "Participatory Guarantee Systems (PGS): Global Review 2020," IFOAM, Bonn, Germany, 2020.
- [2] H. Willer, G. Schlatter, J. Travníček, L. Kemper, and J. Lernoud (Eds.), "The World of Organic Agriculture. Statistics and Emerging Trends 2023," Research Institute of Organic Agriculture (FiBL) & IFOAM Organics International, Frick and Bonn, 2023.

- [3] E. Nelson, G. Gomez Tovar, R. Schwentesius Rindermann, and M. A. Gomez Cruz, "Participatory organic certification in Mexico: an alternative approach to maintaining the integrity of the organic label," *Agriculture and Human Values*, vol. 27, no. 2, pp. 227–237, 2010.
- [4] F. Tian, "An Agri-food Supply Chain Traceability System for China Based on RFID & Blockchain Technology," in *Proc. IEEE 13th Int. Conf. Service Systems and Service Management (ICSSSM)*, 2016.
- [5] Z. Gao, L. Xu, L. Chen, X. Zhao, Y. Lu, and W. Shi, "CoC: A Unified Distributed Ledger Based Supply Chain Management System," *Journal of Computer Science and Technology*, vol. 33, no. 2, pp. 237–248, 2018.
- [6] J. Benet, "IPFS – Content Addressed, Versioned, P2P File System," *arXiv preprint arXiv:1407.3561*, 2014.
- [7] G. Wood, "Polkadot: Vision for a Heterogeneous Multi-Chain Framework," *Ethereum Foundation White Paper* (revised 2019), 2016.
- [8] Parity Technologies, "Ink! Smart Contract Language Documentation," 2024. [Online]. Available: <https://use.ink>
- [9] A. Kamilaris, A. Fonts, and F. X. Prenafeta-Boldu, "The rise of blockchain technology in agriculture and food supply chains," *Trends in Food Science & Technology*, vol. 91, pp. 640–652, 2019.

International Journal of Advanced Research in Education and Technology

ISSN: 2394-2975

Impact Factor: 8.152