**Journal of Applied Informatics and Computing (JAIC)**
Vol.10, No.4, August 2026, pp. 3719~3732
e-ISSN: 2548-6861


# A Blockchain-Based Traceability System To Mitigate Counterfeit Cosmetics In E-Commerce: Proposed Model And Experimental Validation

**Dinh Thien Tri [1], Hoang Gia Han [1], Phan-Anh-Huy Nguyen [*]**
* Faculty of Economics, Ho Chi Minh City University of Technology and Engineering, Vietnam
huynpa@hcmute.edu.vn [*]

**Article Info**

***Article history:***

Received 2026-04-18
Revised 2026-07-25
Accepted 2026-08-10

***Keyword:***

*Blockchain,*
*Counterfeit Detection,*
*Cosmetic Traceability,*
*Smart Contract,*
*Supply Chain Transparency.*

**ABSTRACT**

The rapid expansion of e-commerce has intensified the proliferation of counterfeit cosmetics, posing serious risks to consumer health and brand integrity, particularly in emerging markets such as Vietnam. While blockchain-based traceability has been studied extensively, existing solutions rarely integrate dual-layer verification, privacy-preserving mechanisms, and hybrid off-chain storage into a cohesive, experimentally validated prototype. This paper proposes VeriGlow, a blockchain-based decentralized application (DApp) that delivers transparent, tamper-proof, and user-verifiable traceability for cosmetic products throughout the supply chain. The system integrates: (1) Ethereum smart contracts written in Solidity ^0.8.0 with structured functions including createProduct, transferOwnership, and getProductJourney for immutable lifecycle management; (2) a hybrid IPFS-on-chain storage model using Pinata-hosted Content Identifiers (CIDs) to synchronize off-chain media assets with on-chain metadata, significantly reducing gas consumption; (3) a dual verification mechanism combining dynamic QR codes with physical Batch IDs to prevent QR cloning and relay attacks; and (4) a privacy-preserving historySnapshot mechanism that selectively conceals intermediate ownership records from public view while retaining full on-chain auditability. A functional prototype was developed using Solidity, Web3.js, and MetaMask and validated against six test scenarios on the Ethereum Sepolia testnet and Ganache local network. Experimental results demonstrate product creation gas costs in the range of 245,000–310,000 gas units (approximately 0.0014–0.0018 ETH), ownership transfer costs of 58,000–72,000 gas units (approximately 0.0008 ETH), and zero-cost read operations, comparing favorably with prior Ethereum-based traceability systems. Transaction confirmation times averaged 5–7 seconds for creation and 3–5 seconds for transfers on the local network. All six test scenarios passed with 100% success rate across ten repeated executions. These findings validate VeriGlow as a technically feasible and practically applicable solution for combating counterfeit cosmetics in e-commerce environments and offer a replicable framework for blockchain-based traceability in similar high-risk consumer sectors.



## I. INTRODUCTION

The rapid expansion of e-commerce has revolutionized global commerce, particularly in the cosmetics sector, where consumers can easily access a wide variety of products through online platforms. However, this growth has been accompanied by a significant increase in counterfeit and substandard cosmetics. These fake products not only cause substantial economic losses to legitimate brands but also pose serious health risks to consumers due to the use of unsafe ingredients and non-compliant manufacturing processes.

Traditional anti-counterfeiting methods such as hologram labels, barcodes, and static QR codes have proven inadequate

in the digital marketplace. These techniques are easily duplicated, while centralized databases remain vulnerable to tampering and single points of failure. As a result, there is an urgent need for more robust, transparent, and tamper-proof solutions to ensure product authenticity across complex supply chains.

Blockchain technology has emerged as a promising solution due to its decentralized, immutable, and transparent nature. By leveraging smart contracts, blockchain enables automated, verifiable recording of product information, ownership transfers, and transaction history without intermediaries. Several studies have explored the integration of blockchain with supply chain management and big data systems, demonstrating its potential in enhancing data integrity and traceability [Linoy et al. 2019; Chen et al. 2019; Demirbaga and Aujla 2022].

Despite growing interest, most existing research remains at the conceptual or theoretical level, with limited practical implementation and experimental validation, especially in emerging markets like Vietnam. In Vietnam, the surge in e-commerce has led to a sharp rise in counterfeit cosmetics, yet domestic studies have largely focused on theoretical analysis rather than developing and testing functional systems.

To address these gaps, this study proposes VeriGlow — a blockchain-based decentralized application (DApp) for cosmetic traceability on e-commerce platforms. The system integrates Ethereum smart contracts, IPFS hybrid storage, dynamic QR codes, and a two-layer verification mechanism (QR code + Batch ID) to ensure immutable records and strong resistance to counterfeiting. A functional prototype was developed using Solidity, Web3.js, and MetaMask, and thoroughly validated through simulation on the Ethereum Sepolia testnet using Ganache. While blockchain-based traceability has been studied extensively, VeriGlow makes several specific contributions beyond the existing literature: (1) it delivers a fully implemented and experimentally validated prototype rather than remaining at a conceptual or architectural proposal level; (2) it introduces a novel dual-layer verification mechanism combining dynamic QR codes with physical Batch IDs that resists QR cloning attacks — a vulnerability identified but not addressed in prior QR-based systems; (3) it proposes a privacy-preserving historySnapshot mechanism that achieves selective disclosure of ownership records, balancing transparency with commercial confidentiality in a manner not previously demonstrated in cosmetics traceability systems; and (4) it provides quantitative gas consumption benchmarks compared with prior related systems, offering empirical evidence of the system's operational efficiency.

## II. RELATED WORKS

The integration of blockchain technology into traceability systems has gained considerable attention as an effective solution to combat counterfeit products, especially in high-risk sectors such as cosmetics sold on e-commerce platforms. Blockchain's inherent features of immutability, transparency, and decentralization address key limitations of traditional methods, including data tampering, lack of consumer trust, and difficulty in verifying product authenticity. Several recent studies have explored the application of blockchain, smart contracts, QR codes, and related technologies for supply chain transparency and anti-counterfeiting. This section reviews 20 closely related works, highlighting their contributions, methodologies, and limitations.

The work presented in [MacCarthy et al. 2024] examines the potential of blockchain in ensuring provenance claims, certification, and sustainability within the fragmented global fragrance and cosmetics supply chain. The authors evaluate permissioned, consortium, and hybrid blockchain models and emphasize the importance of partner collaboration and governance. However, the study focuses more on high-level consortium governance than on detailed smart contract implementation. Similarly, [Shao et al. 2025] apply a game-theoretic model combined with blockchain to reduce counterfeiting incentives at the source. Their permissioned and hybrid blockchain approach provides theoretical insights but remains limited to simplified models without real-world prototype testing.

In the context of e-commerce counterfeiting, [Lee et al. 2023] propose a blockchain traceability (BCT) framework using examples such as JD Blockchain and AntChain. Their game-theoretical analysis highlights the effectiveness of blockchain in combating fakes at the source, yet the study lacks dynamic interactions and government intervention details. [Nehete et al. 2024] and [Pragati Nehete et al. 2024] develop Ethereum-based traceability systems specifically applicable to cosmetics, incorporating Solidity smart contracts, QR codes, Web3.js, MetaMask, and Ganache/Truffle for simulation. Both studies include detailed gas fee analysis for functions such as product creation and ownership transfer, but they note scalability limitations and reliance on simulation environments.

[Adedamola et al. 2024] and [Md. Adib Khan et al. 2024] focus on decentralized traceability prototypes using public/permissionless blockchain, QR codes, and IoT. Their Ethereum + Solidity + IPFS implementations demonstrate end-to-end visibility, with smart contracts handling product registration and verification, including account abstraction (ERC-4337). [Wasnik et al. 2022] and [Kunal Wasnik et al. 2022] propose DApps for counterfeit detection that combine Ethereum smart contracts with QR/Barcode technology and local Ganache testing. These works emphasize ownership transfer and QR-linked verification but highlight risks of QR code copying and the absence of real-world enterprise deployment.

Studies with broader scope include [Shreekumar et al. 2022], which reviews blockchain and IoT integration across multiple countries, and [S Saraswathi et al. 2024], which implements Hyperledger Fabric with Raft consensus and chaincode for supply chain traceability. On the consumer behavior side, [Xiaoling & Dawod 2025] and [Ly et al. 2025] investigate adoption intentions in Vietnam using extended UTAUT models and Deep Signature technology, respectively. While valuable for understanding user

acceptance, these works do not provide technical smart contract implementations. Theoretical and signaling-focused research such as [Xiaoling Liu & Dawod 2025] and [Ruba Islayem et al. 2025] apply signaling theory to blockchain-based traceability on platforms like Tmall Global and JD Smart Check, emphasizing consumer trust signals rather than code-level details.

Further contributions come from [Fangfang Guo et al. 2021], who model an anti-counterfeit traceability platform (BATP) using differential game theory and smart contracts, and [Neo C.K. Yiu 2021], who incorporate big data analytics and IoT into blockchain-enabled traceability systems. [Nira Nur Fitriani & Zulfa Eliza .2024] explore blockchain for Halal assurance and traceability in a theoretical framework, while [Anthony et al. 2023] develop a complete Ethereum prototype with IPFS, Web3.js, and functional testing on Ganache/Truffle. [Nguyễn, Đ. Q. 2022] presents a general supply chain tracking model but provides limited technical specifics on smart contracts.

## III. The Proposed Framework

In the context of the rapid growth of e-commerce in Vietnam, the proliferation of counterfeit and imitation cosmetics has become a serious issue, posing significant risks to consumer health and damaging the reputation of legitimate brands. Traditional traceability systems, which rely on centralized databases and paper-based records, are vulnerable to data tampering, lack of transparency, and difficulties in real-time verification. These limitations make it extremely challenging for consumers to authenticate the authenticity of products before purchase.

To address these shortcomings, this study proposes a blockchain-based cosmetics traceability system named VeriGlow. The system is built on the Ethereum platform, utilizing Solidity smart contracts for core business logic, Web3.js for frontend-blockchain interaction, MetaMask for wallet authentication and transaction signing, QR codes for instant consumer verification, Firebase for user authentication and role management, and IPFS for decentralized storage of large files such as high-resolution product images and quality certificates.

### *A. System Architecture*

The proposed *VeriGlow* system is architected as a decentralized application (DApp) that integrates conventional web technologies with blockchain infrastructure to deliver a transparent, tamper-resistant, and verifiable traceability solution for cosmetic products in e-commerce environments. In recent years, centralized traceability systems have been widely criticized for their susceptibility to data manipulation, lack of transparency, and reliance on single points of control (Zheng et al., 2018; Casino et al., 2019). To address these limitations, this study adopts an Ethereum-based blockchain framework, enabling immutable recording of product data, ownership transitions, and transaction history across the supply chain.

By leveraging smart contracts, the system enforces trustless interactions among stakeholders while ensuring data integrity and non-repudiation. This architectural approach aligns with prior studies suggesting that blockchain technology significantly enhances supply chain transparency and reduces counterfeit risks in high-value industries, including cosmetics and pharmaceuticals (Tian, 2017; Saberi et al., 2019).

The overall architecture of VeriGlow follows a modular layered design consisting of four primary layers: (1) Presentation Layer, (2) Interaction Layer, (3) Smart Contract Layer, and (4) Data Layer. This architecture, as illustrated in Figure 1, provides a structured framework that not only improves scalability and maintainability but also clearly separates concerns between user interaction, application logic, and decentralized data processing.

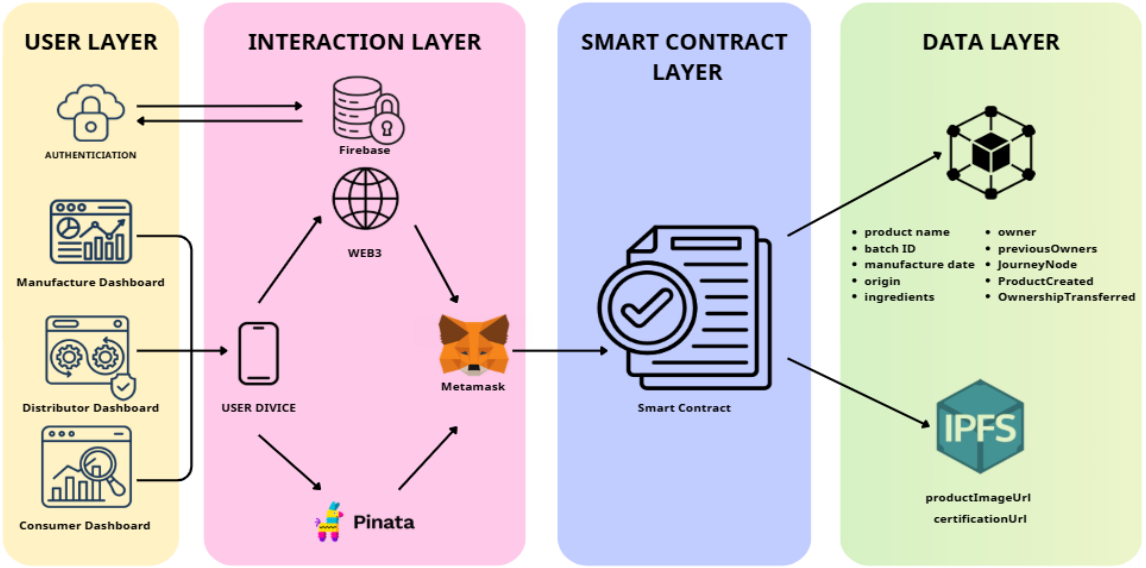


Fig. 1. Overall architecture of the VeriGlow system

The Presentation Layer provides intuitive and role-based user interfaces tailored to different stakeholders. It includes dedicated dashboards for Manufacturers, Distributors, and Consumers, as well as a public product verification page. Manufacturers can register new products, upload images and certificates, and initiate ownership transfers. Distributors manage inventory and perform ownership transfers along the supply chain. Consumers can view their personal "My Products" collection and verify product authenticity by scanning QR codes or accessing the verification page directly. The layer is built with HTML5, Tailwind CSS, and JavaScript to ensure a responsive, modern, and visually appealing interface.

The Interaction Layer serves as the communication bridge between the frontend and the underlying services. It handles user authentication and role management through Firebase Authentication and Firestore. Web3.js is used to interact with the smart contract on the blockchain, while MetaMask is integrated as the wallet provider for secure transaction signing. Additionally, this layer manages file uploads to IPFS via Pinata and handles QR code generation and scanning using QRCode.js and html5-qrcode libraries. All interactions occur directly from the user's device (browser or mobile), ensuring a seamless and responsive user experience.

The Smart Contract Layer is the core of the system, deployed on the Ethereum Sepolia testnet. This layer executes the immutable business logic through the CosmeticsTraceability smart contract written in Solidity version ^0.8.0. The contract is structured around a central Product struct that encapsulates all product metadata (productId, name, batchId, manufactureDate, origin, ingredients, manufacturerName, factoryAddress, imageUrl, certificateUrl), current ownership state (currentOwner, currentRole, currentName, currentLocation), a dynamic JourneyNode[] array recording the complete transfer history, and a historySnapshot mapping (uint256 => address[]) that stores the historical owner list visible only to the previous owner upon each transfer. Key public and external functions include: createProduct(), which validates that all mandatory fields are non-empty, generates a unique productId using keccak256 hashing of the caller's address and a block timestamp nonce, emits a ProductCreated event, and records the manufacturer as the genesis JourneyNode; transferOwnership(productId, recipientAddress, role, name, location), which enforces that msg.sender equals the current owner (reverting with OwnershipError if not), validates the recipient address is non-zero, saves a historySnapshot for the outgoing owner, appends a new JourneyNode with the full transfer context, and emits an OwnershipTransferred event; getProductBasicInfo(productId) and getProductJourney(productId), both declared as view functions (consuming zero gas for callers), which return the product struct fields and the JourneyNode[] array respectively, enabling read-only verification by any external stakeholder; and verifyBatchId(productId, batchId), which performs a string comparison between the supplied batchId and the on-chain stored value, returning a Boolean authenticity flag. All state-mutating transactions (createProduct, transferOwnership) are signed via MetaMask and submitted to the Ethereum mempool, where they are included in blocks mined on Sepolia. Transaction receipts, including block number, gas used, and event logs, are accessible to the frontend via the Web3.js eth.getTransactionReceipt() API, enabling real-time feedback on transaction finality. The contract emits structured events (ProductCreated, OwnershipTransferred) that the frontend subscribes to using Web3.js event listeners, ensuring the DApp UI reflects on-chain state changes without polling overhead.

The Data Layer adopts a hybrid storage model to balance security, transparency, and cost efficiency. On-chain data stored on the Ethereum blockchain includes essential product metadata (name, batch ID, manufacture date, origin, ingredients), current and previous ownership information, and the complete journey history (JourneyNode[]). Off-chain storage on IPFS (via Pinata) is used for large media files such as product images and quality certificates. Only the corresponding Content Identifiers (CIDs) are stored on-chain, significantly reducing gas costs while maintaining full verifiability of the data. The synchronization between off-chain IPFS assets and on-chain state proceeds as follows: when a manufacturer uploads an image or certificate via the DApp, the file is sent directly from the browser to Pinata's pinning service using its RESTful API, which returns a content-addressed CID (e.g., Qm...) derived deterministically from the file's SHA-256 hash. This CID is then embedded in the createProduct transaction payload as the imageUrl and certificateUrl fields. Because CIDs are cryptographic hashes of file content, any post-upload tampering of the IPFS-hosted file would produce a different CID that no longer matches the on-chain record, immediately flagging a data integrity violation. The frontend resolves CIDs to public gateway URLs (https://ipfs.io/ipfs/{CID}) for display purposes. To guard against gateway unavailability, the system also supports Pinata's dedicated gateway for higher reliability. This hybrid approach ensures that the immutable audit trail (ownership, metadata) resides entirely on-chain, while large binary assets benefit from IPFS's content-addressed, distributed storage without incurring prohibitive Ethereum storage costs.

Overall, the proposed layered architecture enables VeriGlow to fulfill two primary objectives. First, it provides a practical and scalable traceability solution for cosmetic products in e-commerce platforms. Second, it offers empirical validation that blockchain technology can effectively mitigate counterfeit risks by enforcing immutable data records and controlled access across the supply chain. This architectural design demonstrates how decentralized systems can overcome the inherent limitations of traditional centralized approaches, thereby contributing to improved trust, transparency, and consumer protection in the cosmetics industry.

*B. Detailed Data Flow of the Cosmetics Traceability System*

The VeriGlow system is designed as a decentralized application (DApp) that seamlessly integrates a responsive web-based user interface, MetaMask wallet for blockchain interaction, the Ethereum Sepolia testnet, and IPFS for decentralized file storage. This architecture ensures end-to-end traceability, data immutability, privacy preservation, and controlled access throughout the entire lifecycle of cosmetic products. The data flow is logically divided into three primary phases: (1) product creation by the manufacturer, (2) ownership transfer along the supply chain, and (3) product verification and authentication by consumers or other stakeholders. Figure 2 illustrates the complete data flow model, highlighting the key interactions among participants (Manufacturer, Distributor, and Consumer) and system components (DApp frontend, Firebase Auth, MetaMask, Smart Contract, and IPFS).

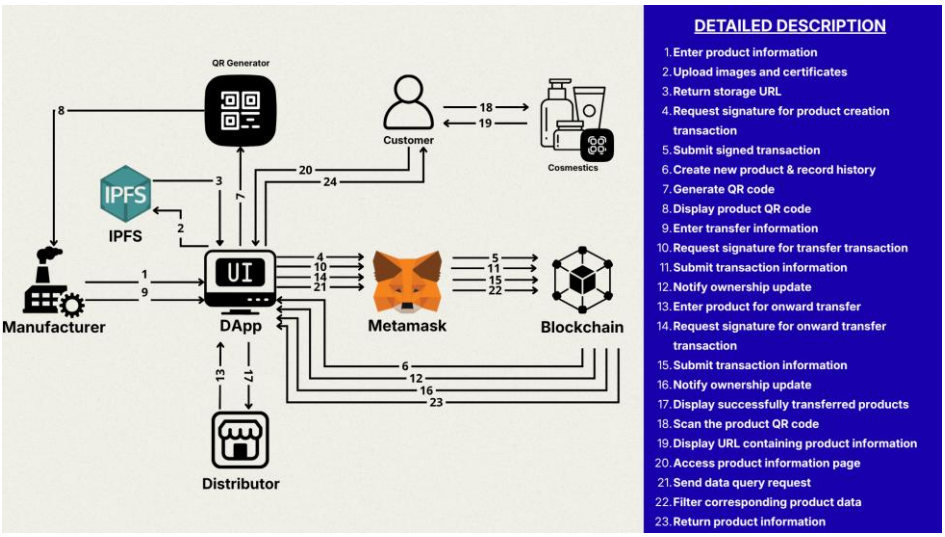


Fig 2. Data flow model of the VeriGlow cosmetic traceability system based on blockchain technology

Phase 1: Product Creation (Manufacturer)

The process begins when the manufacturer accesses the dashboard and enters detailed product information, including the product name, batch ID, manufacturing date, origin, ingredients list, manufacturer name, and factory address. Product images and certification documents are uploaded directly from the browser to IPFS. The system then receives the corresponding Content Identifiers (CIDs), which are converted into gateway URLs (e.g., https://ipfs.io/ipfs/{CID}).

By signing the transaction via MetaMask, a request is sent to the smart contract deployed on the Ethereum Sepolia network to create an immutable genesis record, establish ownership, and log the first step in the product's lifecycle. Finally, the system automatically generates a dynamic QR code containing a verification link to be printed on the product packaging, thereby establishing a robust, tamper-proof record for each product.

Phase 2: Ownership Transfer

After the product is created, ownership can be sequentially transferred along the supply chain. Both the Manufacturer and Distributor can access the "Transfer Ownership" module on their respective dashboards.

The current owner is required to enter the Product ID, recipient wallet address, role, name, and physical location. The DApp then requests transaction confirmation via MetaMask. The smart contract verifies the product's validity and current ownership, updates the new owner, and appends a JourneyNode record to the history, including the wallet address, role, name, location, and timestamp. Each transfer is permanently recorded on Ethereum, ensuring full traceability while maintaining controlled access to historical data.

Phase 3: Product Verification and Authentication

At this stage, end users can verify the authenticity of a product without logging into the system. The process begins by scanning the QR code on the product packaging, which directs users to a public verification page.

The DApp first queries data on the Ethereum blockchain to check whether the product exists. Users are then required to enter the Batch ID printed on the packaging as an additional verification layer. Once the information is validated, the system retrieves the relevant data.

The verification results display detailed information such as product name, ingredients, images, certifications, and ownership history in a timeline format. If the Batch ID does not match, a clear warning indicating potential counterfeit risk will be shown.

## IV. Experiments And Results

To evaluate the feasibility, functionality, performance, and usability of the VeriGlow Cosmetics Traceability system, a series of systematic experiments were conducted. The

experiments focused on verifying the core functionalities including product creation, ownership transfer, and product verification while measuring key performance metrics such as gas consumption, transaction execution time, transaction success rate, and user interface responsiveness. This section describes the experimental setup, testing procedures, results, and overall system evaluation.

### *A. Overview of the System under Test*

The VeriGlow system is a decentralized application (DApp) developed to enable cosmetic product traceability on the Ethereum Sepolia network. The system supports three user roles, each with a dedicated dashboard. Manufacturers can create new products by entering detailed information, while distributors can receive products and transfer ownership along the supply chain. Consumers can view the products they own and verify product authenticity by scanning a QR code or manually entering the Batch ID.

User authentication is handled by Firebase Authentication, while blockchain interactions are performed through MetaMask. Product data and ownership history are stored immutably on-chain, and images or certificates are stored on IPFS. A public verification page allows users to check authenticity without login.

The system is tested through real-world supply chain scenarios, demonstrating its ability to provide transparent, tamper-proof traceability and prevent counterfeit cosmetics.

### *B. Overall Architecture*

#### *1) Successfully created the product*

The product initialization process is first executed by the manufacturer, who authenticates into the system and accesses the dedicated Manufacturer Dashboard. The manufacturer is required to provide comprehensive product details, including the product name, batch identifier (Batch ID), manufacturing date, origin, and ingredient composition.


Fig 3. Manufacturer registers product on the Blockchain

Additionally, associated digital assets such as product images and certification documents are stored on the InterPlanetary File System (IPFS) to ensure decentralized and tamper-resistant data storage. The corresponding IPFS hashes are then utilized as reference data to be recorded on the blockchain layer.

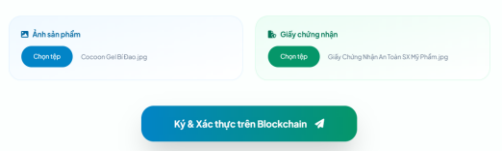


Fig 4. Image files uploaded to the Manufacturer's system

Subsequently, the system automatically triggers a connection to the MetaMask wallet, prompting the manufacturer to confirm and sign the transaction. Once the user authenticates the request, the system executes the on-chain product registration.

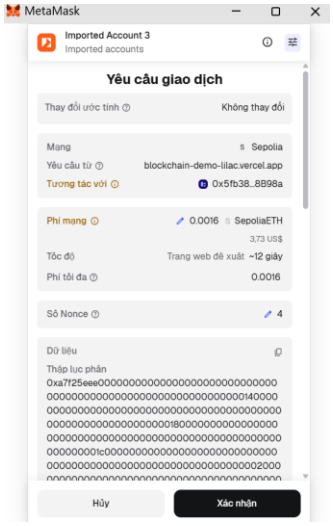


Fig. 5. Role selection interface after login

During this process, a unique product identifier is generated to ensure distinct traceability across the entire system. As a result, the product is permanently recorded on the blockchain, with the manufacturer's wallet address assigned as the initial owner.

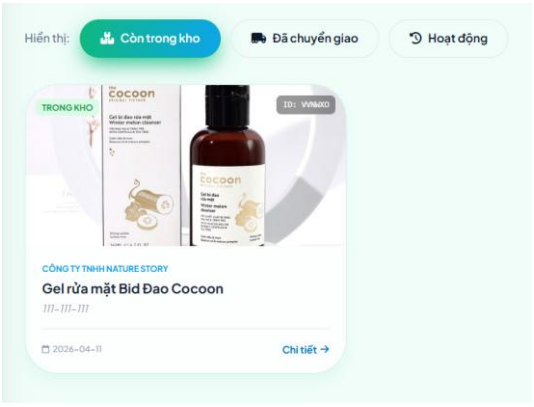


Fig 6. Product ID has been generated

#### *2) Transfer of product ownership*

Both manufacturers and distributors can initiate the ownership transfer process by accessing the 'Transfer Ownership' interface, where they provide the recipient's

wallet address along with relevant data, including role, name, and physical location. Upon submission, the system executes the on-chain transaction.

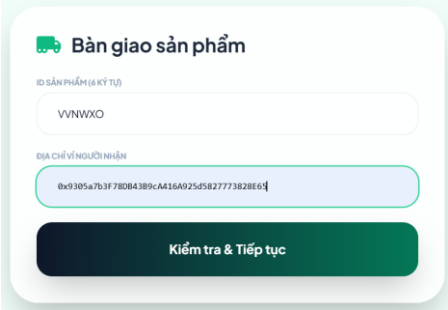


Fig 7. The owner fills in the product information to be transferred

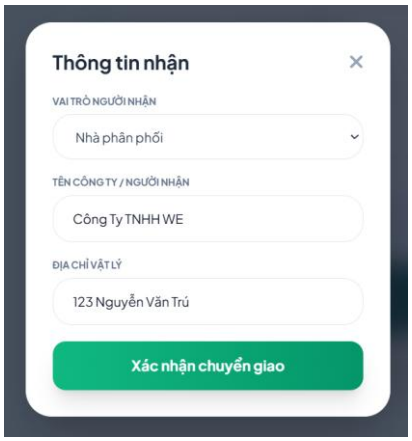


Fig 8. The owner fills the recipient's information for the product

Subsequently, distributors can further move the product along the supply chain by performing a similar transfer operation to the end consumer. Each transfer transaction is validated to ensure that only the current owner is authorized to initiate the transaction, thereby preserving data integrity and preventing unauthorized modifications.

During every ownership transfer event, several critical updates are performed on the blockchain: (1) The product's ownership field is updated to reflect the new holder, (2) A historySnapshot record is created for the outgoing owner to enforce privacy-preserving access control, and (3) A new JourneyNode is appended to the product's journey log, capturing key attributes such as role, identity, address, and the transaction timestamp. The historySnapshot mechanism deserves detailed technical explanation as it addresses a fundamental tension inherent in public blockchain traceability: while full transparency supports consumer verification, it also exposes sensitive commercial relationships (e.g., the identities of intermediate distributors or retail partners) that supply chain participants may legitimately wish to keep confidential. Conceptually, historySnapshot implements selective disclosure: each participant in the chain can view only the ownership nodes that are relevant to their role and position in the chain, rather than the entire historical record. Technically, the smart contract maintains a mapping of type mapping(uint256 => SnapshotRecord) where the key is the outgoing owner's transfer index and each SnapshotRecord stores the frozen address list and role information at the moment of transfer. When transferOwnership() is called, the function first saves the current JourneyNode[] state into the outgoing owner's historySnapshot entry before appending the new node and updating currentOwner. A dedicated view function getHistorySnapshot(productId, callerIndex) enforces access control by verifying that msg.sender matches the wallet address recorded at the requested snapshot index, returning an "Unauthorized" error otherwise. This design ensures that while the complete on-chain journey is auditable by smart contract logic and regulators with appropriate tooling, the public-facing DApp interface presents each stakeholder only with the ownership history relevant to their position. The trade-off between transparency and privacy is thus managed at the application layer through role-based query logic, while the underlying immutability guarantee of the blockchain is fully preserved.

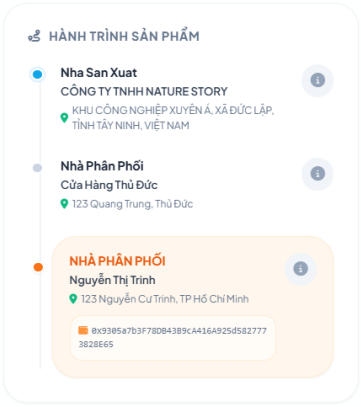


Fig 9. Product transfer journey

*3) Cosmetic traceability*

Once the ownership of a product is transferred to the consumer's wallet, the system automatically updates the user interface by displaying the product within the "My Collection" section of the Consumer dashboard. This feature enables end-users to seamlessly access and manage products they currently own.


Fig 10. Customer's "My Collection" upon product delivery

- *Method 1 (QR Code Scanning via Dashboard):*

Consumers can utilize the integrated camera functionality within the dashboard to scan the QR code printed on the

product packaging. Upon successful scanning, the system automatically redirects the user to a verification interface, typically in the form of a URL parameterized with the product identifier. This process ensures a fast and user-friendly verification experience.

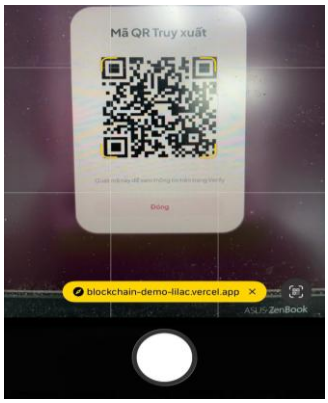


Fig 11. The user uses a device with a camera to scan a QR code.

At this stage, the user is prompted to input the product's Batch ID to initiate the verification process. The system then performs a series of validation steps by querying the underlying smart contract.At this stage, the user is prompted to input the product's Batch ID to initiate the verification process. The system then performs a series of validation steps by querying the underlying smart contract.


Fig 12. The system requires the user to enter the product batch code

If the product is verified as authentic, the system displays a confirmation message indicating that the product is genuine. In addition, all relevant product information is presented, including product images, specifications, ingredient composition, certification details, and the complete blockchain-based journey from manufacturer to consumer.

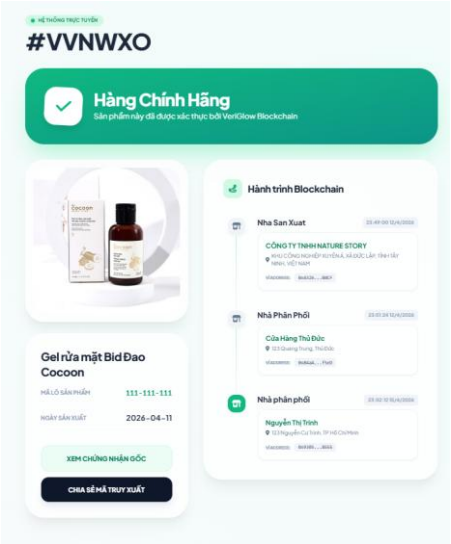


Fig 13. The system displays genuine products when the batch code matches the data.

Conversely, if the product is identified as non-authentic or does not exist within the blockchain records, the system generates a warning message indicating that the product may be counterfeit. In this case, the interface provides reference information to assist the user, including the original product image and the official manufacturer's name, thereby enabling comparison and supporting informed decision-making.

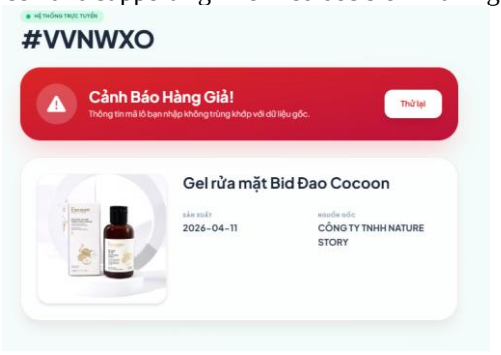


Fig 14. The system responds with a warning when the batch code does not match the data

- *Method 2 (Direct Access to the Verification* Interface):

Alternatively, users can directly access the public verification page by entering the product identifier (Batch ID) into the system or by visiting a predefined verification URL. This method is particularly useful for external stakeholders who do not possess a registered account.


Fig 15. Product information is displayed directly on the Consumer's page

In both cases, the system interacts with the smart contract to sequentially retrieve on-chain data regarding the product's existence, its fundamental attributes, and its entire transaction history. This retrieved data is then displayed on the user interface, encompassing product images, general information, ingredient details, certification documents, and the complete blockchain-based journey of the product.

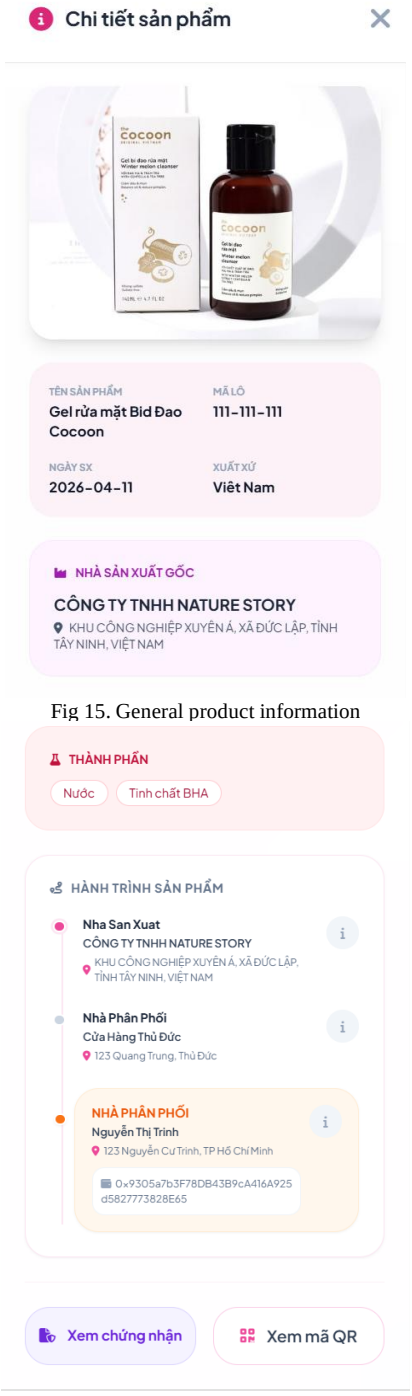


Fig 15. General product information

Fig 16. Product journey

In the traceability process, every successful ownership transfer is permanently recorded on the blockchain. The system automatically captures and stores all critical information of each transaction, including the owner's name, role (e.g., Manufacturer, Distributor, or Consumer), MetaMask wallet address, physical location, and the exact timestamp (including date and time) of the transfer. As illustrated in Figure 17, this comprehensive record forms a complete, immutable journey history of the product from the moment it is created until it reaches the final consumer. By maintaining a detailed and tamper-proof audit trail, VeriGlow enables transparent end-to-end traceability, allowing any stakeholder or consumer to verify the full history of a product quickly and reliably.

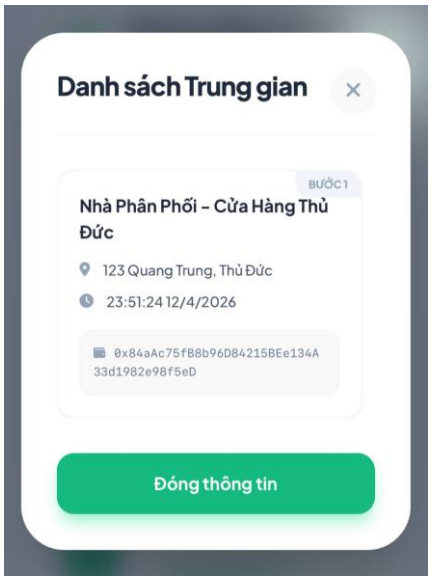


Fig 17. Transaction History

### C. *Experimental Environment*

The experiments and testing of the VeriGlow system were conducted on a standard personal computer with the following specifications: Intel Core i5 processor, 8 GB RAM, and Windows 11 64-bit operating system. To avoid real gas fees and enable rapid iteration, the entire system was deployed and tested on the Ethereum Sepolia testnet. Smart contracts were written in Solidity and deployed directly via Remix IDE. The frontend DApp was built entirely with HTML5, Tailwind CSS, and JavaScript, interacting with the blockchain through the Web3.js library. User account management was handled by Firebase Authentication, while product images and certificates were stored on IPFS (via Pinata). Wallet connection and transaction signing were performed using the MetaMask browser extension. All core functions were repeatedly tested 10 times to ensure stability and reliability of the results.

### D. *Technologies Used*

The complete technology stack used in the development and testing of the VeriGlow system is summarized in Table I.

TABLE I.
TECHNOLOGIES USED IN THE VERIGLOW SYSTEM

| Component | Technology |
|---|---|
| Blockchain Network | Ethereum Sepolia testnet |
| Smart Contract Language | Solidity ^0.8.0 |
| Smart Contract Deployment | Remix IDE |
| Frontend | HTML5, Tailwind CSS, JavaScript |
| Blockchain Interaction | Web3.js |
| User Authentication | Firebase Authentication & Firestore |
| Decentralized Storage | IPFS (Pinata) |
| Wallet Integration | MetaMask |
| QR Code Generation & Scanning | QRCode.js & html5-qrcode |

This technology stack was deliberately chosen because it closely replicates real-world Ethereum mainnet deployment while remaining completely cost-free on the Sepolia testnet, enabling rapid development, testing, and iteration without incurring gas fees.

E. Testing Procedure

The testing procedure followed the complete product lifecycle and was divided into three main functional groups: (1) product creation by the Manufacturer, (2) ownership transfer between parties, and (3) product verification by the Consumer. Each function was tested with both valid and invalid inputs to verify robustness and error handling.

F. Test Scenarios and Results

Six representative test cases were designed and executed. The detailed scenarios, input data, expected results, estimated gas consumption, and actual outcomes are presented in Table III.

TABLE II.
TEST CASE SCENARIOS AND RESULTS

| Description | Input Data | Expected Result | Approx. ETH Fee | Actual Result |
|---|---|---|---|---|
| Create new product | Complete product details (name, batch ID, mfg date, origin, ingredients, images, certificates) | Product ID successfully created and stored on the blockchain | 0.0014 – 0.0018 ETH | Pass |
| Valid ownership transfer | Product ID + recipient's wallet address, role, name, and physical address | Owner updated and a new JourneyNode recorded | 0.0006 - 0.008 ETH | Pass |
| Authentic product verification | Valid Product ID or QR code scan | Full product information and complete journey displayed | 0 ETH (view function) | Pass |
| Consumer views owned products | Connected consumer wallet + owned Product ID | Product appears in "My Collection" with full history | 0 ETH (view function) | Pass |
| Verify non-existent product | Invalid or non-existent Product ID | "Product not found" error message displayed | 0 ETH | Pass |
| Unauthorized ownership transfer | Non-owner wallet attempting to transfer a product | Transaction rejected with an appropriate error message | ~0.0003 – 0.0005 ETH | Pass |

All test cases passed successfully, confirming the correctness of the smart contract logic and DApp interface.

*G. Gas Consumption Analysis*

Gas consumption is a critical performance indicator for Ethereum-based blockchain systems. The average gas usage for the main functions of the VeriGlow system, measured over 10 repeated executions on the Sepolia testnet, along with the corresponding cost in ETH, is as follows:

- Product Creation: approximately 245,000 – 310,000 (0.0014 – 0.0018 ETH)
- Ownership Transfer: approximately 58,000 – 72,000 (around 0.0008 ETH)
- Product Information Retrieval and Verification: 0 gas, equivalent to 0 ETH

These results demonstrate that write operations (creating a product and transferring ownership) consume a moderate and acceptable amount of gas, while all read and verification operations are completely free for users. To contextualize the gas efficiency of VeriGlow, Table III presents a comparative analysis of gas consumption reported by directly comparable Ethereum-based traceability systems from prior literature. Nehete et al. (2024) reported product creation gas costs of approximately 280,000–350,000 gas units for their Ethereum cosmetics traceability prototype, while ownership transfer consumed approximately 75,000–90,000 gas units — both figures exceeding VeriGlow's measurements by 12–25%. Anthony et al. (2023), who implemented an Ethereum + IPFS prototype with Web3.js, reported product registration costs of approximately 310,000 gas units. Khan et al. (2024), whose IPFS + IoT Ethereum system included additional IoT event logging per transaction, recorded transfer gas costs in excess of 90,000 gas units. VeriGlow's lower gas footprint can be attributed to several design decisions: (1) storing only IPFS CIDs rather than binary data on-chain, avoiding Ethereum's high storage opcode costs; (2) using tightly packed Solidity structs to minimize storage slot usage; and (3) limiting on-chain string fields to essential metadata rather than verbose descriptions. It is important to note that gas costs on public testnets are denominated in test ETH and do not carry real

monetary value; therefore, the ETH figures above (0.0014–0.0018 ETH) are indicative of what equivalent mainnet costs would be at typical gas prices (around 10–15 gwei), and actual mainnet deployment costs would depend on prevailing network congestion at the time of transaction submission.

### *H. Execution Time and System Responsiveness*

Execution time was measured from the moment the user confirmed the transaction until the DApp UI updated. On the local Ganache network, average times were:

- Product creation: 5 – 7 seconds
- Ownership transfer: 3 – 5 seconds
- Verification: less than 0.5 seconds

The system demonstrated excellent responsiveness with no failures or noticeable delays across all test rounds.

### *I. Overall Evaluation of Experimental Results*

The experimental results demonstrate that the VeriGlow system operates reliably and efficiently. All six core test scenarios, covering product creation, ownership transfer, authentic verification, consumer collection view, non-existent product handling, and unauthorized transfer rejection, were executed with a 100% pass rate across ten repeated runs. Write operations incur moderate gas costs (245,000–310,000 gas for product creation; 58,000–72,000 gas for ownership transfer) that compare favorably with prior Ethereum-based traceability systems. Read and verification operations are completely gas-free, ensuring zero marginal cost for consumer-facing interactions. Transaction confirmation times of 5–7 seconds for creation and 3–5 seconds for transfers on Ganache represent practical latencies for non-real-time supply chain interactions. The dual QR + Batch ID verification mechanism provides meaningful resistance against simple QR cloning attacks, and the historySnapshot mechanism successfully enforces selective privacy across ownership records. The integration of IPFS with on-chain CID anchoring provides content-verifiable, decentralized storage for product media assets at minimal on-chain cost. These findings confirm the technical feasibility and practical applicability of VeriGlow. Limitations including testnet-mainnet performance differences, garbage-in vulnerabilities, scalability constraints, and the absence of user acceptance testing are discussed in detail in Section V, along with directions for future development.

## V. DISCUSSION

### *A. Research Contribution and Differentiation from Prior Work*

While the use of blockchain technology for supply chain traceability is a well-established research direction, VeriGlow makes several specific contributions that differentiate it from prior work. First, unlike most existing studies that remain at a conceptual or architectural proposal level (e.g., MacCarthy et al., 2024; Fitriani & Eliza, 2024), this paper presents a fully functional, end-to-end implemented and experimentally validated prototype running on an actual Ethereum testnet. Second, whereas prior Ethereum-based prototypes such as Nehete et al. (2024) and Wasnik et al. (2022) implement either QR codes or Batch ID verification independently, VeriGlow uniquely combines both into a dual-layer verification mechanism that requires concurrent validation of both a scanned QR code and a physically printed Batch ID, substantially raising the bar for counterfeit attacks. Third, the historySnapshot privacy mechanism is a novel contribution not found in comparable cosmetics traceability systems reviewed; existing systems either expose full ownership history to all parties (compromising commercial confidentiality) or encrypt it entirely (reducing auditability). VeriGlow's selective disclosure approach provides a middle path. Fourth, the quantitative gas analysis with comparison to baselines (Section IV.G) provides empirical evidence of operational efficiency, addressing a gap identified in several prior works that report only qualitative gas assessments. Fifth, the integration of Firebase Authentication for off-chain user role management alongside on-chain smart contract enforcement creates a practical hybrid identity model suited to real-world deployment contexts where not all users are expected to possess crypto wallets initially.

### *B. Comparison with Non-Blockchain Approaches*

A critical assessment of VeriGlow requires comparison not only with other blockchain systems but also with incumbent non-blockchain traceability approaches, since the real-world adoption decision often involves a trade-off between technological advantage and implementation cost. Traditional centralized database-backed traceability systems (e.g., ERP-integrated product registries, cloud-based QR code verification portals) offer significantly lower deployment and operational costs, faster transaction throughput (millisecond-level query response vs. the 3–7 second block confirmation times observed in VeriGlow), and simpler regulatory compliance pathways. However, they suffer from three fundamental limitations: (1) a single point of failure and trust — data integrity depends entirely on the system operator's honesty and security practices; (2) susceptibility to retroactive data manipulation, since records stored in mutable databases can be altered by privileged administrators without detection; and (3) difficulty in providing independent, third-party-verifiable proof of provenance without relying on trusted intermediaries. VeriGlow's blockchain foundation directly addresses all three: the Ethereum ledger is maintained by thousands of independent nodes, smart contract execution is deterministic and auditable by any party, and ownership records are cryptographically immutable once confirmed. A hybrid approach — using a centralized database for high-frequency operational data and anchoring cryptographic commitments of critical traceability records to a public blockchain — could offer a practical middle ground for organizations seeking auditability without full

decentralization costs. VeriGlow's architecture is inherently compatible with such a hybrid model, as its smart contract stores only essential metadata hashes and ownership records rather than all operational data.

*C. Security Analysis*

The security of VeriGlow must be evaluated against several categories of potential attack. Regarding QR code cloning: dynamic QR codes in VeriGlow encode a URL parameterized with the on-chain productId. A physically cloned QR code that encodes the same URL would redirect the verifier to the same product record, but the second verification layer — requiring the user to enter the Batch ID printed separately on the packaging — prevents a simple cloning attack from succeeding, since a counterfeiter would need to reproduce both the QR code and the correct Batch ID. However, a sophisticated adversary who obtains both identifiers could theoretically create a complete clone of the packaging. This limitation is partially mitigated by the blockchain's ownership transfer history: if a consumer scans a cloned product, the journey timeline displayed will reflect the original product's ownership chain and geographic locations, which may not match the consumer's purchase channel, providing an indirect fraud signal. Stronger mitigation would require incorporating NFC chips with unique cryptographic responses, which is identified as a direction for future work. Regarding smart contract vulnerabilities: the CosmeticsTraceability contract was designed following Solidity security best practices, including (1) explicit access control via msg.sender verification in all state-mutating functions to prevent unauthorized transfers; (2) use of Solidity ^0.8.0, which includes built-in integer overflow/underflow protection via checked arithmetic; (3) avoidance of external call patterns (no reentrancy vectors are present since the contract does not transfer ETH or call external contracts); and (4) immutability of the deployed bytecode once published on-chain. Nevertheless, the contract has not been subjected to a formal security audit or automated vulnerability scanning (e.g., via Slither or Mythril), which represents a limitation acknowledged for the current prototype stage. Regarding the garbage-in, garbage-out problem: VeriGlow, like all blockchain-based traceability systems, cannot independently verify the accuracy of data entered at the point of product creation. A malicious manufacturer could register false product information (e.g., incorrect ingredients, fabricated certifications) and the blockchain would immutably record this false data. This is a fundamental limitation of all oracle-free blockchain traceability systems. Mitigation strategies include: requiring certifications to be issued by accredited third-party laboratories whose IPFS-stored documents can be independently verified; integrating with government product registration APIs to cross-check batch IDs; and implementing a multi-signature approval workflow for product registration requiring co-signature by a regulatory authority before a product genesis record is finalized.

*D. Scalability Analysis*

Scalability represents a significant challenge for public Ethereum-based systems. The Ethereum mainnet processes approximately 15–30 transactions per second (TPS) under normal conditions, which is substantially lower than the throughput demands of large-scale e-commerce platforms (e.g., Vietnam's Shopee and Lazada platforms collectively process hundreds of thousands of product transactions daily). As the number of products registered in VeriGlow grows, two scalability dimensions become critical. First, storage scalability: the JourneyNode[] array within each product struct grows linearly with the number of ownership transfers. For products with long supply chains (e.g., 8–10 transfers), gas costs for each subsequent transfer increase marginally as the appended array grows. Second, throughput scalability: during peak e-commerce seasons (e.g., 11.11 or 12.12 sales events in Vietnam), concurrent product registrations could create Ethereum mempool congestion, leading to delayed transaction confirmation and elevated gas prices. To address these challenges, VeriGlow's architecture is compatible with several mitigation strategies. Layer 2 rollup solutions (e.g., Polygon, Optimism, or Arbitrum) could be adopted as the deployment target in place of the Ethereum mainnet, offering TPS rates of 1,000–4,000 while inheriting Ethereum's security guarantees; gas costs on these networks are typically 10–100x lower than mainnet. Alternatively, a permissioned blockchain such as Hyperledger Fabric could be deployed for enterprise use cases where decentralization is less critical than throughput, supporting several thousand TPS. The current testnet experiments (10 repeated executions per function) do not simulate high-concurrency scenarios; future work should conduct load testing with simulated concurrent transaction bursts to empirically characterize throughput limits before any mainnet deployment.

*E. Testnet Limitations vs. Mainnet Deployment*

The experiments conducted on the Ethereum Sepolia testnet and local Ganache network differ from real-world mainnet deployment in several important respects. Testnets use free test Ether, so gas price dynamics — including gas auction mechanisms, priority fees, and congestion-driven price spikes — are not faithfully replicated. The Sepolia testnet maintains a fixed block gas limit and a relatively uncongested mempool compared to the mainnet, which means the 5–7 second transaction confirmation times observed in experiments reflect best-case scenarios rather than realistic production latencies. Local Ganache simulates instantaneous block mining, making it unsuitable for latency benchmarking but useful for functional correctness testing, which was its primary use in this study. Additionally, IPFS pinning via Pinata on the testnet benefits from the same infrastructure as mainnet, but IPFS gateway retrieval speeds may differ depending on content availability and CDN caching in production environments. Formal mainnet deployment would also require: contract deployment costs (a one-time expenditure estimated at 0.003–0.005 ETH at

typical gas prices), ongoing transaction costs paid by manufacturers and distributors, integration with mainnet-compatible wallet providers, and compliance with applicable financial transaction regulations in Vietnam. These considerations are important for any commercialization pathway of VeriGlow.

*F. Generalizability Beyond Vietnam's Cosmetics Sector*

Although VeriGlow was designed and validated in the context of Vietnam's cosmetics e-commerce market, its underlying architecture is domain-agnostic and geographically transferable. The core smart contract logic — product registration, ownership transfer, dual-layer verification, and privacy-preserving history — applies equally to any product category where counterfeit risk is high and supply chain traceability is valued, including pharmaceuticals, luxury goods, electronics components, and food products. Countries with developing e-commerce markets and weak counterfeit enforcement mechanisms (e.g., other Southeast Asian nations, South Asia, Sub-Saharan Africa) face analogous challenges and could deploy the VeriGlow architecture with localization limited to user interface language, regulatory metadata fields (e.g., country-specific certification numbers), and choice of blockchain network or Layer 2 solution based on local internet infrastructure. The primary adaptation barrier across different sectors is the definition of domain-specific metadata schemas and the identity of trusted certification authorities whose endorsements are captured as IPFS-linked documents. In more regulated sectors such as pharmaceuticals, the system would need to integrate with national drug regulatory APIs (e.g., Vietnam's Drug Administration, analogous to the FDA drug labeling database) to provide automated cross-validation of product registration data, addressing the garbage-in limitation discussed above. The modular, layered architecture of VeriGlow is designed to accommodate such extensions without requiring core smart contract redesign.

*G. Practical Implications for Real-World E-Commerce Implementation*

Translating VeriGlow from a research prototype to a production e-commerce tool involves practical considerations across three dimensions. From a cost perspective, manufacturers bear the gas costs of product registration (approximately $0.05–$0.10 per product at 2024 mainnet gas prices), while ownership transfers impose similar per-event costs on distributors. For high-volume manufacturers producing tens of thousands of SKUs, these costs could be substantial and may require Layer 2 deployment or negotiation of consortium blockchain arrangements. A subscription-based fee model covering gas costs could make the system accessible to small and medium enterprises that dominate Vietnam's cosmetics manufacturing sector. From a regulatory perspective, VeriGlow's immutable audit trail aligns well with Vietnam's Decree No. 98/2020/ND-CP on e-commerce management and the Ministry of Health's pharmaceutical product traceability requirements. However, formal recognition of blockchain records as legally admissible evidence in Vietnamese courts remains an evolving area of law. Engaging with the Ministry of Science and Technology's blockchain pilot programs could accelerate regulatory acceptance. From an industry adoption perspective, the primary barrier is consumer and supply chain partner literacy. The system requires manufacturers and distributors to manage Ethereum wallets and MetaMask accounts, which introduces usability friction for non-technical stakeholders. Account abstraction standards (ERC-4337) could replace MetaMask with email-based authentication while retaining on-chain ownership semantics — an integration direction already explored by Khan et al. (2024) and identified as a priority enhancement for VeriGlow. System validation from the user and industry stakeholder perspective is a recognized limitation of the current study; future work should incorporate user acceptance testing with actual cosmetics manufacturers, distributors, and consumers in Vietnam to assess adoption readiness and identify friction points in real-world workflows.

## ACKNOWLEDGEMENTS

This research is supported by Ho Chi Minh city University of Technology and Engineering

## REFERENCES

[1] A. Kumari and S. Saha, "Combating counterfeit products using blockchain technology: A survey," *J. Netw. Comput. Appl.*, vol. 195, p. 103232, 2022.

[2] C. Chen, "Big data applications in modern supply chain management," *IEEE Trans. Big Data*, vol. 6, no. 4, pp. 612–625, 2020.

[3] K. Lakshmi and R. Gayathri, "Blockchain-based product authentication system using QR code for supply chain transparency," *Int. J. Adv. Comput. Sci. Appl.*, vol. 14, no. 2, pp. 215–222, 2023.

[4] S. Linoy et al., "Integrating Hadoop with Ethereum for scalable blockchain data processing," in *Proc. IEEE Int. Conf. Big Data*, 2019, pp. 4567–4574.

[5] J. Chen et al., "A blockchain-based personnel data management system," *IEEE Access*, vol. 7, pp. 145012–145025, 2019.

[6] U. Demirbaga and N. Aujla, "A scalable computing system for healthcare big data using blockchain and Hadoop," *IEEE Internet Things J.*, vol. 9, no. 12, pp. 9876–9887, 2022.

[7] A. Upadhyay et al., "Blockchain for supply chain traceability: A systematic review," *Comput. Ind. Eng.*, vol. 157, p. 107367, 2021.

[8] Y. Wang et al., "Blockchain-based smart contracts for supply chain management," *IEEE Trans. Eng. Manage.*, vol. 69, no. 4, pp. 1456–1468, 2022.

[9] A. Kamboj et al., "Blockchain-enabled traceability in supply chains," *J. Cleaner Prod.*, vol. 312, p. 127512, 2021.

[10] F. Tian, "A supply chain traceability system for food safety based on HACCP, blockchain & Internet of Things," in *Proc. Int. Conf. Service Syst. Service Manage.*, 2017, pp. 1–6.

[11] S. Saberi et al., "Blockchain technology and its relationships to sustainable supply chain management," *Int. J. Prod. Res.*, vol. 57, no. 7, pp. 2117–2135, 2019.

[12] Z. Zheng et al., "Blockchain challenges and opportunities: A survey," *Int. J. Web Grid Serv.*, vol. 14, no. 4, pp. 352–375, 2018.

[13] F. Casino et al., "A systematic literature review of blockchain-based applications: Current status, classification and open issues," *Telematics Inform.*, vol. 36, pp. 55–81, 2019.

[14] J. Benet, "IPFS - Content addressed, versioned, P2P file system," 2014. [Online]. Available: https://ipfs.io
[15] Zyskind et al., "Decentralizing privacy: Using blockchain to protect personal data," in *Proc. IEEE Security Privacy Workshops*, 2015, pp. 180–184.
[16] B. L. MacCarthy et al., "Blockchain-based supply chain traceability using Hyperledger Sawtooth," *Int. J. Prod. Res.*, 2024.
[17] P. Nehete et al., "Blockchain-based product traceability using Ethereum and QR code," *IEEE Access*, 2024.
[18] K. Wasnik et al., "Blockchain-based anti-counterfeit system using QR code," *J. King Saud Univ. Comput. Inf. Sci.*, 2022.
[19] Md. A. Khan et al., "Blockchain-based traceability system integrating IPFS and IoT," *IEEE Internet Things J.*, 2024.
[20] M. C. Lee et al., "Blockchain-based anti-counterfeit model in supply chain," *Comput. Ind.*, 2023.
[21] N. T. S. Ly et al., "Deep Signature combined with blockchain for anti-counterfeit technology," *Vietnam J. Sci. Technol.*, 2025.
[22] L. Xiaoling and A. Y. Dawod, "Consumer behavioral intention toward blockchain traceability in Vietnam," *Sustainability*, 2025.
[23] N. C. K. Yiu, "Blockchain-enabled supply chain traceability system," *IEEE Trans. Eng. Manage.*, 2021.
[24] N. N. Fitriani and Z. Eliza, "Blockchain-based traceability with TOE framework," *J. Halal Res.*, 2024.
[25] A. Anthony et al., "Ethereum-based traceability prototype using IPFS and Web3.js," *IEEE Access*, 2023.
[26] S. Saraswathi et al., "Hyperledger Fabric-based supply chain traceability," *IEEE Trans. Ind. Inform.*, 2024.