iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** XI **Month of publication:** November 2026

DOI: https://doi.org/10.22214/ijraset.2026.85507

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue XI Nov 2026- Available at www.ijraset.com

CottonTrace for Seed to Cloth Traceability with IoT and Digital Product Passports

Dr Puja Gupta[1], Palkesh Prajapati[2], Darshil Vatyani[3], Ayush Sharma[4]
Department of Information Technology, Shri Govindram Seksaria Institute of Technology and Science, Indore, Madhya Pradesh, India

Abstract: ***Cotton production involves repeated changes in material form, ownership and quantity, making a garment difficult to trace using disconnected records. We present CottonTrace, a web-based prototype that links eight material stages through quantity-labelled lot genealogy, contract-based validation, an application-managed permissioned-ledger model, off-chain evidence, Internet of Things ingestion and a QR-accessible Digital Product Passport. An explainable detection layer combines robust statistics with domain rules to route suspicious records to auditors. The design is examined against the implementation and the project-reported synthetic evaluation of 18 lots, 61 blocks and 2,488 sensor readings. That evaluation records rejection of invalid operations, detection of edited evidence, eight findings corresponding to planted irregularities, and verification of 12 upstream lots for a garment passport. These observations demonstrate functional integration within the pilot. They do not establish distributed consensus, field accuracy, calibrated fraud probabilities or a complete environmental life-cycle assessment. The main contribution is a transparent workflow connecting deterministic validation, evidence integrity, material attribution and human review in a single cotton traceability prototype.***

Index Terms: ***Cotton supply chain, traceability, permissioned ledger, Internet of Things, explainable anomaly detection, Digital Product Passport.***

I. INTRODUCTION

A finished cotton garment may combine material from several fields and processing facilities. Seed cotton becomes lint, fibre, yarn, fabric and finished garments; shipments can also divide a batch between recipients. A list of successive owners cannot by itself represent these transformations. Traceability needs to preserve which inputs contributed to each output, how much was consumed, and which organisation asserted each event.

Blockchain-based textile traceability has already been explored through sector-specific networks and transaction rules [1]. CottonTrace builds on this problem setting through an integrated teaching and research prototype. Its purpose is to make material history inspectable while keeping the distinction between a consistent digital record and a truthful physical claim explicit. A stored hash can expose an altered statement, but a correctly hashed statement may still contain a false measurement.

A. Problem Definition and Research Objectives

The traceability problem has three distinct dimensions. The first is identity continuity: a supplier must refer to the same material as the next processor even when packaging, ownership or lot codes change. The second is material continuity: a finished batch must retain links to the input quantities consumed during transformation, including mixtures and partial shipments. The third is evidence continuity: events, certificates and measurements must remain connected to the correct material and to the party that recorded them. Losing any one of these relations weakens a downstream provenance claim.

CottonTrace treats traceability as a graph of recorded material relationships with an associated history of assertions. Its operational objective is to help a participant answer what material is available, what it came from, where it went and which supporting records exist. Its audit objective is to distinguish prohibited operations, altered evidence and suspicious patterns. Its communication objective is to convert an upstream graph into a product story that a consumer can inspect without operating the supplier console.

The research scope is software integration and scenario-based verification. It does not include changing cotton cultivation methods, establishing a physical fibre-identification technique, certifying a supplier or determining legal compliance. The prototype can record a declared organic attribute and an auditor event, but it does not perform laboratory tests for organic composition. Keeping those boundaries explicit allows the same architecture to be evaluated without treating a digital assertion as a physical measurement.

©IJRASET: All Rights are Reserved

B. Research Questions and Contributions

RQ1 concerns representation: can the system preserve a useful ancestry graph when several harvests are blended and only part of an intermediate lot moves downstream? RQ2 concerns control: which errors should cause immediate rejection, which changes should produce integrity violations, and which admissible records should enter an auditor queue? RQ3 concerns interpretation: what does a batch passport actually verify, and what assumptions underlie its resource indicators? These questions are assessed through implementation analysis and the existing project evaluation rather than a population-level field experiment.

The resulting contribution is an explicit connection between the operational inventory model and three forms of checking. Contract rules constrain actions before commitment. Hashes make selected stored assertions and file contents comparable with retained references. Statistical and rule-based findings retain explanations for human review. The system also demonstrates how quantity-labelled ancestry can allocate a subset of recorded resource use to a retail batch. These contributions are integrated engineering outcomes and should be judged against those objectives.

II. RELATED WORK

A. Textile Traceability and Event Models

Agrawal et al. propose a blockchain framework for textile and clothing traceability, including network configuration, smart contracts and an organic-cotton example [1]. CottonTrace focuses on combining lot genealogy, sensor evidence, anomaly review and consumer presentation in one inspectable prototype.

GS1 EPCIS 2.0 provides a standard for exchanging visibility events across enterprises. Its TransformationEvent models inputs that are consumed and outputs that are produced [2]. This is conceptually relevant to cotton blending and conversion. CottonTrace currently uses its own lot and event schema; an EPCIS export or conformance assessment is future integration work.

B. Permissioned Ledgers and Evidence Integrity

Hyperledger Fabric separates execution, ordering and validation within a permissioned distributed architecture [3]. CottonTrace adopts selected concepts, including organisation identifiers, contract modules and hash-linked blocks, but executes them in one application. Organisation labels and recorded endorser names are metadata in this prototype, rather than independently verified peer signatures. SHA-256 supplies the digest primitive used for evidence checks [4].

C. Anomaly Detection and Product Passports

Anomaly detection identifies departures from expected behaviour and depends on the assumptions made about normal data [5]. Median-based scale estimates offer an interpretable way to reduce sensitivity to extreme observations [6]. CottonTrace combines such estimates with explicit rules and preserves the evidence behind each finding. This supports auditor triage rather than automatic declarations of fraud.

The Ecodesign for Sustainable Products Regulation establishes a Digital Product Passport framework, with product requirements specified through applicable delegated acts [7]. The CottonTrace passport demonstrates batch identification, provenance display and selected resource indicators. No regulatory conformance assessment is claimed.

D. Supply Chain Objectives and Ledger Selection

Kshetri examines how blockchain may support supply-chain objectives including quality, dependability, risk reduction and sustainability, with particular attention to device integration and identity [8]. That perspective motivates a system in which a transaction is connected to a known organisation and to supporting measurements. CottonTrace evaluates the mechanics of those connections. It does not measure changes in cost, delivery reliability, supplier trust or commercial outcomes.

Wüst and Gervais distinguish blockchain designs from centrally managed databases according to writers, trust and the availability of a trusted intermediary [9]. Their decision framework is relevant because the CottonTrace operator currently controls the application and database. The prototype therefore demonstrates a ledger-shaped audit mechanism within a central service. Independent agreement among mutually distrustful organisations would require a different deployment and governance arrangement.

This distinction influences the interpretation of the term permissioned. In CottonTrace it refers to authenticated application participation and organisation-specific contract checks. It does not imply that every organisation operates a peer, verifies another participant’s signature or independently stores the authoritative ledger. The distinction also prevents a misleading performance comparison between the local prototype and a distributed system with network communication and endorsement costs.

E. Sensor Integration and Alternative Analytics

Atzori, Iera and Morabito describe the Internet of Things as the integration of identification, communication, sensing and intelligent processing technologies [10]. For CottonTrace, the useful link is between an identified device, an organisation and a lot or field. The prototype implements that link digitally and supplies a reading simulator. Establishing that a deployed sensor is attached to the correct machine or field remains an operational responsibility.

MQTT defines a publish-subscribe messaging protocol that can support decoupled device communication [11]. CottonTrace stores MQTT-style topic names but ingests data through an HTTP endpoint. An external bridge could translate broker messages into authenticated HTTP submissions. Such a bridge is an integration option; broker connectivity, delivery guarantees and device reconnect behaviour were not tested in the reported pilot.

Isolation Forest isolates observations through random partitioning and uses isolation behaviour to identify anomalies [12]. It is a possible comparison method once a suitable operational dataset exists. CottonTrace instead uses interpretable univariate deviations and domain rules. The choice provides straightforward evidence for an auditor, but it cannot be interpreted as superior detection accuracy without a common labelled evaluation.

F. Environmental Accounting and the Research Gap

The GHG Protocol Product Life Cycle Accounting and Reporting Standard addresses emissions accounting over a defined product life cycle [13]. It provides a useful methodological reference for explaining why an energy-and-transport sum is narrower than a full product footprint. CottonTrace does not implement that standard, and the reported carbon value is retained as an application indicator with a restricted calculation boundary.

The specific engineering gap addressed here is the connection between transformation-aware records, admissibility checks, retained evidence and product-level presentation in a deployable teaching prototype. The paper does not claim that previous systems lack these capabilities in general. Its narrower contribution is to make the interactions and assumptions visible: a valid document digest can accompany a certificate attached to the wrong lot, and a contract-valid quantity can still be unusual for a processing stage.

III. SYSTEM DESIGN

A. Actors and Architecture

The operational chain comprises farms, a gin, a spinning mill, a weaving or knitting unit, a dyeing unit, a garment factory and a brand. Auditor and administrator roles support review and network operation. Users belong to organisations, and protected server actions derive the acting organisation from an authenticated session. Contract functions then check the permissions and quantities relevant to the requested operation.

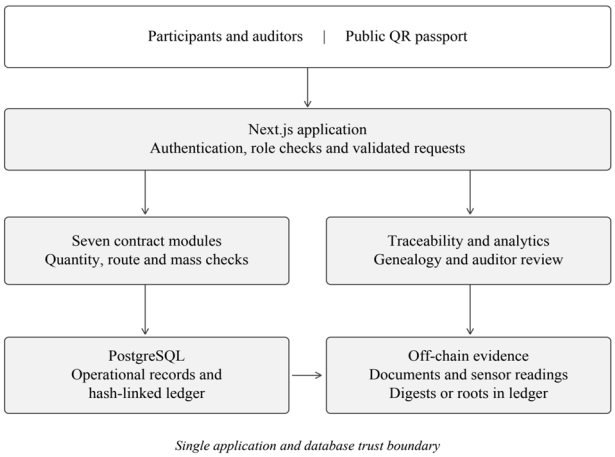


Fig. 1. CottonTrace architecture. Ledger and operational records share a database; evidence files remain off-chain.

The application uses Next.js, React and TypeScript, with Prisma and PostgreSQL for persistence. Local operation supports an embedded PostgreSQL environment, while the documented cloud configuration uses Neon and Vercel Blob. These are implementation choices, not requirements of the traceability model. Figure 1 shows the principal data paths and the shared trust boundary.

B. Lot Genealogy

Eight lot types represent seed, harvest, bale, fibre, yarn, fabric, dyed fabric and garment batches. Each lot stores its identifier, original quantity, available quantity, unit, owner, creator, attributes and creation transaction. A directed parent-to-child link records the quantity drawn from a parent. Multiple parents represent mixing; multiple children represent splitting or repeated consumption.

A transfer initially reserves stock and creates a pending handoff. Only the recipient can accept or reject it, while cancellation belongs to the sender. Acceptance of a partial quantity creates a linked child lot for the recipient. Rejection or cancellation restores the reserved quantity. Transformation consumes available parent quantities and records the resulting output. These operations preserve the relation between inventory and provenance.

Seed-to-harvest links represent agricultural ancestry. They are not interpreted as conservation of seed mass into harvested cotton. Physical yield limits apply to configured processing stages. Upstream traversal uses a visited set and a maximum depth of 20; this is sufficient for the illustrated chain but is not an unrestricted genealogy engine.

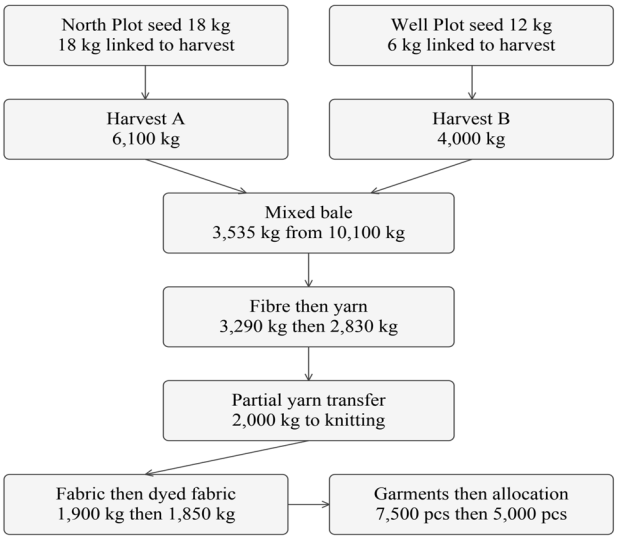


Fig. 2. Recorded quantities in the primary pilot genealogy. Seed links denote ancestry; processing quantities denote consumed material.

C. Contract Responsibilities

TABLE I
Contract modules and principal checks

Module	Principal responsibility
Lot	Authorised type, owned inputs, quantity and yield
Transfer	Valid route, available stock and response actor
Event	Current ownership and permitted event type
Certification	Auditor or administrator authority
Document	Attachment authority and digest format
Sensor	Registered device and organisation authority
Passport	Owned garment batch and authorised publisher

Table I summarises seven contract modules. The common invocation boundary validates inputs, applies state changes and appends a block inside a database transaction. A domain-rule rejection rolls back the operational mutation and creates a separate rejection record. Rejections are not committed into valid blocks; consequently their audit trail does not inherit block-level integrity protection.

D. Functional Requirements and Operating Assumptions

The first requirement is a persistent identity for every material lot and an explicit relation for every recorded transformation or split. The second is inventory reservation: a pending handoff must reduce the quantity available for another action. The third is actor-sensitive validation: creating a bale, accepting a shipment, certifying a lot and publishing a passport are different operations with different prerequisites. The fourth is evidence attachment, including a reference that can be checked after a stored file changes.

A fifth requirement is reviewability. Rejected requests need intelligible reasons; anomaly findings need reproducible evidence; and verification needs to identify which check failed. A sixth is selective presentation: a consumer should reach a coherent batch history while an operator can examine individual records. These requirements explain why the system contains both a transactional console and a public passport instead of relying on a QR label alone.

The prototype assumes that its application service is available, its database executes the requested transactions, and organisation types supplied by the membership workflow represent the intended demonstration roles. It also assumes that lot creation times can be used to order ancestry propagation. These are operating assumptions of the implementation, not properties guaranteed by cryptographic hashing. The limitations section considers what changes when those assumptions fail.

E. Data Model and Separation of Responsibilities

The relational schema separates the current operational view from the event and evidence history. Lot stores the material identity, its original and available quantities, its creator and its current owner. LotLink stores a directed consumption relation. Transfer stores the shipment state and the lot produced by an accepted partial handoff. TraceEvent stores the descriptive event timeline. Block and LedgerTx record committed payloads independently of the mutable operational fields.

Organisation and Field supply ownership context, area and location. Device and SensorReading connect measurements to a device, with optional lot and field associations. Document stores a digest and storage reference rather than embedding file bytes in a block. Anomaly records the finding fingerprint, score, evidence and review status. Passport binds a public identifier to a garment lot and its presentation metadata. Authentication records remain in separate user, account, session and verification models.

Keeping original quantity distinct from available quantity is particularly useful. A 2,830 kg yarn lot can retain its original production quantity while 2,000 kg is reserved for one shipment and later consumed into fabric. Reducing the original quantity instead would change the denominator used by ancestry attribution and would obscure how the lot was divided. A transformation creates a new lot identity; it does not overwrite the type and quantity of its parent.

TABLE II
Principal entities and identifying relations

Entity	Key relation
Lot	Owner, creator and creation transaction
LotLink	Parent, child and quantity drawn
Transfer	Sender, recipient and result lot
TraceEvent	Lot, organisation and transaction
Document	Stored file, digest and anchor
SensorReading	Device, metric, lot and timestamp
Anomaly	Fingerprint, evidence and reviewer
Passport	Public identifier and garment lot

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

F. End to End Operational Workflow

A farm first records seed and field information, then sowing, irrigation and other permitted cultivation events. A harvest lot refers to the seed quantity linked to that harvest. The farmer initiates a transfer to an allowed gin, which accepts custody before consuming the material into a bale. The spinning mill receives bales and creates fibre and yarn lots. Knitting converts a received yarn allocation into fabric; dyeing records finishing attributes; and the garment factory records the number and weight of finished pieces.

The brand receives a garment allocation and may publish a passport for the owned garment lot. Along this chain, an auditor may add certification events or supporting documents. A sensor device may be associated with a lot or field and submit readings. The anomaly scan operates across the resulting records, while the public passport loads the target's ancestry and recomputes the implemented integrity checks. Each screen is a different view of shared entities rather than an independently maintained copy of the supply chain.

This workflow also clarifies the difference between custody and processing. A whole-lot transfer changes ownership without creating a new material type. A partial accepted transfer creates a same-type child because the receiving allocation needs its own identity. A transformation creates an output of a permitted different type and consumes its declared inputs. The distinction is essential for preserving both shipment history and the material genealogy.

G. Implementation and Deployment Choices

The reviewed project configuration uses Next.js 16, React 19, TypeScript and Prisma 7. Server actions perform session resolution and request validation before invoking domain functions. Better Auth supplies email-and-password sessions, with role and organisation fields excluded from the ordinary sign-up input. Zod provides request schemas. A single application boundary makes the prototype easier to run and inspect, while coupling service availability and trust to the central operator.

The local configuration supports an embedded PostgreSQL server and disk-based file storage. The documented cloud configuration uses Vercel application hosting, Neon PostgreSQL and Vercel Blob. Three.js through React Three Fiber supplies the garment visualisation, Leaflet supplies maps, and charts display readings and findings. These interfaces share the same underlying records. The paper treats the configuration as the reviewed implementation and does not claim that particular library versions or cloud settings remain current deployment recommendations.

IV. METHODS

A. Quantity and Processing Constraints

For a processing output with mass Mout, the input mass is the sum of quantities drawn from its parents after unit conversion. Kilogram-based lots use their recorded mass. A garment batch is converted from pieces using its recorded per-piece weight. A configured yield limit umax permits a transformation only when

$$M_{out} \leq u_{max} \sum M_{in} + \varepsilon, \ \varepsilon = 10^{-6} \quad (1)$$

The implementation rounds inventory remainders to three decimal places. Expected yields guide anomaly detection, while hard limits reject an invocation. Table III lists the configured values. They are prototype parameters requiring process-specific validation before use at an actual facility. The dyeing limit allows small mass increases associated with processing additions.

TABLE III
Configured processing yields

Output stage	Expected	Hard limit
Bale	35%	45%
Fibre	93%	100%
Yarn	86%	98%
Fabric	95%	100%
Dyed fabric	97%	103%
Garment mass	80%	95%

1) Validation Order and Inventory Semantics

Lot creation validates the output definition and creating organisation before reading inputs. It requires a positive finite output quantity, a nonempty name and the configured required attributes. When a field is supplied, that field must belong to the actor's organisation. Each input is then checked for existence, ownership, active status, a permitted source type, a positive draw quantity and sufficient availability. The same input identifier cannot appear twice in one request.

For stages with an upper yield limit, output mass is checked against the sum of the declared input masses. Only after validation does the invocation create the lot and its parent links, reduce input availability and write the processing event. When the rounded remainder reaches the implementation tolerance, a consumed input is marked accordingly. This ordering prevents a failed business-rule check from leaving a partially created transformation in the database transaction.

A yield ceiling represents an admissibility rule, not a universal physical law. The dyeing ceiling exceeds one because the configured process allows limited mass additions. A farm's seed-to-harvest relation has no such processing ceiling because a crop grows through biological inputs outside the seed lot's recorded mass. The distinction should be retained if the model is adapted to another commodity or to recycled textile material.

The available-quantity check is evaluated inside a transaction, but atomicity alone is not a proof of correct behaviour under every concurrent inventory race. The source includes retries for unique-key conflicts when block numbers collide. It does not supply a dedicated stress test establishing serialisable stock behaviour across concurrent transforms and handoffs. Concurrency therefore remains a separate evaluation requirement rather than an inferred result of the pilot.

B. Ledger Construction and Verification

Transaction arguments use a deterministic JSON representation with sorted object keys and a nonce. Let c, f, a, m and t denote the contract, function, canonical argument string, submitting organisation and timestamp. With H denoting SHA-256 and C the canonical representation, the transaction digest is

$$h_{tx} = \mathrm{H}(\mathrm{C}(\mathrm{c}, \mathrm{f}, \mathrm{a}, \mathrm{m}, \mathrm{t})) \quad (2)$$

A block stores its number n, previous hash hprev, transaction root r, timestamp t and channel ch. Its header digest is

$$h_b = \mathrm{H}(\mathrm{C}(\mathrm{n}, h_{prev}, \mathrm{r}, \mathrm{t}, \mathrm{ch})) \quad (3)$$

The Merkle procedure hashes pairs of hexadecimal digest strings, duplicating the final leaf when a level has odd length. A normal invocation appends one transaction, so the usual block root equals that transaction digest. Verification recomputes transaction hashes, Merkle roots, block hashes and predecessor links. Per-lot verification additionally compares the lot code, type and quantity with the referenced transaction.

The ledger detects inconsistent edits relative to retained hashes within a single administrative trust boundary. Per-lot comparisons cover the lot code, type and quantity; other mutable attributes, ownership and genealogy links are outside that comparison. Section VII examines the security implications.

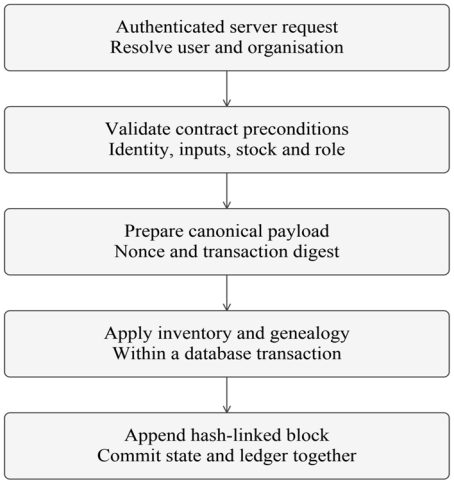


Fig. 3. Normal contract execution path. Domain-rule rejection rolls back the transaction and is logged separately.

1) Canonical Payloads and Block Contents

Canonicalisation recursively sorts object keys, preserves array order, omits undefined object properties and converts dates to ISO strings. This means changing the order in which a client constructed object properties does not change the encoded logical object. Array order remains significant. A nonce generated for a normal invocation distinguishes otherwise identical submissions. The exact serialization convention matters because a different encoder could produce a different digest for equivalent application data.

The transaction hash includes a canonical outer object whose argument field is itself the canonical argument string. The normal code path then applies the state mutation using that digest as a reference. It creates a block with the next number, a predecessor hash and a Merkle root over the committed transaction identifiers. The block timestamp is adjusted if necessary so that it does not precede the previous block. The genesis block uses a zero predecessor and records the configured channel membership.

The block hash covers the number, predecessor, root, timestamp and channel. The stored endorser list is not included in that hash construction, and the list is not a collection of independent signatures. Likewise, the outer transaction digest covers the fields selected by the hashing function; auxiliary database metadata should not be assumed to be protected merely because it is stored beside that digest. Verification scope follows the encoded fields.

2) Merkle Construction and Verification Procedure

At each Merkle level, adjacent hexadecimal digest strings are concatenated and hashed. If the level contains an unmatched final digest, it is duplicated. An empty list returns the implementation's zero hash, and a one-element list returns that element directly. Consequently a usual one-transaction block has no internal tree beyond its leaf. Sensor anchoring is the more substantial example of summarising many leaves with one root.

Algorithm 1 Commit a validated invocation

1 Begin database transaction
2 Check contract preconditions
3 Encode arguments and add nonce
4 Compute transaction identifier
5 Apply operational mutations
6 Append block and valid transaction
7 Commit both changes atomically

On domain rejection: roll back, log reason

Full-chain verification visits blocks in ascending number order and sorts the transactions attached to a block by timestamp. It checks each transaction digest, recomputes the root from stored transaction identifiers, compares the predecessor with the previous stored block hash and recomputes the block header digest. The result includes issue categories and the block or transaction involved. This makes a failed payload check distinguishable from a broken predecessor relation.

Algorithm 2 Verify the stored chain

1 previous ← zero hash; issues ← empty
2 For every block in number order:
3 Load its ordered transactions
4 Recompute each transaction digest
5 Record mismatching transaction IDs
6 Recompute the transaction root
7 Compare root with stored root
8 Compare predecessor with previous
9 Recompute and compare header hash
10 previous ← stored block hash
11 Return counts and collected issues

The four implemented issue classes are transaction-hash, Merkle-root, block-hash and predecessor-link mismatches. A payload edit can produce a transaction-hash error while the stored block root remains unchanged, because that root is recomputed from stored transaction identifiers. The separate payload check is therefore necessary. Conversely, changing a stored identifier or root introduces other mismatches even if the payload comparison alone would not explain the altered block.

An internally consistent rewrite is outside this verifier's detection model when the attacker also replaces every relevant reference. Restoring an older self-consistent snapshot may also be invisible without an independent freshness reference. A production design would need a trusted external checkpoint, independently operated replicas or participant signatures with a suitable verification policy. Hash linkage supports comparison; it does not by itself establish which of two self-consistent histories is authoritative.

C. Documents and Sensor Evidence

Documents are stored outside the ledger. Their SHA-256 digests and descriptive metadata are committed through the document contract. Verification compares a freshly computed file digest with the stored reference and the ledger payload, and checks the referenced transaction digest. This identifies byte-level changes after anchoring. It does not determine whether a certificate was legitimately issued or whether it applies to the attached lot.

The IoT endpoint authenticates a registered device using a device key and accepts one reading or a batch of up to 500. It validates metric names and finite numeric values. An out-of-range value or a metric not permitted for that device is retained with an invalid flag; unknown metrics fail request validation. Keeping invalid readings supports later inspection without including them in resource totals.

A separate action hashes a device's latest 200 readings and records a Merkle root, count and time range through the sensor contract. Sensor storage and anchoring are therefore separate steps. The prototype supports an HTTP bridge suitable for an external MQTT relay, but does not implement or evaluate an MQTT broker or a physical field deployment. A key authenticates possession of a credential; it does not attest sensor calibration or measurement truth.

1) Document Integrity and Certificate Meaning

For an uploaded file, the application reads its bytes, computes a SHA-256 digest and saves the bytes in the selected storage backend. The document contract records the digest, size, kind and optional lot relation. The application accepts several evidence types, including certificates, laboratory reports, inspections, photographs and invoices.

A digest provides a content reference; the document kind and filename remain descriptive metadata that require appropriate interpretation.

Document verification can compare the stored file, the digest in the document record and the digest found in its referenced ledger transaction. A candidate file supplied for comparison is hashed independently. The verification result also indicates whether the referenced transaction's payload still matches its identifier. A missing stored file can be distinguished from a present file with mismatching bytes. These checks support an audit conversation about the evidence actually retained.

A matching digest does not validate a certificate issuer, expiry date, scope, accreditation or attachment to the correct farm. The planted cross-farm reuse example is therefore meaningful even when the bytes match an anchor. Certification is represented separately by an auditor or administrator event. For production, certificate scope and revocation would need explicit structured fields and an issuer-verification process rather than inference from a filename or an unchanged file.

2) Reading Semantics and Device Association

A registered device has an organisation, supported metrics, a key, a topic name and optional lot or field association. At ingestion, the server resolves the device from the key instead of accepting a client-supplied organisation as authority. The stored reading receives the device's lot association, metric unit and a timestamp supplied by the device or generated by the server. The key thereby identifies a submitting device record, while the lot relation supplies attribution context.

Request validation rejects malformed JSON, unknown metrics, nonfinite values and batches outside the allowed size. Range and device-metric checks have different semantics: a recognised but unpermitted metric, or a recognised value outside the configured range, is stored as invalid. That distinction preserves potentially useful diagnostic evidence. Validity means conformity with these configured checks, not independent confirmation of correct installation, calibration or timestamp accuracy.

Water and energy readings are treated as amounts to be summed. A physical integration must therefore define whether each reading is interval consumption, a flow rate or a cumulative counter. Counter differences and rate integration are preprocessing tasks that the current summation does not perform. Without that agreement, apparently valid values can produce misleading batch totals even if every numeric value falls within the configured range.

TABLE IV
Configured reading ranges and aggregation semantics

Metric	Range	Interpretation
Temperature	−10 to 60 °C	Observed condition
Humidity	0 to 100%	Observed condition
Soil moisture	0 to 100%	Observed condition
Water	0 to 5,000 L	Amount per reading
Energy	0 to 800 kWh	Amount per reading

3) Simulation and Batch Anchoring

The simulator combines a baseline, a daily sinusoidal component and Gaussian noise. Humidity and soil moisture use the opposite daily sign from temperature. Farm and default industrial profiles use different baselines and amplitudes. A chosen time step can receive an injected spike; the generated value is rounded and checked against metric limits. These profiles create a convenient functional stimulus, but they were not fitted to observed measurements from the named pilot facilities. Simulation uses the runtime clock and unseeded random draws, so regenerating the pilot need not reproduce identical sensor values or findings. This explains why a fixed report snapshot and a newly generated demonstration may have slightly different resource totals. It also means that reproducible analytics experiments need a frozen dataset or a seeded generator and a controlled evaluation clock. Neither can be replaced by reporting the same lot names alone. The anchoring action selects up to 200 recent readings ordered by recorded time and hashes the metric, value and timestamp of each. The recorded root, count and time interval identify a batch summary. The current action does not persist a dedicated immutable manifest of every selected reading identifier or generate portable inclusion proofs. A later verifier therefore needs the exact leaf membership, order and encoding convention to reproduce the same root.

D. Explainable Anomaly Scoring

The detector implements eight families of checks: transformation yield, field productivity, duplicate lots, reused documents, processing duration, missing events, stale transfers and sensor outliers. Findings contain a stable fingerprint, score, explanation, evidence and associated lot or organisation. Repeated scans update the finding while preserving an existing auditor verdict.

For transformation yield r, a group with at least five peer observations uses the peer median c. With MAD denoting the median absolute deviation, its scale is s = max(1.4826 MAD, 0.04 rexp). For smaller groups, c = rexp and s = 0.07 rexp. Thus a small pilot relies on a configured reference rather than a learned population distribution. The deviation and ranking score are

$$z = \frac{r - c}{s}$$
$$A(z) = \frac{1}{1 + \exp[-1.8(|z| - 2)]} \quad (4)$$

Yield findings require A ≥ 0.5. Severity thresholds are 0.60 for medium, 0.75 for high and 0.90 for critical. Several domain rules use fixed scores instead. The scale and mapping are implementation choices: A is an ordering score and has not been calibrated as the probability of fraud. Sensor checks require at least eight observations for a device and metric. Their scale is max(1.4826 MAD, 0.05|median|, 0.5); readings with |z| > 3.5 or an invalid flag are candidates. The engine groups these into one finding per device and metric. Productivity uses a configured 1,500 kg/ha reference with a 450 kg/ha spread. Such agricultural defaults must be adapted to region, season and growing conditions. Duplicate certificates are detected by identical digests attached to multiple lots. Some certificates legitimately cover several lots, so reuse is evidence for review rather than proof of misrepresentation. Exact duplicate-lot matching similarly checks identical stored attributes and quantities within a time window; it does not provide semantic duplicate detection.

1) Peer Selection and Robust Scale

Yield comparison groups transformations by output lot type. A candidate transformation must have parents, and each parent must differ in type from the child. Same-type split lots are therefore excluded from yield estimation. Each eligible ratio divides the output's kilogram mass by the kilogram-equivalent quantities consumed from its parents. This prevents a simple partial handoff from being mistaken for a low-yield manufacturing operation.

The median describes the centre of a group, while MAD describes the median distance from that centre. Multiplying MAD by 1.4826 produces a conventional normal-consistent scale estimate [6]. CottonTrace additionally applies a scale floor so that nearly identical peer yields do not make every small deviation numerically extreme. With fewer than five peers it uses the configured expected yield and a seven-percent relative spread. The reported two-bale example consequently uses the small-group branch.

This fallback is operationally useful but changes what the score means. A deviation from an expected reference is evidence of disagreement with that reference; it is not a deviation from a well-estimated empirical population. If the reference is inappropriate for a cotton variety, moisture content or machine configuration, a legitimate batch may be flagged. A mature deployment should stratify relevant peer groups and record which configuration version produced a finding.

2) Rule Families and Their Evidence

The harvest-productivity check divides a nonsplit harvest quantity by its field area and compares the result with a 1,500 kg/ha reference and 450 kg/ha spread. It raises findings only above its high-deviation condition. This catches the planted over-declaration but does not model weather, irrigation, variety, picking count or seasonal yield differences. The reference is a demonstration constant, so the phrase unusual productivity is more defensible than an assertion of agricultural impossibility.

Duplicate-lot detection compares original nonsplit lots from the same creator. It requires equal type, near-equal quantity, an identical stored attribute string and creation within 24 hours. The pairwise comparison has a clear explanation but is sensitive to the representation of attributes and scales quadratically with the number of candidate originals. Two semantically identical attribute objects encoded in a different order need not match this string-based rule.

Document reuse groups attached records by digest and flags a group spanning multiple lot codes. It does not first prove that the lots are unrelated or that reuse violates the certificate's scope. A scope certificate covering several consignments may therefore be legitimate. The finding presents the affected codes and common digest so that the auditor can distinguish permitted reuse from the planted cross-farm misattachment.

Processing-duration checks compare a child's creation time with the latest relevant readiness or receipt event on its parents. A duration below the stage's minimum produces a fast-processing finding. With sufficient observations, an unusually long duration can also be flagged relative to the group median. Negative intervals are not treated by the fast-duration branch as positive evidence, so clock errors and impossible ordering deserve a separate chronology check.

Missing-event checks cover bales older than 12 hours without either a quality-test event or a laboratory report, and published garment passports without any certified ancestor. A stale-transfer check flags pending transfers older than five days and raises its score with age up to a cap. These rules describe absent or delayed evidence. They should be reviewed against the actual business process before a delay is treated as misconduct.

Sensor checks use all loaded readings for each device and metric, including invalid values. At least eight readings are required. A scale floor limits division by a nearly zero MAD, and an outlier is a sufficiently large absolute deviation or an invalid value. One finding summarises the metric group, including its median, peak and outlier count. A finding count therefore cannot be interpreted as the number of anomalous individual readings.

TABLE V
Detector families and principal evidence

Family	Main evidence
Yield	Input mass, output mass, reference and z
Productivity	Harvest quantity and field area
Duplicate lot	Creator, attributes, quantity and time
Reused document	Shared digest and affected lot codes
Duration	Input readiness and output timestamp
Missing event	Expected test or certification
Stale transfer	Pending status and elapsed days
Sensor outlier	Device, metric, median and peak

3) Scores and Human Review

The logistic mapping is monotonic in absolute deviation: at |z| = 2 its value is 0.5, at |z| = 3 it is approximately 0.858, and at |z| = 4 it is approximately 0.973. These values are direct calculations from the implemented equation, not detection-rate measurements. The mapping makes different degrees of deviation convenient to rank but supplies no estimate of how often a flagged observation is actually wrong.

Several rules use fixed scores, including 0.82 for duplicate registration, 0.88 for document reuse and 0.62 for a missing bale test. These scores share a display scale with statistical scores without having a common probabilistic calibration. Critical, high, medium and low labels should therefore be understood as review priorities configured by the application. A score of 0.88 is not an 88-percent likelihood of fraud.

Algorithm 3 Detect and retain findings

1 Load lots, links, events and evidence
2 Evaluate yield and productivity rules
3 Compare duplicates and reused digests
4 Inspect durations and missing events
5 Inspect old pending transfers
6 Group device readings by metric
7 Score deviations and attach evidence
8 Upsert by stable fingerprint
9 Preserve any existing review verdict

The review action permits auditor and administrator roles to confirm, dismiss or reopen a finding and attach a note. The finding records the reviewer and review time. Repeated scans update evidence without automatically clearing an existing verdict. That preserves a decision, but it also requires care when changed evidence should trigger reconsideration. The present scan does not automatically close old findings that no longer appear in the latest result set.

E. Product Attribution and Passport

The passport traverses ancestors of a garment lot and presents the participating sites, process events, certifications and selected indicators. It recomputes full-chain integrity and the implemented per-lot checks for the upstream lots. A QR code links to the public batch identifier. A three-dimensional garment, process animation, genealogy graph and map help explain the record; their effect on consumer understanding has not been measured.

For target t, the attribution share is initialised as st = 1. For each parent p of child c, the implementation propagates the fraction of the parent quantity consumed by the child, accumulated across paths:

$$s_p = \sum_c s_c \frac{q_{pc}}{Q_p} \quad (5)$$

Here Qp is the parent lot's original quantity and qpc is the linked consumption quantity in the parent's unit. The implementation processes newer lots before older lots and caps a lot's applied share at one. The calculation assumes time-consistent lineage. It allocates recorded resource use; it is not a complete physical model of fibre composition or agricultural impacts.

Valid water and energy readings are summed with these shares. Dyeing attributes can add recorded process totals. The carbon indicator combines energy and estimated transport:

$$C = 0.71\,E + 0.105 \sum_l \frac{m_l}{1000} d_l \quad (6)$$

E is electricity in kWh, ml is allocated kilograms on transport leg l, and dl is estimated kilometres. The coefficients are the application's fixed factors in kg CO2e/kWh and kg CO2e/tonne-km; they are not treated as independently verified current emission factors. Distances use great-circle separation multiplied by 1.25. Batch totals are divided by the garment count to obtain per-piece values.

1) Upstream Traversal and Attribution Example

The traceability engine walks parent or child links with a frontier and a visited set. In the upstream direction, every frontier contains lots whose parents have not yet been explored. The loop collects newly encountered identifiers and unique edges, then loads the resulting lots and sorts them for stage presentation. A depth cap of 20 prevents unbounded traversal; a longer legitimate chain would need an explicit incomplete-result indication or a revised traversal policy.

Algorithm 4 Retrieve upstream genealogy
1 visited ← target; frontier ← target
2 Repeat while frontier is nonempty:
3 Read links whose child is in frontier
4 Add links to the edge collection
5 Add unseen parents to next frontier
6 Mark those parents as visited
7 Stop at the configured depth limit
8 Load lots for all visited identifiers
9 Return lots and collected edges

Consider the 5,000-piece brand allocation from the 7,500-piece garment batch. Its parent share is 5,000/7,500 = 2/3. The garment batch consumes all 1,850 kg of its dyed-fabric input, so that input inherits the same two-thirds share. The same reasoning propagates through 1,900 kg of fabric and the 2,000 kg yarn child. Because that child came from a 2,830 kg yarn parent, the original yarn's share is (2/3) × (2,000/2,830), approximately 0.4711.

The full chain then propagates this share through the fibre and mixed bale to the harvest quantities consumed by that bale. Shares represent the attribution convention used by the application. A link quantity is always read in the parent's unit, which is why dividing by that parent's original quantity is meaningful. The seed relations preserve cultivation ancestry even though seed mass does not equal harvest mass. Appendix A follows the arithmetic through the complete primary chain.

2) Resource Boundary and Route Approximation

Only valid readings linked to lots in the retrieved ancestry contribute to resource totals. A device associated solely with a field but without an attributed lot does not automatically become part of a passport total through the implemented lot-based query. Manual dyeing water and energy fields add another source of totals. A production installation must map meters to lots or production windows and reconcile those totals to avoid omissions and double counting.

The route model uses the geographic positions of the organisations that created connected lots and, for the final allocation, the relevant owner. Great-circle distance is multiplied by 1.25 as a road approximation, and repeated organisation pairs are aggregated. These coordinates describe sites, not a sampled vehicle trajectory. A map line consequently illustrates the recorded supply-chain relationship rather than proof that a vehicle followed that route.

The energy coefficient is fixed at 0.71 kg CO2e/kWh and the road coefficient at 0.105 kg CO2e/tonne-km in the reviewed implementation. This paper treats them as model parameters, not verified current regional emission factors. The carbon boundary omits unrecorded agricultural inputs, production chemistry, consumer use and disposal. The Product Standard's life-cycle perspective [13] helps explain why those omissions matter when communicating a product-level environmental claim.

3) Passport Verification and Presentation

The passport fetches a published record by its public identifier and loads its garment ancestry. It checks every upstream lot using the per-lot verifier and runs full-chain verification. Its aggregate status depends on the number of matching lot checks and on chain validity. The per-lot comparison covers code, type and original quantity; it does not independently validate all attributes, certification claims, ownership changes or each LotLink edge used by the journey calculation.

The rendered passport includes a material journey, locations, selected certifications, care instructions, end-of-life text and per-piece indicators. The colour of the three-dimensional garment follows the passport's shade, and the fibre-to-garment animation illustrates transformation stages. These features make the digital history accessible, but no user study establishes their effect on comprehension or trust. The scan counter also records page visits, so it is not a verified count of distinct physical garments or consumers.

V. EVALUATION DESIGN

A. Evidence and Pilot Scope

This paper combines inspection of the supplied implementation with results documented in the accompanying major-project report [14, Chs. 5–6]. All empirical counts and timings below are attributed to that existing evaluation; the experiments were not independently rerun for this manuscript. Code inspection establishes how the reported mechanisms are implemented, while the report supplies the observed pilot outcomes.

The synthetic scenario contains fictional organisations, 18 lots, 61 blocks including genesis, six anchored documents and 2,488 simulated sensor readings. It includes a primary cotton-to-garment chain and deliberately irregular records. The garment batch contains 7,500 pieces, of which a 5,000-piece allocation receives the illustrated passport. The reported evaluation covers 22 functional checks, selected invalid invocations, payload and file tampering, anomaly scans and local execution timing.

B. Evaluation Questions and Interpretation

RQ1 is examined through mixed inputs, transformations, partial handoffs and the upstream passport journey. RQ2 is examined through rejected contract calls, verification after edits, and unusual but contract-admissible records. RQ3 is examined through passport record checks and attributed resource calculations. These are scenario-based demonstrations. The dataset is not a held-out sample from an operating supply chain, and no baseline comparison, ablation study or blinded ground-truth labelling is available.

The report identifies its local timing environment as an Apple M1 computer with 8 GB RAM, using the embedded database. Core timings are reported as medians after a warm-up run, with run counts differing by operation. Raw timing traces and a dedicated benchmark harness are not supplied with the reviewed source, limiting independent assessment of variability and reproducibility.

C. Pilot Construction and Evaluation Protocol

The seed module constructs the principal business objects through the same contract functions used by normal application actions. It creates fictional organisations and their actors, establishes fields and devices, registers seed lots, records cultivation, creates harvests and proceeds through transfers and processing. The sequence spans approximately 80 days relative to the execution time. As a result, the pilot exercises several connected workflows rather than isolated example rows inserted directly into a result table.

Two farms supply the scenario. The main garment chain combines two harvest lots from the first farm. The second farm supplies irregular examples, including duplicate seed registration, elevated field productivity and a certificate reused from the first farm. Pending transfers provide elapsed-time cases, and device profiles provide ordinary variation plus injected spikes. The normal and unusual records were designed together, which is useful for demonstration but creates selection bias when estimating detection performance.

The project report supplies the observed outcomes for the functional checks, integrity demonstrations, anomaly scan and timing runs [14]. This expanded manuscript does not re-seed the database, alter the application or rerun the reported benchmarks. Additional numeric examples in the text and Appendix A are arithmetic derivations from the recorded quantities and formulas. They are labelled as derivations rather than presented as new measurements.

TABLE VI
Project-reported pilot inventory [14]

Record type	Count
Organisations and users	10 and 10
Fields	3
Material lots	18
Genealogy links	15
Traceability events	62
Transfers	11
Accepted and pending transfers	9 and 2
Documents	6
Devices and readings	5 and 2,488
Committed blocks including genesis	61
Seeded rejected invocations	4
Published passports	1

D. Outcome Definitions and Evidence Quality

A contract case passes in the reported functional evaluation when an admissible action changes the intended state or an inadmissible action returns the expected rejection reason. An integrity case passes when an edited payload or file produces the corresponding mismatch and restoration returns the check to a valid result. An anomaly case concerns whether a deliberately planted irregularity produces a corresponding finding. These are different units of evaluation and should not be merged into a single accuracy percentage.

A passport case evaluates whether the configured target can retrieve its upstream lots and obtain the implemented record checks. It does not establish complete verification of every displayed claim. A timing result is an elapsed time in the report's environment, with its stated warm-up and repetition count. Without raw traces, dispersion, a frozen database and an executable measurement harness, it remains a reported point estimate rather than an independently reproduced benchmark.

E. Reproducibility and Future Comparative Metrics

A reproducible repeat should preserve the source revision, dependency lockfile, database schema, generated dataset, clock reference and simulator random state. It should also record whether reads and writes use the embedded local database or the cloud database. Timing boundaries need to specify whether connection establishment, network transfer, cache effects and response rendering are included. These distinctions are especially important when comparing an in-process operation with an end-to-end browser request.

For a future field study, the unit of ground truth should be defined before computing precision or recall: a reading, lot, shipment, certificate attachment and review finding are different objects. Each injected or naturally occurring issue should have an independent label, and the normal comparison set should include legitimate exceptions such as multi-lot certificates. Reporting false alerts per auditor-day and time to resolution would complement conventional detection metrics.

A comparative experiment could evaluate rules alone, yield statistics alone and their combination against the same held-out cases. A separate experiment could compare the current detector with Isolation Forest [12] using agreed features and labelling. Operational tests should increase lot count, graph depth, sensor history, simultaneous users and contention on the same stock. These are proposed evaluation procedures, not results obtained in the present paper.

VI. RESULTS AND ANALYSIS

A. Material Traceability and Contract Checks

The reported ledger contains one genesis transaction and 60 subsequent committed transactions [14, Table 6.1]. These include 15 lot creations, 14 event records, 11 transfer initiations, nine acceptances, six document anchors, three certifications, one sensor anchor and one passport publication. Partial acceptance creates additional lot records without requiring a separate lot-creation invocation, explaining why the lot count exceeds the number of creation calls.

TABLE VII
Reported primary-chain transformations [14]

Input → output	Input kg	Output kg	Yield
Harvest → bale	10,100	3,535	35.0%
Bale → fibre	3,535	3,290	93.1%
Fibre → yarn	3,290	2,830	86.0%
Yarn → fabric	2,000	1,900	95.0%
Fabric → dyed fabric	1,900	1,850	97.4%
Dyed fabric → garments	1,850	1,425	77.0%

Table VII shows the recorded quantities. Garment output is 7,500 × 0.19 = 1,425 kg. Its 77.0% yield is below the configured 80% expectation and remains within the 95% hard limit. The chain also includes a planted second bale of 3,870 kg from 9,000 kg of input, corresponding to a 43% yield.

The report documents rejection of a nonexistent-lot transfer, a bale above the mass limit, an invalid yarn-to-brand route, unauthorised certification and a handoff exceeding available stock [14, Table 6.3]. Four were seeded checks and one was an interactive check. These examples support the exercised contract paths; they do not establish exhaustive correctness under all inputs or concurrent requests.

The main chain illustrates why both lot identity and quantity-labelled edges are required. Two harvest lots contribute 10,100 kg to one 3,535 kg bale. Later, only 2,000 kg of the 2,830 kg yarn output is transferred into the fabric branch. If the system retained only the final garment identifier and a list of organisations, it could not reconstruct that partial allocation. The original lot quantities and split relations provide the denominators and paths used by the passport calculation.

The rejected 5,000 kg bale from 9,000 kg of input has an actual ratio of 55.56 percent when calculated from the recorded quantities. It exceeds the 45-percent cap by a wide margin. The report rounds this example to approximately 55 percent, but the decision is unchanged. Separating the exact arithmetic from the displayed approximation avoids treating a rounded percentage as the underlying contract computation.

B. Integrity Checks

The reported payload edit changed the original bale's committed quantity from 3,535 to 4,949 kg without updating its transaction hash. Full-chain verification returned a transaction-hash violation, and the corresponding per-lot check failed. Replacing a certificate with a modified copy produced a digest mismatch. Restoring the originals returned the checks to a valid state [14, Table 6.4].

These observations demonstrate detection of inconsistent local edits. They do not test coordinated rewriting of every dependent hash, compromise of the verifier, cryptographic identity impersonation or Byzantine agreement. The report describes repeated trials but does not supply their count; consequently no detection-rate estimate or confidence interval is derived.

A second integrity distinction concerns where the change occurs. Editing a committed transaction payload causes the transaction-digest comparison to fail. Editing an operational lot quantity can fail the per-lot comparison even if the chain still re-hashes. Editing a file behind an unchanged storage reference is a document-content problem. These are different checks with different references; a valid full chain alone does not prove that every operational row and external file remains consistent with it.

C. Anomaly Findings

TABLE VIII
Eight project-reported pilot findings [14]

Finding	Score	Severity
Soil-moisture spike	1.000	Critical
Temperature spike	0.983	Critical
Bale yield of 43%	0.907	Critical
Certificate reused across lots	0.880	High
Duplicate seed registration	0.820	High
Harvest productivity outlier	0.808	High
Pending shipment delay	0.788	High
Missing bale quality test	0.620	Medium

The project report records eight findings corresponding to the eight planted irregularities and no additional findings in that run [14, Table 6.5]. Table VIII reproduces those reported scores. This count concerns findings rather than eight independent fraud cases: sensor spikes and multiple flags on a shared lot are correlated, and each detector family is not necessarily triggered by this scenario.

The 43% bale is the clearest example of complementary checks. It satisfies the 45% contract cap. With the small-group reference, however, $z = (0.43 - 0.35)/(0.07 \times 0.35) = 3.27$, giving $A \approx 0.907$. It therefore enters the critical review band. Hashing preserves the statement, the contract admits it, and analytics questions its plausibility. The pilot demonstrates this workflow without establishing population precision, recall or false-positive rates.

The planted high-productivity harvest supplies 13,800 kg from a five-hectare field. Dividing gives 2,760 kg/ha, and the configured deviation is (2,760 − 1,500)/450 = 2.8. The logistic mapping yields approximately 0.808, matching the reported high-severity finding. This arithmetic checks consistency between the stated rule and the reported score; it does not independently establish that 2,760 kg/ha is implausible for every growing condition. The reported stale-shipment score is 0.788. Although the report describes the elapsed period as nine days, the implementation uses fractional elapsed days relative to the scan clock. Display rounding and the actual scan time can therefore differ slightly from the score calculated using exactly nine whole days. This reinforces the need to retain timestamps and evaluation time when reproducing duration-based findings.

A single scan result also cannot establish coverage of all eight detector families. The eight displayed findings include two device-metric outlier groups, while other rules may produce no finding. Several findings refer to the same harvest or its downstream branch. The evidence supports the designed examples; estimates of independent case sensitivity, missed issues or workload would require a larger, explicitly labelled evaluation set.

D. Passport and Resource Indicators

The illustrated passport traverses 12 upstream lots and reports matching identity, type and quantity checks for all 12, together with a valid full-chain check [14, Tables 6.6–6.7]. The reported route comprises six legs with a total estimated distance of 3,459 km. Table IX presents the per-garment indicators for the report's 5,000-piece allocation.

TABLE IX
Project-reported passport indicators [14]

Indicator	Per garment
Allocated metered water	17.8 L
Allocated process energy	0.94 kWh
Energy and transport carbon	0.79 kg CO2e

Water and energy originate from the simulated and manually recorded pilot data. The report notes small changes when sensor noise is regenerated. Water excludes unmodelled life-cycle components such as rainfall, and carbon covers only the implemented energy and transport terms. The values should therefore be read as illustrative allocated indicators, rather than independently audited product footprints. The reported batch totals are 89,014 L of water, 4,714 kWh of energy and 3,938 kg CO2e for 5,000 pieces [14]. Dividing by the piece count gives 17.8028 L, 0.9428 kWh and 0.7876 kg CO2e, which round to the displayed passport values. This is a consistency calculation from the report snapshot. It does not recover the original randomly generated readings or independently validate the resource inventory. The six reported route legs are reproduced in Table X. Their allocated masses differ because different processing stages carry different material quantities. Summing those masses would double count material moving through successive stages. Transport emissions instead sum each leg's tonne-kilometres multiplied by the configured factor. The total route distance is a sum of estimated legs and should not be read as the journey of a uniquely identified fibre.

TABLE X
Reported route distances and allocated masses [14]

Transport leg	km	Allocated kg
Rajkot → Kadi	247	4,759
Kadi → Coimbatore	1,814	1,665
Coimbatore → Tiruppur	54	1,333
Tiruppur → Erode	61	1,267
Erode → Bengaluru	227	1,233
Bengaluru → Mumbai	1,057	950
Total estimated distance	3,459	—

E. Reported Performance

The project report gives median local times of 1.8 ms for full verification of 61 blocks, 5.2 ms for genealogy traversal, 6.7 ms for passport calculations and 15.8 ms for the anomaly scan [14, Table 6.8]. Full-chain verification is additionally reported at five chain sizes, as replotted in Fig. 4. The 61-block point uses the core timing experiment; the additional sizes use nine-run medians.

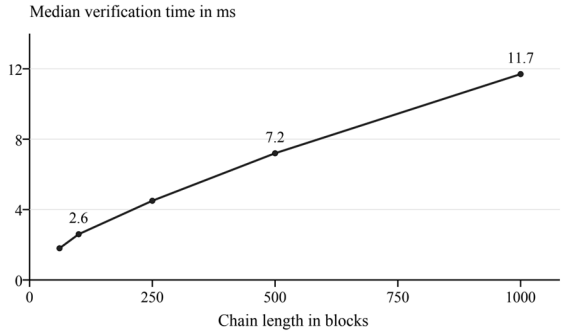


Fig. 4. Local verification medians reported in [14, Sec. 6.6]. These are prior project measurements, not new benchmark runs.

The reported trend is consistent with a verifier that scans blocks and transaction payloads. It is not evidence of distributed-ledger throughput. The results also do not quantify concurrent inventory contention, cold starts, growing sensor histories or multitenant access. Duplicate-lot detection compares pairs, and each passport request verifies the full chain; these paths warrant separate scaling experiments.

Table XI includes the report's full set of local core-operation medians and repetition counts. Cryptographic micro-operations, database-backed graph traversal and a complete anomaly scan have different timing boundaries and workloads. Their values should be compared only with those boundaries in mind. A 0.4 ms file digest and a 0.4 ms per-lot query are not evidence that the operations have the same scaling behaviour.

For the reported production deployment, the passport endpoint has a 0.99 s median and a 0.82–1.31 s range over seven warmed requests. The QR endpoint has a 0.30 s median and a 0.29–0.31 s range [14, Table 6.9]. These results include public network and deployment effects and were recorded for that report. They are not current availability checks or service guarantees. Differences from local millisecond operations are expected because the measured paths are different.

The source also suggests where larger workloads may become expensive. The verifier loads the stored chain, the duplicate-lot rule compares pairs, and sensor analysis loads historical readings by device. Genealogy traversal issues successive frontier queries, while attribution repeatedly examines incoming edges. The pilot timings show that the small demonstration is workable; they do not establish a production capacity limit or justify an extrapolated transactions-per-second claim.

TABLE XI
Local execution medians reported in the project evaluation [14]

Operation	Median ms	Runs
SHA-256 of 1 MB file	0.4	50
Merkle root of 1,000 leaves	0.8	50
Verify 61-block chain	1.8	15
Verify one lot	0.4	30
Retrieve genealogy	5.2	15
Calculate passport journey	6.7	15
Run complete anomaly scan	15.8	7
Invoke contract and commit	3.0	15

VII. DISCUSSION AND LIMITATIONS

A. Answers to the Research Questions

RQ1 is addressed at the data-model level: quantity-labelled links preserve provenance through mixing and partial transfers, and the pilot passport retrieves an upstream subgraph spanning the configured material types. Seed ancestry, process mass and garment counts nevertheless have different meanings. Retaining these units and semantics is necessary to avoid interpreting every genealogical edge as physical mass conservation.

RQ2 is addressed by three complementary decisions. Contracts determine whether an operation is admissible; hashes reveal whether retained evidence has changed relative to its reference; anomaly checks identify records that warrant explanation. An auditor's judgement is still needed to distinguish legitimate exceptions from errors or misrepresentation. A high score or a matching digest cannot replace verification of the underlying physical process.

RQ3 is supported as a presentation and calculation demonstration: a QR code provides access to batch history and resource attribution. Consumer understanding, supplier reporting accuracy and improvements in purchasing or audit decisions remain questions for user and field studies.

B. Validity of the Reported Evaluation

Internal validity is limited by planted irregularities, the small dataset and the absence of raw measurement logs. External validity is limited by fictional organisations, simulated devices and the absence of live enterprise integrations. No comparison isolates the incremental value of sensors, contracts or analytics. The scoring thresholds, yield expectations and resource factors also require domain validation and sensitivity analysis.

C. Trust Boundaries and Access Control

The central security limitation is the shared administrative boundary. An attacker controlling both the database and application can rewrite history and its references. The system has no cryptographic endorsements from independent participants or externally anchored chain head. Even within an intact chain, per-lot verification covers only selected fields, and consumer-visible attributes can exceed that verified scope. A production claim should explicitly enumerate which fields and evidence have been checked.

Membership deserves separate attention from sign-in. The current onboarding action permits a signed-in user to create an organisation or join an existing one, then assigns the role associated with its type. It does not implement independent consortium approval of the claimed organisation or accreditation. Consequently an auditor-labelled account in this prototype is a demonstration role. A production consortium would need controlled invitations, verified organisational authority and a governed certification role before relying on those permissions. The document download endpoint checks for a valid session but does not perform an organisation-specific authorisation check on each file in the reviewed code. Several list views deliberately provide network-wide visibility to auditors, brands and administrators. These choices are acceptable only within an agreed sharing model; they do not establish confidential tenant isolation. A production passport and evidence service should explicitly separate publicly disclosed fields, partner-visible evidence and restricted commercial records.

Device credentials also need an operational lifecycle. The implemented device key authenticates the registered entry, but possession of that key does not establish that a reading came from uncompromised hardware. The ingestion path does not provide a dedicated signed-message sequence, replay ledger or calibration certificate. These are concrete extensions for a physical trial. They should be tested as device-data controls instead of being assumed to follow from later Merkle anchoring.

D. Environmental and Data Quality Constraints

Resource attribution has further constraints. A manually entered dyeing total may overlap with sensor totals unless their boundaries are reconciled. Repeated meter values must represent interval consumption before they are summed. Estimated transport routes, fixed emission factors, time-ordered allocation and share capping introduce modelling assumptions. None of the reported indicators substitutes for an independently defined and validated life-cycle assessment.

VIII. CONCLUSION AND FUTURE WORK

CottonTrace demonstrates an integrated cotton traceability workflow connecting eight material types, contract-checked inventory operations, hash-based evidence checks, simulated IoT data, explainable anomaly review and a public Digital Product Passport. The project-reported pilot shows how a transaction can remain within a hard processing limit while still requiring auditor attention. This separation of admissibility, evidence consistency and plausibility is the central practical result.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

The next evaluation should use prospectively collected field data, independently labelled cases and a versioned benchmark harness. It should report uncertainty, false alerts, missing evidence and concurrent stock behaviour. Architectural work should add independently operated peers or external checkpoints, cryptographic participant identities, broader record verification and EPCIS-compatible events. Device calibration, sensor replay protection and certificate-scope validation are also necessary before production use. A field deployment and a consumer study can then test the value of the passport beyond the present synthetic demonstration.

A. Practical Development Priorities

The first priority is to freeze and validate the existing model: record dataset versions, add repeatable benchmark cases, make verification scope explicit and test simultaneous actions against the same lot. The next priority is membership and evidence governance, including organisation approval, document access policies and certificate scope. These steps address the trust assumptions visible in the current implementation before adding more complex prediction models or stronger public claims.

A physical pilot should then connect a small number of calibrated meters and a clearly defined processing window to actual lot identifiers. It should compare recorded input and output weights with independent records and reconcile water and energy readings with meter semantics. Supplier and auditor feedback should be collected on data-entry effort, false alerts, missing information and the clarity of rejection reasons. Success would be judged by usable, verifiable records rather than by the number of connected technologies. Distributed deployment is a separate architectural step. The common invocation boundary is a useful interface to replace, but moving to Fabric would require identity issuance, independently operated peers, endorsement policy, ordering, chaincode packaging and reconciliation with the application database. It is not merely a change of hash function or hosting provider. A comparative deployment should measure those additional guarantees and costs on the same workload.

IX. ACKNOWLEDGMENT

The authors acknowledge Dr. Puja Gupta, Department of Information Technology, Shri Govindram Seksaria Institute of Technology and Science, Indore, for supervision of the CottonTrace major project.

A. Appendix A Worked Seed to Garment Attribution

1) Recorded Material and Allocation Quantities

This example derives the application's attribution shares from the primary-chain quantities reported in [14]. It is an arithmetic reconstruction of the model, not an additional sensor experiment. Two harvests of 6,100 kg and 4,000 kg are combined into a 3,535 kg bale. Subsequent quantities are 3,290 kg of fibre and 2,830 kg of yarn. A 2,000 kg yarn allocation becomes 1,900 kg of fabric, then 1,850 kg of dyed fabric and 7,500 garments weighing 0.19 kg each. The brand receives a 5,000-piece child allocation.

The target has share one in itself. The garment parent has share 5,000/7,500 = 0.666667. Because that parent consumes the complete dyed-fabric input, the dyed-fabric share is also 0.666667. The same share reaches the fabric and the 2,000 kg yarn child. The original 2,830 kg yarn parent receives 0.666667 × 2,000/2,830 = 0.471143. Full consumption of the earlier fibre and bale carries that share upstream without another partial-allocation reduction.

TABLE XII
Derived upstream shares in the primary pilot example

Lot or stage	Share
Retail garment allocation	1.000000
Original garment batch	0.666667
Dyed fabric and fabric	0.666667
Transferred 2,000 kg yarn child	0.666667
Original 2,830 kg yarn	0.471143
Fibre and mixed bale	0.471143
Each fully consumed harvest	0.471143
18 kg seed lot	0.471143
12 kg seed lot with 6 kg linked	0.235571

The two seed shares differ because the first harvest uses all 18 kg of its seed lot, while the second harvest links 6 kg of a 12 kg seed lot. This does not mean that the first farm's seeds constitute that fraction of the garment's physical mass. It is an allocation of the recorded upstream activity along the genealogical relation. The meaning of that relation should remain explicit whenever agricultural readings are attached to seed lots.

2) Allocated Transport Mass and Indicators

For the farm-to-gin relationship, the allocated input is 10,100 × 0.471143 = 4,758.54 kg, which rounds to the report's 4,759 kg. The bale-to-spinning quantity is 3,535 × 0.471143 = 1,665.49 kg. The yarn transfer contributes 2,000 × 0.666667 = 1,333.33 kg; fabric contributes 1,266.67 kg; dyed fabric contributes 1,233.33 kg; and the final garment allocation weighs 5,000 × 0.19 = 950 kg. These values explain why route legs use different masses.

The report's aggregate resource values produce 89,014/5,000 = 17.8028 L of water, 4,714/5,000 = 0.9428 kWh of energy and 3,938/5,000 = 0.7876 kg CO2e per garment. Rounding gives 17.8 L, 0.94 kWh and 0.79 kg CO2e. The divisions verify the relationship between batch totals and displayed per-piece indicators. They do not verify the completeness of the inputs, the suitability of emission factors or the environmental performance of a real garment.

B. Appendix B Functional Checks and Implementation Mapping

Table XIII summarises the 22 checks documented in the project report [14, Table 5.3]. Every listed outcome was reported as passing in that source. The table is a record of the supplied evaluation, not a new execution log or a security certification. In particular, successful static checks and selected role tests do not establish exhaustive correctness, confidentiality or resistance to concurrent stock races.

TABLE XIII
Functional checks and reported outcomes [14]

ID	Case and reported outcome
01	Valid credentials create a session
02	Protected dashboard redirects a signed-out user
03	Excess-quantity transfer is rejected
04	Permitted partial handoff creates a pending transfer
05	Bale above the mass cap is rejected
06	Yarn-to-brand route is rejected
07	Knitting unit cannot certify its fabric
08	Nonexistent lot cannot be transferred
09	Edited transaction payload produces a hash issue
10	Restored transaction verifies again
11	Modified stored certificate produces a mismatch
12	Restored certificate matches its anchor
13	New document receives a ledger anchor
14	Unknown device key receives HTTP 401
15	Mixed-range readings are stored with an invalid flag
16	Unknown metric receives HTTP 400
17	Scan returns the eight planted findings
18	Public passport verifies 12 upstream records
19	Unknown passport returns HTTP 404
20	Unauthorised reseeding is rejected
21	Lot search is case-insensitive
22	Type check, lint and production build pass

The reviewed modules separate responsibilities as follows. The domain definitions hold stage types, routes, units, limits and metric ranges. The contract module performs business validation and mutation. The ledger module constructs and verifies digests and blocks. The detector produces review findings. The journey module retrieves genealogy and calculates attribution, routes and indicators. Sensor ingestion, storage actions and public passport rendering connect those functions to the user-facing application.

For a repeatable release, these module boundaries provide useful validation points: serialized payload fixtures for the ledger, quantity and state transitions for contracts, labelled records for the detector, and known graphs with expected shares for the journey engine. Document byte fixtures and controlled device submissions can test the evidence paths. This is a proposed verification structure derived from the implementation, not a claim that the repository already contains that complete test suite.

REFERENCES

[1] T. K. Agrawal, V. Kumar, R. Pal, L. Wang, and Y. Chen, “Blockchain-based framework for supply chain traceability: A case example of textile and clothing industry,” Computers & Industrial Engineering, vol. 154, Art. no. 107130, 2021. doi: 10.1016/j.cie.2021.107130.

[2] GS1, EPCIS Standard, release 2.0, Jun. 2022. Available: https://ref.gs1.org/standards/epcis/2.0.0/

[3] E. Androulaki et al., “Hyperledger Fabric: A distributed operating system for permissioned blockchains,” in Proc. 13th EuroSys Conf., Art. no. 30, pp. 1–15, 2018. doi: 10.1145/3190508.3190538.

[4] National Institute of Standards and Technology, Secure Hash Standard, FIPS PUB 180-4, Aug. 2015. doi: 10.6028/NIST.FIPS.180-4.

[5] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” ACM Computing Surveys, vol. 41, no. 3, Art. no. 15, pp. 1–58, 2009. doi: 10.1145/1541880.1541882.

[6] NIST/SEMATECH, “Detection of outliers,” e-Handbook of Statistical Methods, Sec. 1.3.5.17. Available: https://itl.nist.gov/div898/handbook/eda/section3/eda35h.htm

[7] European Parliament and Council of the European Union, Regulation (EU) 2024/1781 establishing a framework for the setting of ecodesign requirements for sustainable products, Jun. 2024, Arts. 9–11. Available: https://eur-lex.europa.eu/eli/reg/2024/1781/oj/eng

[8] N. Kshetri, “Blockchain’s roles in meeting key supply chain management objectives,” International Journal of Information Management, vol. 39, pp. 80–89, 2018. doi: 10.1016/j.ijinfomgt.2017.12.005.

[9] K. Wüst and A. Gervais, “Do you need a blockchain?,” in Proc. Crypto Valley Conf. Blockchain Technology, pp. 45–54, 2018. doi: 10.1109/CVCBT.2018.00011.

[10] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” Computer Networks, vol. 54, no. 15, pp. 2787–2805, 2010. doi: 10.1016/j.comnet.2010.05.010.

[11] A. Banks, E. Briggs, K. Borgendale, and R. Gupta, Eds., MQTT Version 5.0, OASIS Standard, Mar. 7, 2019. Available: https://docs.oasis-open.org/mqtt/mqtt/v5.0/os/mqtt-v5.0-os.html

[12] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation forest,” in Proc. 8th IEEE Int. Conf. Data Mining, pp. 413–422, 2008.

[13] World Resources Institute and World Business Council for Sustainable Development, Product Life Cycle Accounting and Reporting Standard, Greenhouse Gas Protocol, 2011. Available: https://ghgprotocol.org/product-standard

[14] P. Prajapati, D. Vatyani, and A. Sharma, Cotton Seed-to-Cloth Production: Blockchain-Enabled Traceability, IoT, AI and Digital Product Passport System, major-project report, Dept. of Information Technology, SGSITS, Indore, academic session 2026–2027, Chs. 3–6. Unpublished project report supplied with the implementation.

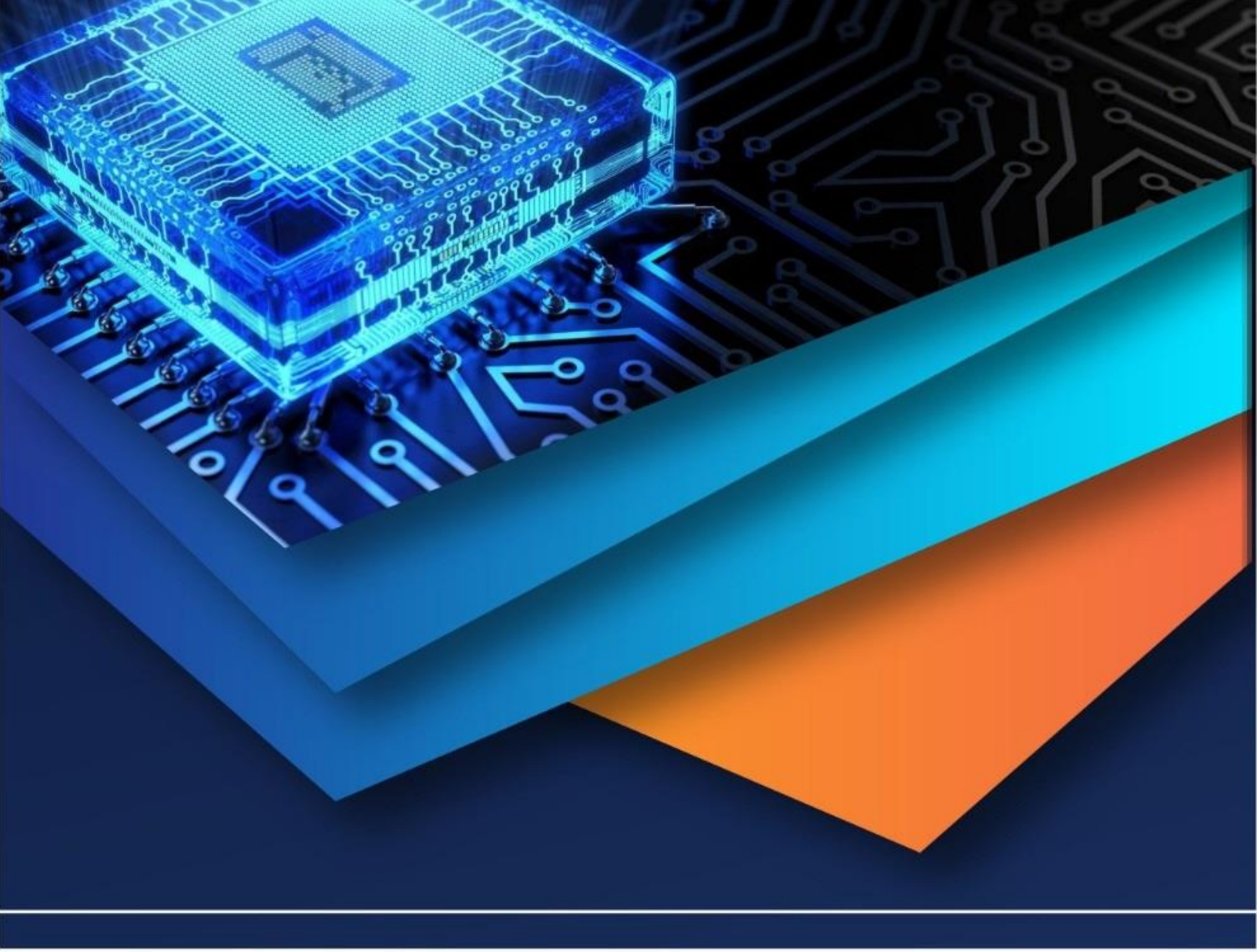
[15] Gupta, P., Sharma, V. and Varma, S., 2021. People detection and counting using YOLOv3 and SSD models. Materials Today: Proceedings.

[16] Gupta, P., Sharma, V. and Varma, S., 2022. A novel algorithm for mask detection and recognizing actions of human. Expert Systems with Applications, 198, p.116823.

[17] Gupta, P. and Kulkarni, N., 2013. An introduction of soft computing approach over hard computing. International Journal of Latest Trends in Engineering and Technology (IJLTET), 3(1), pp.254-258

[18] Kushwaha, U., Gupta, P., Airen, S. and Kuliha, M., 2022, December. Analysis of CNN Model with Traditional Approach and Cloud AI based Approach. In 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS) (pp. 835-842). IEEE

[19] Gupta, P., Arya, N., Singar, C.P., Chaudhari, A., Singh, U. and Gupta, S., 2025. Safety of Pedestrians in AI-Optimized VANETs for Autonomous Vehicles via Real-Time Vehicle-to-Vehicle Communication. In AI-Driven Transportation Systems: Real-Time Applications and Related Technologies (pp. 169-181). Cham: Springer Nature Switzerland.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)