Nigerian Journal of Technology (NIJOTECH)
Vol. 44, No. 2, June, 2025, pp.282 - 292
www.nijotech.com

Print ISSN: 0331-8443
Electronic ISSN: 2467-8821
https://doi.org/10.4314/njt.v44i2.12

DEVELOPMENT OF A BLOCKCHAIN-BASED ANTI-COUNTERFEITING SYSTEM LEVERAGING PRODUCT INHERENT FEATURES AND LOCATION INFORMATION

AUTHORS:
J. Wosu[1], G. Chukwudebe[2], L. Ezema[3], C. Agubor[4], E. B. Mfonobong[5,*], and M. E. Nwanga[6]

AFFILIATIONS:
[1,2,3,4,5]Department of Electrical and Electronic Engineering, Federal University of Technology Owerri, Imo State, Nigeria.
[6]Department of Information Technology, Federal University of Technology Owerri, Imo State Nigeria

***CORRESPONDING AUTHOR:**
Email: benson.mfonobong@futo.edu.ng

ARTICLE HISTORY:
Received: 06 February, 2025.
Revised: 04 June, 2025.
Accepted: 05 June, 2025.
Published: 07 July, 2025.

KEYWORDS:
Blockchain Technology, Consensus Algorithm, Counterfeiting, Distributed Applications, Inherent Product Features, QR Code.

ARTICLE INCLUDES:
Peer review

DATA AVAILABILITY:
On request from author(s)

EDITORS:
Ozoemena Anthony Ani

FUNDING:
None

Abstract

The proliferation of counterfeit items has hurt the economic growth, public health, and safety. This work aims to develop an innovative system that can counter and mitigate the threat posed by local and global counterfeiters whose activities have caused untold health and economic hardship to society. This paper proposes a novel blockchain-based anti-counterfeiting system that makes use of a product's unique characteristics and its geographical location. Prototype system modelling in this study was accomplished using object-oriented software analysis and design techniques, Rapid Unified Process (RUP) and, Rapid Application Development (RAD) methodologies for QR Code and Blockchain applications respectively. Ganache, a private Ethereum blockchain network, was set up to serve as the backend platform. Open-source software such as the Truffe suite and the Solidity compiler were utilised in setting up the Ganache network as well as in compiling and deploying smart contracts written in Solidity. Results proved that the system, when tested on 50 products, shows low energy consumption, high speed of execution at 38.4s on average, QR code scanning time of 9.5ms on average, very high data integrity, and 100% accuracy record when validating whether or not a product is a counterfeit. This work provides a solution for cost-effective and comprehensive anti-counterfeiting measures, featuring key elements such as traceability, immutability, and transparency. The developed system is unparalleled as it combines blockchain technology, unique product inherent features, location information (GPS coordinates), and Track and Trace technologies, to offer a reliable and secure solution to counterfeit trading. This work, therefore, represents a potentially innovative approach to curbing the proliferation of counterfeit products.

1.0 INTRODUCTION

Counterfeiting remains a significant challenge in both local and global commercial systems, particularly in the food and pharmaceutical industries. Not only do illegal trades in counterfeit goods harm businesses' profits, but they also support criminal organizations at the expense of legitimate businesses and governments [1]. Counterfeit products significantly impact community safety and the economy, especially in the case of food and medicine. In developing countries, counterfeit drugs are responsible for many deaths, with the counterfeit drug market growing twice as fast as legal medications and now making up 2.5% of the global pharmaceutical market [2]. According to the International Chamber of Commerce, counterfeiting and piracy are expected to threaten 5.4 million legal jobs and cost the global economy $4.2 trillion (USD)

HOW TO CITE:
Wosu, J., Chukwudebe, G., Ezema, L., Agubor, C., Mfonobong, E. B., and Nwanga, M. E. "Development of a Blockchain-Based Anti-Counterfeiting System Leveraging Product Inherent Features and Location Information", *Nigerian Journal of Technology*, 2025; 44(2), pp. 282 – 292; https://doi.org/10.4314/njt.v44i2.12

by 2025 [3]. In today's fast-paced global market, the ease of buying and selling counterfeit goods is on the rise. Legal trade facilitation tools, such as free trade zones, are often misused due to lax regulatory frameworks and insufficient advanced tools to combat counterfeiting [4]. These illegal activities are harmful to consumers and businesses, making it crucial for organizations to work towards ending counterfeiting. Anti-counterfeit solutions are being developed, utilizing advanced technologies. These solutions protect businesses and consumers from financial losses, ensuring ease of use, with difficulty in replication, and effective implementation, as noted in [5]. However, current systems are largely centralized and face challenges such as data redundancy and processing delays [6][7].

Blockchain enables quick, secure, and cost-effective transactions across a decentralized network of computers, reducing the risk of data loss in the event of system failures [8]. The features of blockchain ensure that transactions remain transparent and immutable, making it a strong tool for combating counterfeiting [5][9-12]. This study advocates for an innovative blockchain-based solution to track counterfeiting through a Quick Response Code Generator (QRCG), leveraging location data and product attributes. The prototype system will utilize blockchain technology and a decentralise application (dApp), the system will use Ethereum for its flexibility in storage and efficient block generation.

Further, the research in [13] discusses a Byzantine fault-tolerant consensus mechanism for sharded blockchain networks, allowing for the certification of proposed blocks and completing large blocks in about 10 seconds without needing costly leader-driven communication. A study in [14] proposed that blockchain-based crowdsourcing services benefit from an enhanced Proof-of-Trust (PoT) consensus system, improving trust and reliability in decentralized environments. A new consensus mechanism, Mixed Byzantine Fault Tolerance (MBFT), introduced in [15], enhances scalability and efficiency while improving security by partitioning nodes involved in consensus and incorporating credit mechanisms for added resilience.

Research on Smart Contract Languages (SCL) in [16] aimed at identifying essential qualities for legally binding contracts, using a Systematic Literature Review (SLR) of works published between 2015 and 2019. In [17], 29 common issues in smart contracts (such as code visibility and updateability) were identified, and 60 potential solutions (like off-ledger computations and gas limit definitions) were proposed. The study in [18] introduced MOVO, a decentralized application (dApp) for intelligent transportation that operates without a central server. This dApp collects context-based data and shares it via Distributed Ledger Technologies (DLT), showcasing the practical use of decentralized systems.

Sungari, as presented in [19] is an automated framework for testing dApps and association between dApp's frontend and blockchain. [20] presented an approach to solving the person re-identification (ReID) problem by improving the accuracy and efficiency of individual code generation. In [21], Blockchain-based systems that allow consumers to identify a product's origin (manufacturer) was presented. While in [22], a blockchain-based anti-counterfeiting solution was presented. It enables producers to sell directly to consumers without the need for retail outlets. Ethereum's Solidity language powers the smart contracts [23] in these systems, with tools like Ganache and Truffle used for testing and interaction.

The research gap in existing anti-counterfeiting systems is the lack of integration between product features and location (GPS data) for generating QR codes. This research addresses this gap by introducing a novel QR code generator that incorporates product characteristics, manufacturing details, and GPS coordinates using an object-oriented system modelling approach. To further strengthen security, Copy-Sensitive Graphical Codes (CSGC) was embedded into the QR codes, with blockchain technology providing an additional layer of protection.

2.0 METHODS

In this work, a composite methodology has been adopted that involves object-oriented system analysis and design with the integration of Unified Model Language (UML) patterns. Blockchain development technique adopts the Rapid Application Development (RAD) approach.

In adopting an object-oriented system analysis and design approach for modelling the proposed blockchain-based product anti-counterfeiting system, the Modelio UML tool was utilized. In so doing, a user-scenario case study was captured for the system, which gave rise to the system architecture, functional requirements, and use cases upon which the entire solution was implemented, as detailed in the subsections below.

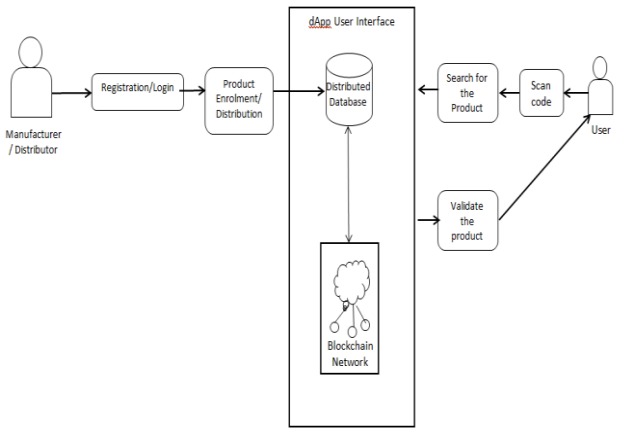

Figure 1: Proposed system architecture

2.1 Modelling of the Proposed System

System Description (User scenario). The blockchain-based anti-counterfeiting system's execution is in the following four stages:

i. Enrolment of manufacturers and distributors on the network: The first stage in system execution is to bring all manufacturers and distributors into the blockchain network. Manufacturer and distributors' authentication is done via registration. In so doing, each manufacturer and distributor will furnish the system with their business name, contact information (e.g., phone number(s), email address, and location address), location information (i.e., Global Positioning System (GPS) coordinates), etc for proper identification and login access.

ii. Product enrolment on the network: A manufacturer will request to add a product to a network, with the product being enrolled if the requester is verified as a legitimate manufacturer. During enrolment, the product's texture is captured as an image, then converted into text. Key details like the manufacturing date, expiration date, serial number, and batch number are recorded. The texture and product information are combined to create a unique QR code (refer to the journal article *"Design of a Novel Quick Response Code Generator and Scanning System for Counterfeit Product Detection"* for more details). After registration, a smart contract and unique code are generated, securely storing the product's QR code details in encrypted form.

iii. The end user gets details about the product: In this final stage, end users will scan the unique QR code embedded on the product using a mobile app. The app automatically transmits information retrieved from the QR code as well as transaction location information to the blockchain platform. The blockchain network compares the information received from the app with the details of the manufacturer and distributor captured during the enrolment of the product. The result of the blockchain network is forwarded as a message to the end user as either genuine or fake.

The architecture of the blockchain-based anti-counterfeiting system is presented in Figure 1.

2.2 Functional Requirements of the System

The system should allow the following steps:

i. Manufacturers to register on the blockchain network through the process of request and approval after ascertaining their authenticity. Also, the system will enable approved distributors to record transactions (purchases of genuine products) on the blockchain network.
ii. The system will enable end users to register and access the blockchain network using the developed application, and the system will enable manufacturers, distributors, and the general public to validate the authenticity of a product using the blockchain network.
iii. The system will enable mutual agreement between manufacturers and distributors before ownership of a product (or products) can be transferred from the manufacturer to the distributor.
iv. The records of the state and status of every product (manufacturer to end user) are maintained by the system, and end users can scan the embedded QR code of products to ascertain their history and details as well as their location.
v. The system will alert administrators, manufacturers, distributors, end users, and officials if a counterfeit product is detected.

The Blockchain will be used to achieve the following functions:

i. The blockchain will allow distributors to register with a specific manufacturer on the network and will allow manufacturers to authenticate and approve distributors on the network.
ii. The blockchain will generate a unique QR code for every product that is registered on the network, and should issue unique IDs to manufacturers (and distributors alike), which will enable them to log onto the network. The blockchain will allow approved manufacturers to register their products on the network.

2.3 Use Case and Conceptual Class Diagrams of the Proposed System

The use case for registration of a manufacturer is shown in Table 1, and the use case and conceptual class diagrams of the system are shown in Figures 2 and 3, respectively.

Table 1: Use case for registering a manufacturer

Use Case Number: 1	**Use Case Name:** Register Manufacturer
Precondition: The Manufacturer approaches the system to be registered.	**Trigger:** The Manufacturer attempts to enrol in the BBACS network
Description: This use case details the steps involved in registering a manufacturer.	**Primary Actor:** Manufacturer/Admin.
Main Flow: The system prompts the manufacturer to input basic details The manufacturer inputs basic details System checks that basic details are correct If the details are not correct Stop the registration process. Notify the manufacturer. Go back to step A. Else if details are correct. Proceed to step D. System prompts Admin to authenticate the manufacturer The system checks if authentication is successful If authentication is not successful Stop the registration process Notify the manufacturer Go back to step A Else if authentication is successful Proceed to step F. The system completes the registration process by performing the following operations: System issues ID to the manufacturer The system stores details of the registration	
Special Requirements: Company registration certificate; confirmation of office address and location information (i.e., longitude and latitude)	
Post Condition: The Manufacturer has successfully registered, and details are stored in the blockchain.	

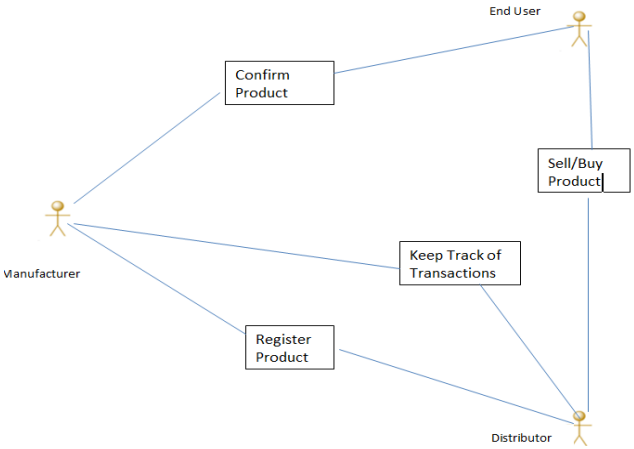

Figure 2: Use case diagram of the system

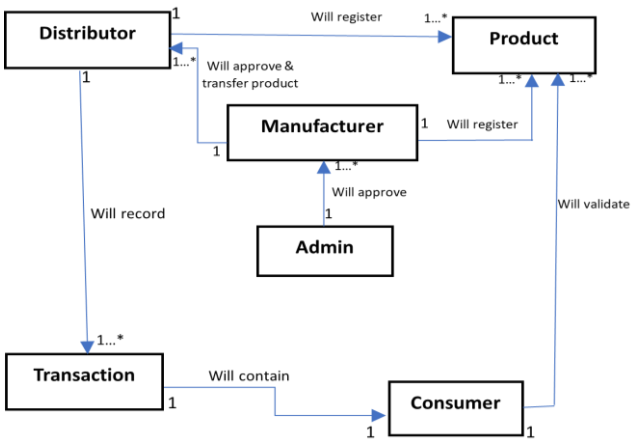

Figure 3: Conceptual class diagram of the system

2.4 Setting up a Private Ethereum Blockchain Network

In setting up a private Ethereum blockchain network for the product anti-counterfeiting system, the following steps are taken:

Step 1: Download and install Ganache. Ganache is the principal component of the Truffle Suite that will be utilised in setting up our personal Ethereum blockchain network.

Step 2: Download and install VS Code. VS Code is the main programming environment where the codes are written, edited, and debugged. VS Code was downloaded from http://code.visualstudio.com. Then, the installer was run.

Step 3: Develop the configuration file for the Ganache network. This is accomplished by writing a programme file named “anti-counterfeiting” in the VS Code environment.

Step 4: Run the Ganache. This was achieved by selecting Ethereum and clicking on New Workspace.

Step 5: Create a workbench on Ganache. We named the workspace "anti-counterfeiting" and added the configuration file that was created earlier.

Step 6: Install Truffle and Node.js in the configuration file.

Step 7: Download and install MetaMask. MetaMask is an extension for Chrome that enables interaction between the Ethereum dApp, blockchain network, and user interface.

2.5 Writing Smart Contracts

In this work, smart contracts are coded using the Solidity programming language. The algorithms and flowcharts of key system smart contracts are presented below.

Algorithm 1: Smart Contract for Registering Manufacturers

A. The system prompts the manufacturer to input basic details
B. The manufacturer inputs basic details
C. System checks that basic details are correct
  a. If details are not correct
    i. Stop the registration process.
    ii. Notify the manufacturer.
    iii. Go back to step A.
  b. Else if details are correct.
    i. Proceed to step D
D. System prompts Admin to authenticate the manufacturer
E. System checks if authentication is successful
  a. If authentication is not successful
    i. Stop the registration process
    ii. Notify the manufacturer
    iii. Go back to step A
  b. Else if authentication is successful
    i. Proceed to step F.
F. The system completes the registration process by performing the following operations:
  a. System issues ID to the manufacturer
  b. The system stores details of the registration

Algorithm 2: Smart Contract for Registering Product

A. The manufacturer is to input basic product details, as well as scan the product texture
B. The manufacturer inputs basic details
C. System checks that basic details are correct
  a. If details are not correct
    i. Stop the registration process.
    ii. Notify the manufacturer.
    iii. Go back to step A.
  b. Else if details are correct.
    i. Proceed to step D.
D. The system completes the registration process by performing these operations:
  a. System issues ID and generates QR code for the product
  b. The system stores details of the registration

Algorithm 3: Smart Contract for Product Confirmation

A. End User scans the QR code embedded on the product
B. The system automatically uploads product details generated from the scan to the network
C. System compares product details generated from the scan with those stored in the blockchain during product registration.
  a. If details are not the same
    i. Display “Product not found on BCACS network; product may likely be counterfeit”
    ii. Notify the enforcement officials.
    iii. Store details of the unsuccessful confirmation process
    iv. Go back to step A.
  b. Else if details are the same
    i. Display “Product is genuine”
    ii. The system stores details of the product confirmation

2.6 QR Code Generation

To ensure the development of an accurate model capable of generating QR codes for objects with varying textures and physical attributes, the following steps were undertaken:

2.6.1 Product features and image capture

The application prompts the user to supply product features. These inherent features include product name, size, weight, batch number, manufacturing date, expiry date, product texture, price, and any other relevant data. Once these features are submitted, the device's camera is activated to capture an image of the product.

2.6.2 Image processing

The captured image is processed to enhance the quality and extract relevant features using Convolutional Neural Networks (CNNs). This process involves image enhancement via CNNs and image feature extraction via CNNs:

(a) Image Enhancement via Convolutional Neural Networks (CNNs): to improve the quality of the captured images, this process incorporated denoising and super-resolution techniques using an Image Enhancement Network (IEN). The IEN, integrated within a Convolutional Neural Network (CNN) framework, is specifically designed to enhance image fidelity by learning and reconstructing high-quality representations from degraded inputs. This is shown in the code excerpt below;

```
import torch
import torch.nn as nn
import torchvision.models as models
# Image Enhancement Network using PyTorch Library
class EnhancementCNN(nn.Module):
    def __init__(self):
        super().__init__()
        self.enhance = nn.Sequential(
            nn.Conv2d(3, 64, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.Conv2d(64, 64, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.Conv2d(64, 3, kernel_size=3, padding=1)
        )
    def forward(self, x):
        return self.enhance(x)
```

(b) Image Feature Extraction via CNNs: the objective of this stage is to transform the enhanced image into a high-dimensional feature vector that encapsulates rich semantic information, including object identity, texture, and shape. To achieve this, the Feature Extraction Network (FEN) component of the Convolutional Neural Network (CNN) was employed. Specifically, a pre-trained ResNet-18 model was utilized, with its final classification layer removed to

obtain deep feature representations, as illustrated in the code excerpt below;

```
from torchvision.models import resnet18
class FeatureExtractor(nn.Module):
    def __init__(self):
        super().__init__()
        resnet = resnet18(pretrained=True)
        self.features = nn.Sequential(*list(resnet.children())[:-1])  # Remove
        FC layer

    def forward(self, x):
        x = self.features(x)
        return x.view(x.size(0), -1)  # Flatten (1, 512)
```

2.6.3 Resizing

The image is pre-processed to guarantee consistency in the later operations phase. To simplify the calculations in the next phase and maintain the vital visual features of the product, the image is resized to a standardized size, such as 24 X 24 pixels.

2.6.4 Convolutional filters

The specific feature of the product is extracted by applying 405 convolutional filters, popularly called Kernels. This technique traverses the image and executes mathematical activities on tiny patches of pixels with emphases on patterns, textures, edges, and other essential features of the product image.

2.6.5 Max pooling

Max pooling is a down-sampling method applied in CNNs to decrease the spatial dimensions of feature maps obtained from convolutional operations thereby reducing the computational complexity of the operations. Max pooling performs two primary functions in image processing:

(a) Translation Invariance: Max pooling aids in capturing the most fundamental features while being invariant to shifts in the input image. This helps the CNN to recognise features irrespective of their precise location in the image.

(b) Dimension Reduction: This dimension reduction helps in extracting more abstract and higher-level features from the image using the down-sampling effect of max pooling.

2.6.6 Matrix representation

The down-sampled feature map is translated into a matrix representation with decreased dimensions, similar to the matrix conversion. This Down-sampling lowers the computational complexity and enhance the efficiency of the resulting operations, finally helping to the generation of exact and detailed QR codes.

2.6.7 Matrix to Text Conversion

After the image processing phase, the image is converted to a matrix representation. The image dimension is also reduced to 24x24 pixels. Then the matrix representation is finally converted to plain text. This task was accomplished using Bootstrapped Language Image Pre-training (BLIP), a state-of-the-art vision-language model developed for unified image understanding and caption generation. BLIP leverages transformer-based architectures and contrastive pre-training to learn joint visual-linguistic representations from large-scale image-text datasets [24][25].

In the context of this work, BLIP was employed to automatically generate semantically rich textual descriptions of image textures. The model accepts preprocessed texture images as input and outputs human-readable captions that encapsulate the visual characteristics of the texture (e.g., pattern, structure, material type). This enables the transformation of low-level visual features into high-level semantic information, which can be used for downstream tasks such as indexing, classification, or metadata generation. An excerpt of the code implementation of this process is presented below as thus:

```
from transformers import BlipProcessor, BlipForConditionalGeneration
from PIL import Image
processor = BlipProcessor.from_pretrained("Jerryblockchain/blip-image-
captioning-base")
model=
BlipForConditionalGeneration.from_pretrained("Jerryblockchain/blip-
image-captioning-base")

def describe_image(image_path):
    raw_image = Image.open(image_path).convert('RGB')
    inputs = processor(raw_image, return_tensors="pt")
    out = model.generate(**inputs)
    return processor.decode(out[0], skip_special_tokens
```

2.6.8 Product information encryption

The extracted product features are converted into a JSON string. The JSON string combined with the image matrix as an input for an encryption algorithm. The encryption technique protects the data and ensures its confidentiality during transmission and storage. At this stage, the extracted feature vector was secured using the Advanced Encryption Standard (AES) algorithm with a 128-bit key (AES-128). The encryption was applied before storage or transmission to ensure data confidentiality and protect against unauthorized access or tampering. An excerpt is however represented below thus:

```
from Crypto.Cipher import AES
from Crypto.Random import get_random_bytes
import numpy as np

def encrypt_features(features, key):
    features_bytes = features.numpy().astype(np.float32).tobytes()
```

```
cipher = AES.new(key, AES.MODE_EAX)
ciphertext, tag = cipher.encrypt_and_digest(features_bytes)
return cipher.nonce, ciphertext, tag
```

2.6.9 QR code generation and uploading

The encrypted image from the encryption algorithm is an input to a QR code generator. This generates a QR code image that encodes the information used to unambiguously identify the product. The generated and encrypted data was uploaded to the IPFS via the Web3Storage. IPFS is a decentralized, peer-to-peer file storage and sharing protocol designed to make the web more distributed and resilient [26][27]. Instead of relying on a central server to store and retrieve files, IPFS uses a distributed network where files are identified by their content hash (i.e., a unique cryptographic fingerprint), not their location (like URLs). Therefore, after encrypting the data (an image feature vector), the encrypted file is uploaded to IPFS, making it accessible on a decentralized network.

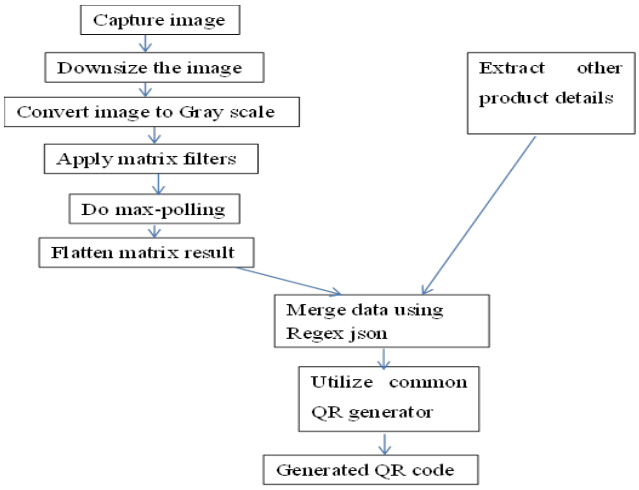

Figure 4: System flow diagram

2.6.10 Saving the QR code

The generated QR code is saved for future use or sharing. The saved QR code can be displayed on the screen, printed, or transmitted to be stored in a network. At this stage, a Solidity-based smart contract was designed and implemented to facilitate the decentralized storage and retrieval of image-associated metadata within a hash table structure. This contract enables efficient interaction with the blockchain by maintaining an immutable mapping between unique identifiers and their corresponding metadata records. The use of a hash table ensures constant-time complexity for storage and access operations, thereby optimizing performance. The implementation details are illustrated in the subsequent code excerpt.

```
{
  "image_id": "123abc",
  "feature_hash": "0xdeadbeef...",
  "ipfs_uri": "ipfs://QmXYZ...",
  "encryption_nonce": "base64...",
  "encryption_tag": "base64...",
  "uploader": "0xUserWallet..."
}
```

These steps are represented in the flow diagram shown in Figure 4.

3.0 Results and Discussion

3.1 Results

The results obtained from this work are shown and discussed in this section. Snapshots in Figures 5 and 6 confirm that the applications for generating and scanning QR codes are functioning properly. Figure 5 depicts the application window for a manufacturer to capture the product details, while Figure 6 is a QR code generated for a specific product. The manufacturer has five requirements to generate the QR code, and no two products share the same QR code.

Figure 5: Required product details for QR code generation

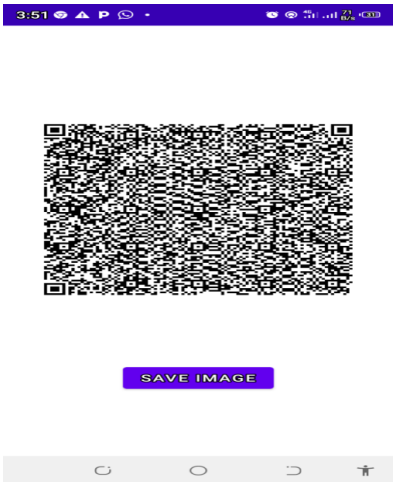

Figure 6: QR code generated by the application

Table 2 and Figure 7 show the results obtained by testing the developed system in terms of speed of execution, energy consumption, and ability to resist attacks.

Table 2: Speed of execution, energy consumption, and ability to resist attacks

S/N	Execution Time seconds)	Energy Consumption	Product Validation
1	39.0	Low	Genuine
2	41.0	Low	Genuine
3	36.0	Low	Fake
4	39.0	Low	Genuine
5	37.0	Low	Fake
Average	38.4		

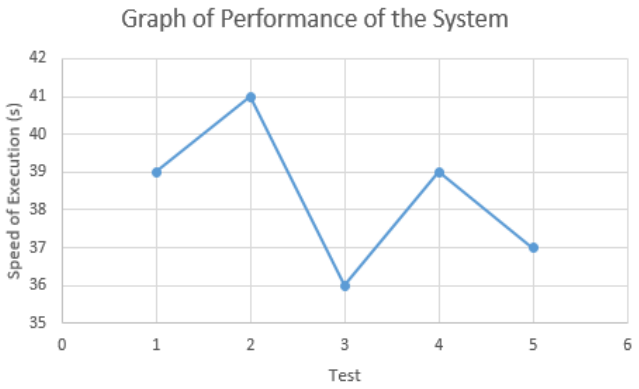

Figure 7: Graph of the performance of the system

From Table 2, the system has the following advantages: Energy is not wasted, as it is in proof-of-work and many other consensus mechanisms. Typically, the election of a new signer is made by the use of random criteria and weighted depending on the product contribution of a node. So, making more attempts per second does not increase the possibility of being selected as the new signer. Hence, a reduction in time and energy can be noticed. Without the need to compute a lot of hashes to find the correct one, it is possible to create a chain with a higher rate of block generation per 38 seconds.

The result in Table 3 is the comparative analysis of the developed QR code system with existing works of literature to prove its validity. Two Key Performance Indicators (KPIs) used in this test include the scanning speed and the accuracy of the scanned result. Fifty (50) products' QR codes were scanned at a light intensity of 1000 lux to determine their originality.

Table 3: Comparative analysis

	Detection Accuracy (%)	QR Code Scanning Time
[28]	92 and 98	35ms
[29]	78.63	23ms
[30]	93.50	22ms
[31]	99.13	12ms
[32]	98.90	10.3ms
This work	100	9.5ms

3.2 Discussion

The immutability of stored data, security strength, speed of execution, product validation accuracy, as well as the uniqueness of QR codes embedded on products, are considered the key parameter indicators (KPIs) for testing this system.

Energy is not wasted, as it is in proof-of-work and many other consensus mechanisms. Typically, the election of a new signer is made by the use of random criteria and weighted depending on the product contribution of a node. So, making more attempts per second does not increase the possibility of being selected as the new signer. Hence, a reduction in time and energy can be noticed in Table 2. Without the need to compute a lot of hashes to find the correct one, it is possible to create a chain with a higher rate of block generation per 38 seconds as presented in Table 3.

The test results show that in a well-regulated environment where the luminous intensity is kept constant at a good level (1000 lux), the accuracy of the system is 100% from the thirty (30) genuine and twenty (20) counterfeit products scanned as shown in Table 2. The average scanning time is 9.5ms using the same computer system, an Intel Core i5 laptop running on a 1TB HDD and 8GB RAM at a processor speed of 3.5GHz. Therefore, the developed system accurately detects counterfeit products at a very high speed, which is quite different from already existing systems.

a. Limitations

Smart contracts lack the computational power and memory to run deep learning models; hence CNN computations cannot directly run on Blockchain. Thus, CNN inference must happen off-chain (e.g., cloud, edge devices), and only metadata or hashes are stored on-chain. In addition, AES encryption of large feature vectors can be computationally intensive for low-power devices, which may increase latency.

b. Scalability

Blockchain scalability solutions aimed to increase the transaction processing capacity of the blockchain networks were employed. This is crucial for wider adoption as blockchains face limitations in handling high transaction volumes. Sharding, Consensus Mechanism Improvements and Sidechains solutions were applied to improve the scalability of the developed system.

c. Cost implications

It is important to note that although blockchain integration requires the deployment of smart contracts on public or private chains, public chains incur gas fees whereas private/consortium chains such as the one developed in this work reduce costs but have limited decentralization [33].

d. Deployment and Device Requirements

For the successful deployment and full operationalization of the blockchain-based anti-counterfeiting system, it is imperative to utilize a computational device equipped with GPU acceleration capabilities, specifically those compatible with NVIDIA CUDA architectures, to facilitate expedited inference processing. The computational environment must support Python runtime with comprehensive cryptographic libraries (e.g., PyCryptodome, cryptography). Additionally, end-users are required to possess Ethereum-compatible digital wallets—such as MetaMask or Ledger hardware wallets—to interact with the blockchain layer securely.

4.0 CONCLUSIONS

This work has been able to achieve its primary objective of developing a blockchain-based anti-counterfeiting system leveraging product inherent features and location information. First, it demonstrated that blockchain technology has the potential to redefine the battle against counterfeiting as it is chronologically updated and cryptographically sealed, thereby guaranteeing immutable transactions that are publicly available and distributed globally.

Thus, Ganache, a private Ethereum blockchain network, was set up to serve as the backend platform due to the fact that our system requires multi-state and flexibility in block storage to record data, as well as a short block time. The system prototype is a distributed application (dApp) with a supporting blockchain network. Furthermore, smart contracts were written in the Solidity programming language, compiled, and deployed to the developed anti-counterfeiting blockchain network via Truffle.

The developed distributed application allows manufacturers and distributors to enrol in the private Ganache blockchain-based anti-counterfeiting network. Registered manufacturers are also granted the privilege to register their products using a QR code generator. Finally, end-users scan QR codes embedded on products and verify from the network whether the product is genuine or counterfeit.

REFERENCES

[1] Yiu, N. C. K. "Toward Blockchain-Enabled Supply Chain Anti-Counterfeiting and Traceability", Future Internet, 13(4), 2021; 86. https://doi.org/10.3390/fi13040086

[2] OECD/EUIPO (2020), *Trade in Counterfeit Pharmaceutical Products*, Illicit Trade, OECD Publishing, Paris, https://doi.org/10.1787/a7c7e054-en.

[3] Keith, W., and Jith, L. Z. "The Evolution of Counterfeit Luxury Consumption", *Research Handbook on Luxury Branding*, pp. 265 – 281, 2020 ISBN: 978 1 78643 634 4. https://url-shortener.me/1SCB

[3] Tavares, I. D. "Counterfeiting Of Fake Drugs in Africa: Current Situation, Causes and Countermeasures", *Mondaq*, 2020. https://www.mondaq.com/nigeria/trademark/988968/counterfeiting-of-fake-drugs-in-africa-current-situation-causes-and-countermeasures.

[4] Y. Dabbagh, R. Khoja, L. AlZahrani, G. AlShowaier and N. Nasser, "A Blockchain-Based Fake Product Identification System," 2022 5th Conference on Cloud and Internet of Things (CIoT), Marrakech, Morocco, 2022, pp. 48-52, https://doi.org/10.1109/CIoT53061.2022.9766493.

[5] Templars ThoughtLab, "Securing Legitimacy: Anti-counterfeiting Strategies for Brand Owners in Nigeria", 2024, accessed from www.templars-law.com on 27th May 2025.

[6] EIC and SMEs Executive Agency, "Challenges of counterfeiting in Nigeria", 2024, accessed from https://intellectual-property-helpdesk.ec.europa.eu on 27th May 2025

[7] David N., and Ohanado U. "Leveraging Emerging Technologies for Operational Optimization and Business Model Innovation in African Enterprises", *Nigerian Journal of Technology*, vol. 43, no. 4, pp. 829 – 838, Jan. 2025, doi: 10.4314/njt.v43i4.23.

[8] Okide C. P., Ahaneku M. A., Nwawelu U. N., Chijindu V. C., Ezeja, O. M., and Ekengwu B. O. "Deployment of Smart Contracts on Blockchain Technology in Early Mitigation of Distributed Denial of Service in Software Defined Networks", *Nigerian Journal of Technology*, vol. 44, no. 1, pp. 105–113, Apr. 2025, doi: 10.4314/njt.v44i1.12.

[9] Ahmed, S. "Enhancing Data Security and Transparency: The Role of Blockchain in Decentralized System", *International Journal of Advanced Engineering, Management and Science*, 11(1), 593258, 2025. https://doi.org/10.22161/ijaems.111.12

[10] Wood, G. "Ethereum: A Secure Decentralised Generalised Transaction Ledger (EIP-150)", 2018. https://www.yellowpaper.io.

[11] Khan, U., An, Z. Y., and Imran, A. "A Blockchain Ethereum Technology-Enabled Digital Content: Development of Trading and Sharing Economy Data", *IEEE Access*, 8, 217045-217056; 2020. https://doi.org/10.1109/ACCESS.2020.3041317.

[12] Santiago, C., Ren, S., Lee, C., and Ryu, M. "Concordia: A Streamlined Consensus Protocol for Blockchain Networks", *IEEE Access*, 9, 13173-13185; 2021. https://doi.org/10.1109/ACCESS.2021.3051796.

[13] Zhu, P., Hu, J., Zhang, Y., and Li, X. "A Blockchain Based Solution for Medication Anti-Counterfeiting and Traceability", *IEEE Access*, 8, 184256-184272; 2020 vol. 8, pp. 184256-184272, 2020, https://doi.org/10.1109/ACCESS.2020.3029196

[14] Du, M., Chen, Q., and Ma, X. "MBFT: A New Consensus Algorithm for Consortium Blockchain", *IEEE Access*, 8, 87665-87675; 2020. https://doi.org/10.1109/ACCESS.2020.2993759.

[15] Dwivedi, V., Norta, A., Wulf, A., Leiding, B., Saxena S., and Udokwu, C. "A Formal Specification Smart-Contract Language for Legally Binding Decentralized Autonomous Organizations", *IEEE Access*, vol. 9, pp. 76069-76082; 2021. https://doi.org/10.1109/ACCESS.2021.3081926.

[16] Kannengiesser, N., Lins, S., Sander, C., Winter, K., Frey, H., and Sunyaev, A. "Challenges and Common Solutions in Smart Contract Development", *IEEE Transactions on Software Engineering*, 2021 vol. 48, no. 11, pp. 4291-4318. https://doi.org/10.1109/TSE.2021.3116808.

[17] Zichichi, M., Ferretti, S., and D'Angelo, G. "MOVO: a dApp for DLT-based Smart Mobility", *International Conference on Computer Communications and Networks (ICCCN)*, 1-6; 2021. https://doi.org/10.1109/ICCCN52240.2021.9522257.

[18] Gao, J. "Guided, Automated Testing of Blockchain-Based Decentralized Applications" , *IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)*, 138-140; 2019. https://doi.org/10.1109/ICSE-Companion.2019.00059

[19] Wei, W., Lin, J., Lin, Y., and Liao, H. M. "What Makes You Look Like You: Learning an Inherent Feature Representation for Person Re-Identification", *16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, Pg 1-6; 2019 https://doi.org/10.1109/AVSS.2019.8909892.

[20] Wang, S., Li, D., Zhang, Y., and Chen, J. "Smart Contract-Based Product Traceability System in the Supply Chain Scenario", *IEEE Access*, 7, 115122-115133; 2019. https://doi.org/10.1109/ACCESS.2019.2935873.

[21] Ma, J., Lin, S., Chen, X., Sun, H., Chen, Y., and Wang, H. "A Blockchain-Based Application System for Product Anti-Counterfeiting", *IEEE Access*, 8, 77642-77652; 2020. https://doi.org/10.1109/ACCESS.2020.2972026.

[22] Srivatsa, D., Aakash, N., and Sahisnu, S. "A Product Authentication Scheme for Supply Chain system via Smart Contracts using Blockchain Technology and Facial Recognition", *Journal of Physics: Conference Series*, 1767. 012057. 2020. https://doi.org/10.1088/1742-6596/1767/1/012057

[23] Li, J., Li, D., Xiong, C., and Hoi, S. C. H. "BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation", *Proceedings of the 39th International Conference on Machine Learning*, 162, 12888–12900; 2021 https://doi.org/10.48550/arXiv.2201.12086

[24] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., and Sutskever, I. "Learning transferable visual models from natural language supervision", *Proceedings of the 38th International Conference on Machine Learning*, 139, 8748–8763; 2022. https://doi.org/10.48550/arXiv.2103.00020.

[25] Benet, J. "IPFS - Content addressed, versioned, P2P file system", *arXiv* preprint, arXiv:1407.3561. 2014. https://arxiv.org/abs/ 1407.3561

[26] Singh, R., and Sharma, A. "InterPlanetary File System (IPFS): A decentralized storage network", *Materials Today: Proceedings*, 47, 2267–2271; 2021. https://doi.org/10.1016/j.matpr.2021.04.853

[27] Prabhu Shankar, B., Jayavadivel, R., and Viswanath Kani, T. "A Novel Approach For Detect Counterfeit Product Using Color QR Code", *International Journal of Scientific and Technology Research*, Vol. 9, No. 01, 2020, http://www.ijstr.org/final-print/jan2020/A-Novel-Approach-For-Detect-Counterfeit-Product-Using-Color-Qr-Code.pdf

[28] Wahsheh, H. "BarSec Droid", 2018. [Online]. Available: https://m.apkpure.com/barsec-droid/barcode_security.heider.bsr

[29] Rafsanjani, A. S., Kamaruddin, N. B., Rusli, H. M., and Dabbagh, M. "QsecR: Secure QR Code

Scanner According to a Novel Malicious URL Detection Framework", *in IEEE Access*, vol. 11, pp. 92523-92539, 2023, https://ieeexplore.ieee.org/document/10172018

[30] P. Kamnounsing, K. Sumongkayothin, P. Siritanawan and K. Kotani, "Adversarial Halftone QR Code," in IEEE Access, vol. 12, pp. 126729-126737, 2024, https://doi.org/10.1109/ACCESS.2024.3405408

[31] Chindaudom, A., Siritanawan, P., Sumongkayothin, K., and Kotani, K. "Adversarial QR: An adversarial patch in QR code format", *2020 Joint 9th International Conference on Informatics, Electronics and Vision (ICIEV) and 2020 4th International Conference on Imaging, Vision and Pattern Recognition (icIVPR)*, Kitakyushu, Japan, 2020, pp. 1-6, doi: 10.1109/ICIEVicIVPR48672.2020.9306675.

[32] Zheng, Z., Xie, S., Dai, H., Chen, X., and Wang, H. "An overview of blockchain technology: Architecture, consensus, and future trends", *Proceedings of the 2017 IEEE International Congress on Big Data (BigData Congress)*, 557–564; 2017. https://doi.org/10.1109/BigDataCongress.2017.85