RV COLLEGE OF ENGINEERING

BENGALURU- 560059

(Autonomous Institution affiliated to VTU, Belagavi)

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

"Fake Product Identification Using Blockchain"

BLOCKCHAIN TECHNOLOGY AND USE CASE (18IS7F2)

Experiential Learning VII Semester

Academic year 2023-2024

Submitted by

Shashi Ranjan (1RV20IS050)

Under the guidance of

Prof. Sharadadevi K

Assistant Professor, Dept of ISE

RV College of Engineering

RV COLLEGE OF ENGINEERING

BENGALURU- 560059

(Autonomous Institution affiliated to VTU, Belagavi)

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the work titled **"Fake Product Identification Using Blockchain"** has been carried out by **Shashi Ranjan** (1RV20IS050), bona fide student of RV College of Engineering, Bengaluru, who has submitted in partial fulfillment for the Assessment of Course: **Blockchain Technology and Use Case (18IS7F2) – Experiential Learning** during the year 2023-2024. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report.

Faculty in-charge
Prof Sharadadevi K

Head of Department
Dr Sagar B M

ABSTRACT

There are many fake products in the existing supply chain. It is necessary to have a system for end user to check all details about product that they are buying so that the customer can check if the product is genuine or not. In recent years, Counterfeit products play an important role in product manufacturing industries. This affects the company name, sales, and profit of the companies. Block technology is used to identification of real products and detects fake products. Blockchain technology is the distributed, decentralized and digital ledger that stores transactional information in the form of blocks in many database/node-computers which is connected with the chains. Blockchain technology is secure as the data stored once in the chain is immutable therefore any block cannot be changed or hacked. By using Blockchain technology, customers or users do not need to rely on third-party users for confirmation of product authenticity and safety.

This System provides the emerging technology of web use cases, Quick Response (QR) codes provide a robust technique to fight the practice of counterfeiting the products. Counterfeited products can be detected using a QR code scanner, where a QR code of the product is linked to Blockchain. So, this system may be used to store product details and generated unique code of that product as blocks in database. It collects the unique code from the user and compares the code against entries in the Blockchain database. If the code matches, it will give all the information of the product otherwise no information will be outputted to the customer which shows that the product is fake or counterfeited.

TABLE OF CONTENTS

Chapter 1
INTRODUCTION

In the current advancing world of technology, the global development of a product or technology always comes with risk factor such as counterfeiting and duplication, which can affect the company name, company revenue, and customer health. The basic idea of the project is to verify that the product purchased by the customer is fake or real. In comparison with blockchain we have traditional supply chain. Traditional supply chain provides centralized network where the data is in the hand of the company which provides the service or the products in the market, and they own the data so they can manipulate as per their wish so they are not secure. Counterfeiting of the product are produced to take advantage of the superior value of the imitated products.

As mentioned, traditional supply chain provides centralized network whereas Blockchain provides decentralized data base, every transaction involving the data value for the product. This is done by creating a record whose authenticity can be verified by the entire community since blockchain runs by peer-to-peer network. In such a way manufacturer can use this system to provide genuine products to the customer. This will help to maintain the customer trust and to increase the brand value of the product in the market. In blockchain every block consists of data, hash and previous block hash. Data contains the relevant information and hash consists of the unique code. It is impossible to change data of any block since person changing the data requires to own the majority of the network. If we try to change the data of any block the hash will get changed. So, this becomes the major advantage over the traditional centralized architecture where the data in blockchain is immutable so that the customer who buys the product gets the genuine information of the product.

1.1 Topic relevance

- **Immutable Product Records, Resistance to Tampering:** Blockchain ensures the immutability of product-related data, making it resistant to tampering or fraudulent activities. Each stage in the product lifecycle, from manufacturing to distribution, can be securely recorded on the blockchain, providing an unalterable history of the product's authenticity.
- **Enhanced Data Ownership and Privacy for Consumers**: Implementing blockchain in product identification allows consumers to have ownership and control over the data related to the products they purchase. Decentralized storage and encryption mechanisms ensure that consumers can protect their privacy, preventing unauthorized access to personal information associated with their product choices.
- **Transparent Rewards for Anti-Counterfeiting Efforts**: Introducing blockchain enables the creation of a transparent and trustless incentive system. This system can directly reward individuals or entities contributing to anti-counterfeiting efforts, such as reporting counterfeit products or verifying the authenticity of items. This fosters a community-driven approach to combating fake products and establishes a self-sustaining ecosystem for identifying and eliminating counterfeits.

1.2 Objectives

- To Design an Anti-Counterfeit System using Blockchain.
- To secure product details using a QR code.
- To stop the making of fake products by conducting transparency about the products to the notice of the consumers.
- To become aware of fake products from the primary product available in the marketplace and to enhance this performance

Chapter 2
LITERATURE SURVEY

[1] Barcodes were broadly utilized for computerized item distinguishing proof purposes for a wide scope of ongoing applications. Holograms are utilized to battle fakes and lessen phony. Joining the possibility of standardized tags into Holograms offers two Protections against counterfeiting. In a paper, the creator presented recognizable proof by utilizing strategic standardized tags that increment the wellbeing and dependability of the item. Client characterized item recognizable proof code that is remarkable and conveyed to each is changed over to Product Quick Response (QR) code. QR-produced code is then changed over into a multidimensional image to give higher security to the item. The creator has centered on holographic scanner tags utilizing PC-produced Holograms are carried out utilizing Matlab. The Barcode interpreting gives the reproduced from the visualization the item ID number relegated to the item. At last, it provides an idea about will assist with expanding security and credibility in item recognizable proof and to forestall duplicating of items. Notwithstanding, the creator has thought about just the PC-created standardized tag utilizing MATLAB and has not thought of imitation of QR code.

[2]In the have analyzed with respect to counterfeit things are filling significantly in the online market. The square market is the biggest test in-store organization. The government has introduced a couple of guidelines and rules against fake things regardless of the way that the government can't deal with counterfeit things. Consequently, a system needs to be designed for recognizing counterfeit things and giving security techniques to alert both maker and purchaser underway organization. Creators may use the square chain the leader's system to store critical thing bargains information inside the square chain, which is accessible to all. The total number of arrangements the dealer can sell and the rest left behind by the seller are clear. The client can perform a vendor-side check using an encryption computation. In this paper, we proposed blockchain the chiefs system institutes the client and undertaking merchant to follow and perceive the real thing using a PDA. It will recognize counterfeit things as well as the authenticity of the producer for both end client and adventure vendor.

[3]In this paper the author proposes a system that provides a solution to the originality and authenticity of published and posted online digital content like music, books, etc. The system utilizes emerging technologies that primarily include blockchain and (interplanetary file system) IPFS. The solution is focused on the authenticity of online books, but the solution in terms of architecture, design, logic, smart contract code is generic enough to be easily extended and is used to provide the originality, authenticity, and integrity to all the other forms of digital assets. The author considered two scenarios based on the approval results provided by the author for every publication requesting an attestation or validation before uploading the content.

[4]This system proposes a solution that relies on machine learning-based technology which enables end-consumers to identify and verify products without any special equipment. By using image and text recognition. For identification, the end-consumers take photos of an item packaging, which contains product text information, logos, and perhaps accreditation marks/logos. These photos will be sent in a solicitation to the worker for processing and confirmation. Afterward, the detection result will be returned to the end-consumer to make a further decision. In the case of fake product detection, the end-consumer has the ability to report this counterfeit product to the government system, such as the Safety Gate - EU's Rapid Alert System

[5]Holograms have been utilized to fight against fakes and to diminish forgery. Incorporating the possibility of standardized identifications into holograms give two level protection from forging. This framework acquaints a technique with increased security and genuineness in item distinguishing proof by utilizing holographic standardized identifications. The unique and client characterized item distinguishing proof code accommodated every item is changed over into Quick Response (QR) Code. The QR Code created is then changed over into a 3D image, hence giving a prominent security to the item. In this framework, Holographic standardized tags utilizing Computer created visualizations are actualized utilizing MATLAB. The decoding of the barcode recreated from the holograms gives the corresponding product recognizable proof number assigned for the product.

Chapter 3

SYSTEM ARCHITECTURE AND TECHNOLOGY

3.1 Architecture Diagram

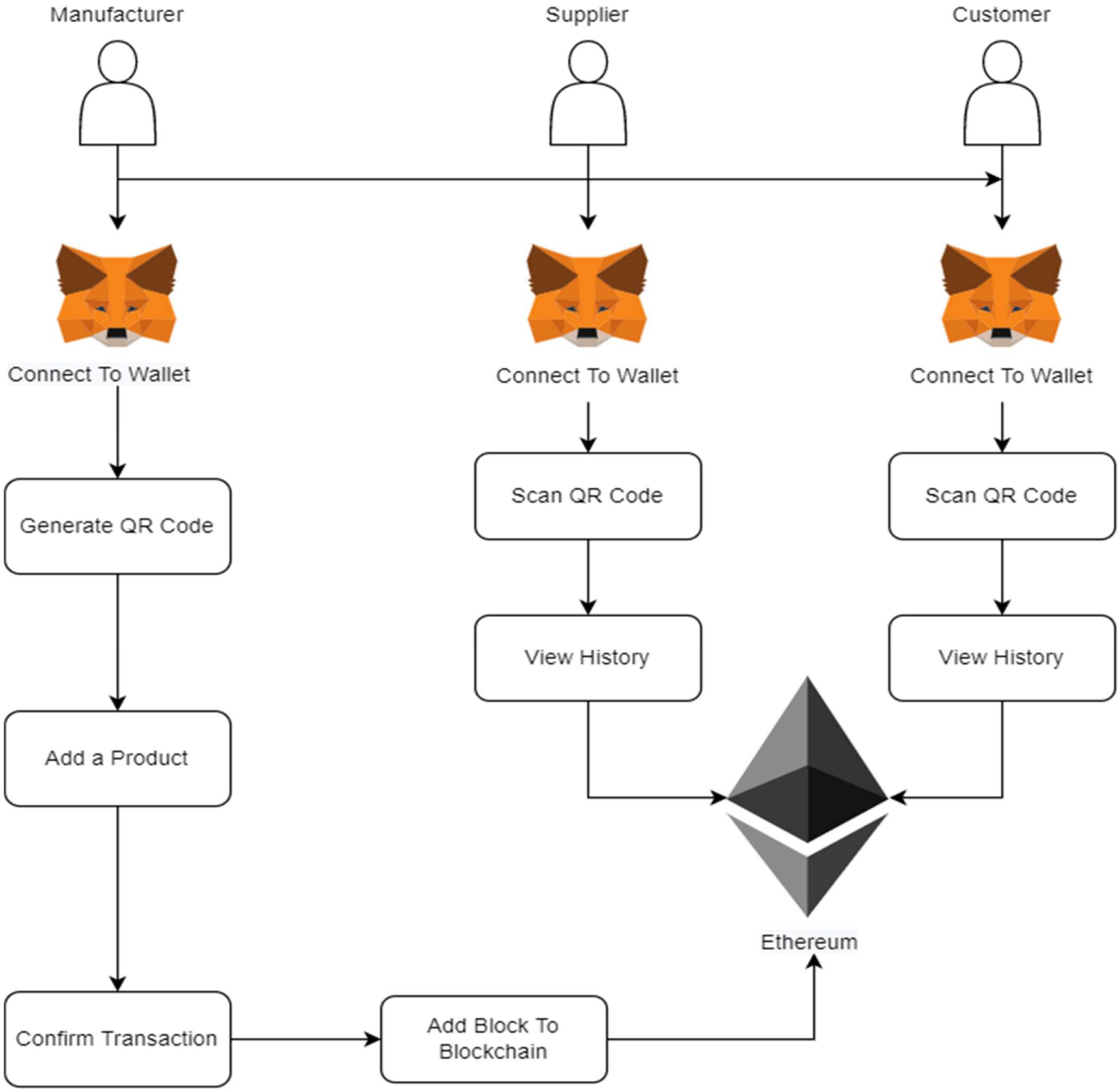

Figure 3.1. Architecture Diagram

3.2 Technology/Tool Used

1. **HTML**:-
    - Structuring the layout of the web pages for the product identification system.
    - Defining the necessary forms and input fields for user interaction.
    - Displaying essential information related to product details and identification.
2. **CSS:-**
    - Styling and formatting HTML elements to create a visually appealing and consistent user interface
    - Ensuring a responsive design for seamless user experience across various devices.
3. **JavaScript :-**
    - Adding dynamic functionality to enhance user interactions and responsiveness.
    - Implementing real-time updates for product information and transaction history.
    - Managing asynchronous operations for a smoother user experience
4. **Solidity:** Smart Contract Development
    - Writing smart contracts to define rules and logic for product identification on the blockchain.
    - Ensuring the security and integrity of transactions through blockchain-based verification.
    - Creating a tamper-resistant environment by leveraging blockchain technology.
5. **MetaMask** :-Cryptocurrency wallet and browser extension that allows users to interact with decentralized applications (DApps) on the Ethereum blockchain.
6. **Ganache :-**local blockchain emulator that facilitates Ethereum development by providing a simulated blockchain environment for testing and debugging decentralized applications .

Chapter 4

IMPLEMENTATION

4.1 Product.sol

```
pragma solidity ^0.8.12;

contract product {
    uint256 sellerCount;
    uint256 productCount;

    struct seller{
        uint256 sellerId;
        bytes32 sellerName;
        bytes32 sellerBrand;
        bytes32 sellerCode;
        uint256 sellerNum;
        bytes32 sellerManager;
        bytes32 sellerAddress;
    }
    mapping(uint=>seller) public sellers;

    struct productItem{
        uint256 productId;
        bytes32 productSN;
        bytes32 productName;
        bytes32 productBrand;
        uint256 productPrice;
        bytes32 productStatus;
    }

    mapping(uint256=>productItem) public productItems;
    mapping(bytes32=>uint256) public productMap;
    mapping(bytes32=>bytes32) public productsManufactured;
    mapping(bytes32=>bytes32) public productsForSale;
    mapping(bytes32=>bytes32) public productsSold;
    mapping(bytes32=>bytes32[]) public productsWithSeller;
    mapping(bytes32=>bytes32[]) public productsWithConsumer;
    mapping(bytes32=>bytes32[]) public sellersWithManufacturer;


    //SELLER SECTION
```

```
function addSeller(bytes32 _manufacturerId, bytes32 _sellerName, bytes32 _sellerBrand, bytes32
    _sellerCode,
uint256 _sellerNum, bytes32 _sellerManager, bytes32 _sellerAddress) public{
   sellers[sellerCount] = seller(sellerCount, _sellerName, _sellerBrand, _sellerCode,
   _sellerNum, _sellerManager, _sellerAddress);
   sellerCount++;

   sellersWithManufacturer[_manufacturerId].push(_sellerCode);
}

function viewSellers () public view returns(uint256[] memory, bytes32[] memory, bytes32[] memory,
    bytes32[] memory, uint256[] memory, bytes32[] memory, bytes32[] memory) {
   uint256[] memory ids = new uint256[](sellerCount);
   bytes32[] memory snames = new bytes32[](sellerCount);
   bytes32[] memory sbrands = new bytes32[](sellerCount);
   bytes32[] memory scodes = new bytes32[](sellerCount);
   uint256[] memory snums = new uint256[](sellerCount);
   bytes32[] memory smanagers = new bytes32[](sellerCount);
   bytes32[] memory saddress = new bytes32[](sellerCount);

   for(uint i=0; i<sellerCount; i++){
      ids[i] = sellers[i].sellerId;
      snames[i] = sellers[i].sellerName;
      sbrands[i] = sellers[i].sellerBrand;
      scodes[i] = sellers[i].sellerCode;
      snums[i] = sellers[i].sellerNum;
      smanagers[i] = sellers[i].sellerManager;
      saddress[i] = sellers[i].sellerAddress;
   }
   return(ids, snames, sbrands, scodes, snums, smanagers, saddress);
}

//PRODUCT SECTION

function addProduct(bytes32 _manufactuerID, bytes32 _productName, bytes32 _productSN, bytes32
    _productBrand,
uint256 _productPrice) public{
   productItems[productCount] = productItem(productCount, _productSN, _productName,
    _productBrand,
   _productPrice, "Available");
   productMap[_productSN] = productCount;
   productCount++;
   productsManufactured[_productSN] = _manufactuerID;
}
```

```
function viewProductItems () public view returns(uint256[] memory, bytes32[] memory, bytes32[]
    memory, bytes32[] memory, uint256[] memory, bytes32[] memory) {
   uint256[] memory pids = new uint256[](productCount);
   bytes32[] memory pSNs = new bytes32[](productCount);
   bytes32[] memory pnames = new bytes32[](productCount);
   bytes32[] memory pbrands = new bytes32[](productCount);
   uint256[] memory pprices = new uint256[](productCount);
   bytes32[] memory pstatus = new bytes32[](productCount);

   for(uint i=0; i<productCount; i++){
      pids[i] = productItems[i].productId;
      pSNs[i] = productItems[i].productSN;
      pnames[i] = productItems[i].productName;
      pbrands[i] = productItems[i].productBrand;
      pprices[i] = productItems[i].productPrice;
      pstatus[i] = productItems[i].productStatus;
   }
   return(pids, pSNs, pnames, pbrands, pprices, pstatus);
}

//SELL Product

function manufacturerSellProduct(bytes32 _productSN, bytes32 _sellerCode) public{
   productsWithSeller[_sellerCode].push(_productSN);
   productsForSale[_productSN] = _sellerCode;

}

function sellerSellProduct(bytes32 _productSN, bytes32 _consumerCode) public{
   bytes32 pStatus;
   uint256 i;
   uint256 j=0;

   if(productCount>0) {
      for(i=0;i<productCount;i++) {
         if(productItems[i].productSN == _productSN) {
            j=i;
         }
      }
   }

   pStatus = productItems[j].productStatus;
   if(pStatus == "Available") {
      productItems[j].productStatus = "NA";
      productsWithConsumer[_consumerCode].push(_productSN);
      productsSold[_productSN] = _consumerCode;
```

```
    }


}


function queryProductsList(bytes32 _sellerCode) public view returns(uint256[] memory, bytes32[]
    memory, bytes32[] memory, bytes32[] memory, uint256[] memory, bytes32[] memory){
   bytes32[] memory productSNs = productsWithSeller[_sellerCode];
   uint256 k=0;

   uint256[] memory pids = new uint256[](productCount);
   bytes32[] memory pSNs = new bytes32[](productCount);
   bytes32[] memory pnames = new bytes32[](productCount);
   bytes32[] memory pbrands = new bytes32[](productCount);
   uint256[] memory pprices = new uint256[](productCount);
   bytes32[] memory pstatus = new bytes32[](productCount);


   for(uint i=0; i<productCount; i++){
      for(uint j=0; j<productSNs.length; j++){
         if(productItems[i].productSN==productSNs[j]){
            pids[k] = productItems[i].productId;
            pSNs[k] = productItems[i].productSN;
            pnames[k] = productItems[i].productName;
            pbrands[k] = productItems[i].productBrand;
            pprices[k] = productItems[i].productPrice;
            pstatus[k] = productItems[i].productStatus;
            k++;
         }
      }
   }
   return(pids, pSNs, pnames, pbrands, pprices, pstatus);
}

function querySellersList (bytes32 _manufacturerCode) public view returns(uint256[] memory,
    bytes32[] memory, bytes32[] memory, bytes32[] memory, uint256[] memory, bytes32[] memory,
    bytes32[] memory) {
   bytes32[] memory sellerCodes = sellersWithManufacturer[_manufacturerCode];
   uint256 k=0;
   uint256[] memory ids = new uint256[](sellerCount);
   bytes32[] memory snames = new bytes32[](sellerCount);
   bytes32[] memory sbrands = new bytes32[](sellerCount);
   bytes32[] memory scodes = new bytes32[](sellerCount);
   uint256[] memory snums = new uint256[](sellerCount);
   bytes32[] memory smanagers = new bytes32[](sellerCount);
```

```
        bytes32[] memory saddress = new bytes32[](sellerCount);

        for(uint i=0; i<sellerCount; i++){
            for(uint j=0; j<sellerCodes.length; j++){
                if(sellers[i].sellerCode==sellerCodes[j]){
                    ids[k] = sellers[i].sellerId;
                    snames[k] = sellers[i].sellerName;
                    sbrands[k] = sellers[i].sellerBrand;
                    scodes[k] = sellers[i].sellerCode;
                    snums[k] = sellers[i].sellerNum;
                    smanagers[k] = sellers[i].sellerManager;
                    saddress[k] = sellers[i].sellerAddress;
                    k++;
                    break;
                }
            }
        }

        return(ids, snames, sbrands, scodes, snums, smanagers, saddress);
    }

    function getPurchaseHistory(bytes32 _consumerCode) public view returns (bytes32[] memory,
        bytes32[] memory, bytes32[] memory){
        bytes32[] memory productSNs = productsWithConsumer[_consumerCode];
        bytes32[] memory sellerCodes = new bytes32[](productSNs.length);
        bytes32[] memory manufacturerCodes = new bytes32[](productSNs.length);
        for(uint i=0; i<productSNs.length; i++){
            sellerCodes[i] = productsForSale[productSNs[i]];
            manufacturerCodes[i] = productsManufactured[productSNs[i]];
        }
        return (productSNs, sellerCodes, manufacturerCodes);
    }

    //Verify

    function verifyProduct(bytes32 _productSN, bytes32 _consumerCode) public view returns(bool){
        if(productsSold[_productSN] == _consumerCode){
            return true;
        }
        else{
            return false;
        }
    }
```

Chapter 5:- RESULTS AND OUTPUTS

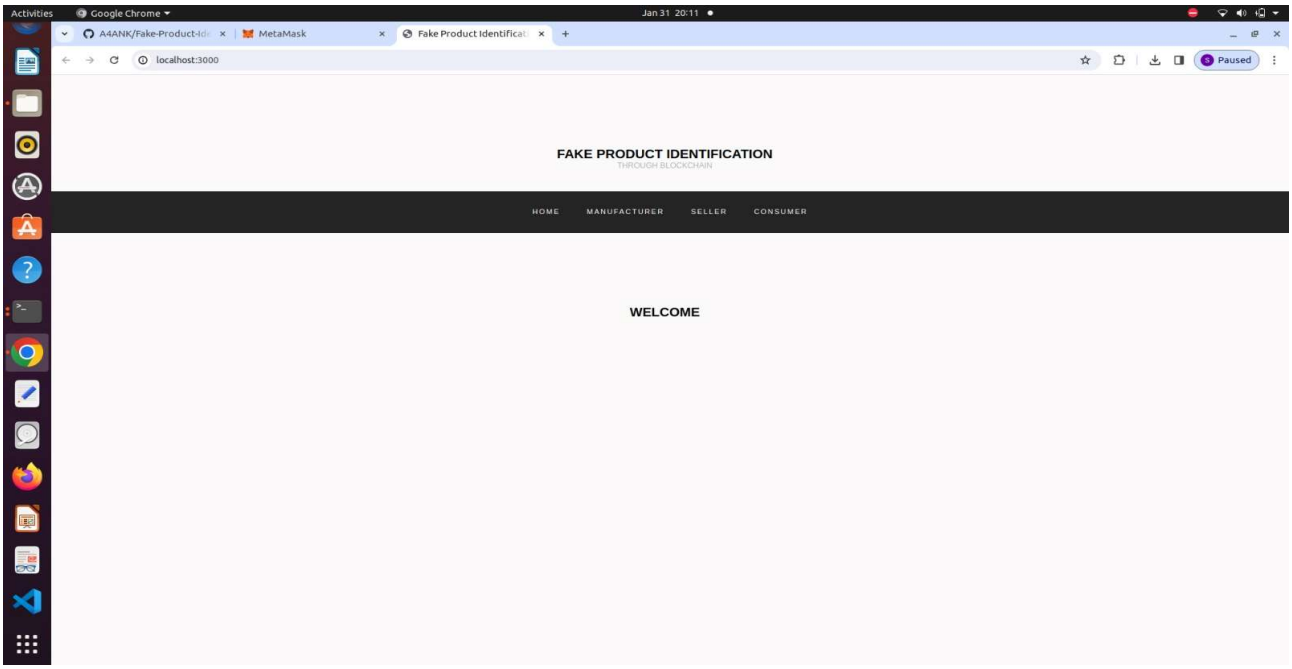

Fig 5.1 :-Home Page

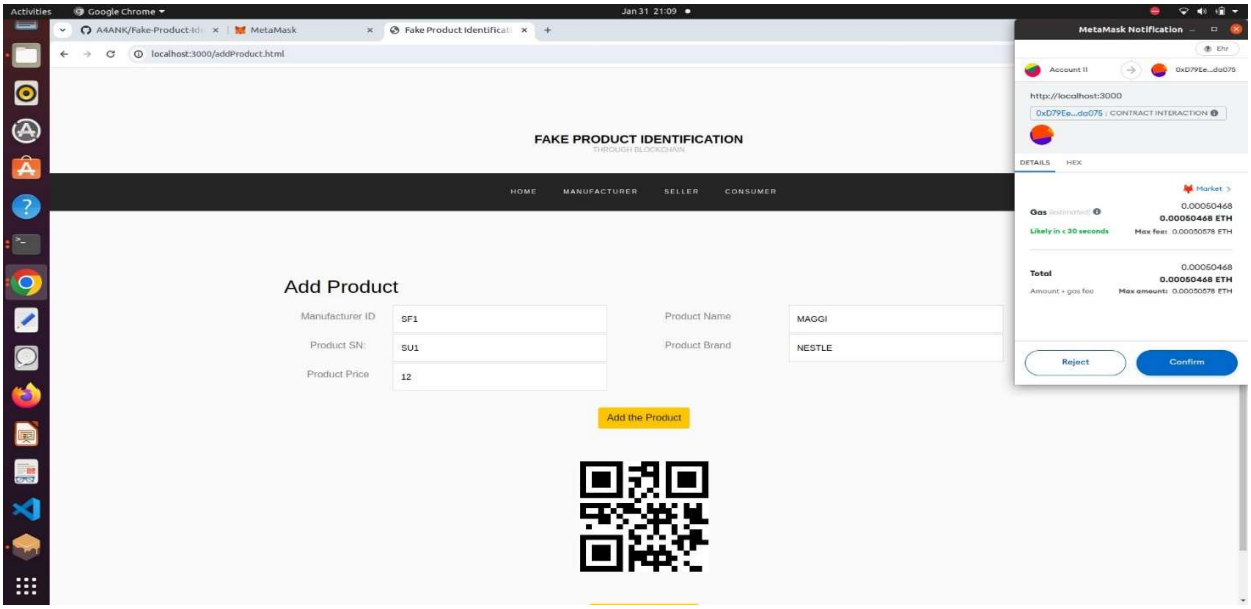

Fig 5.2 :- Add Product

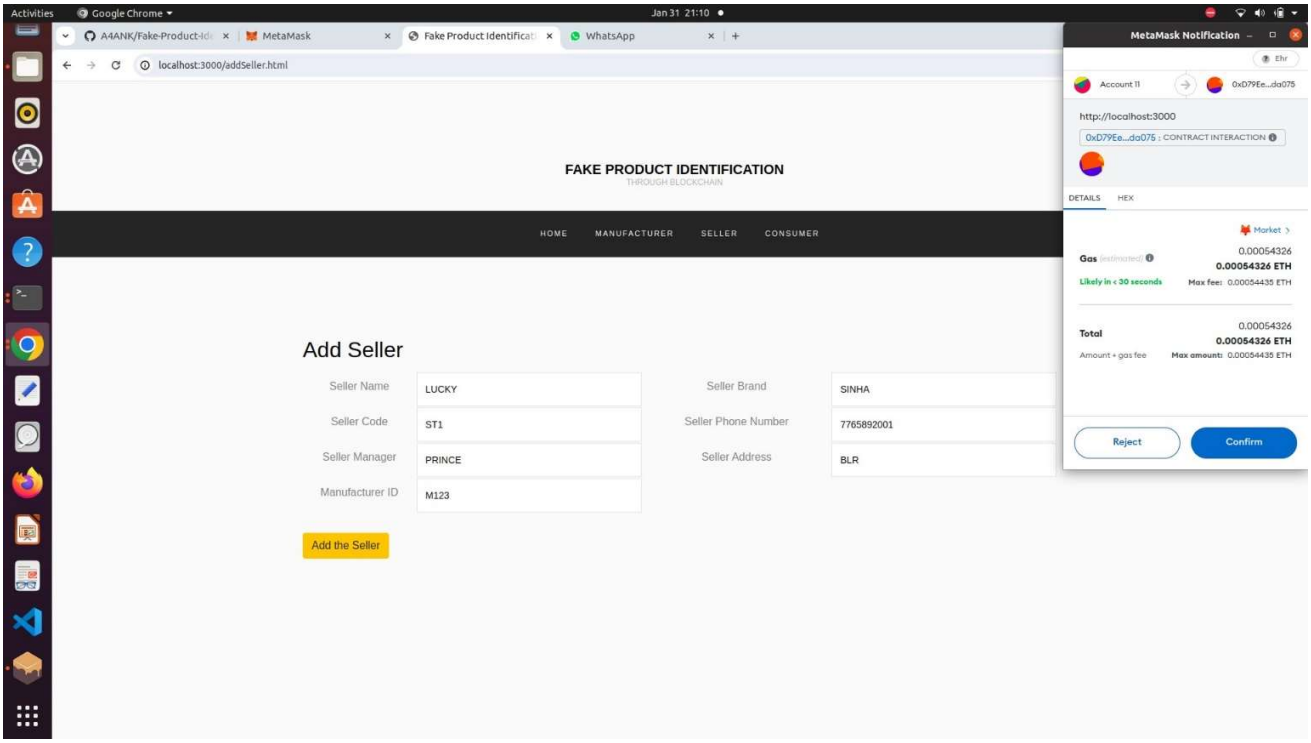

Fig 5.3 :- Add Seller

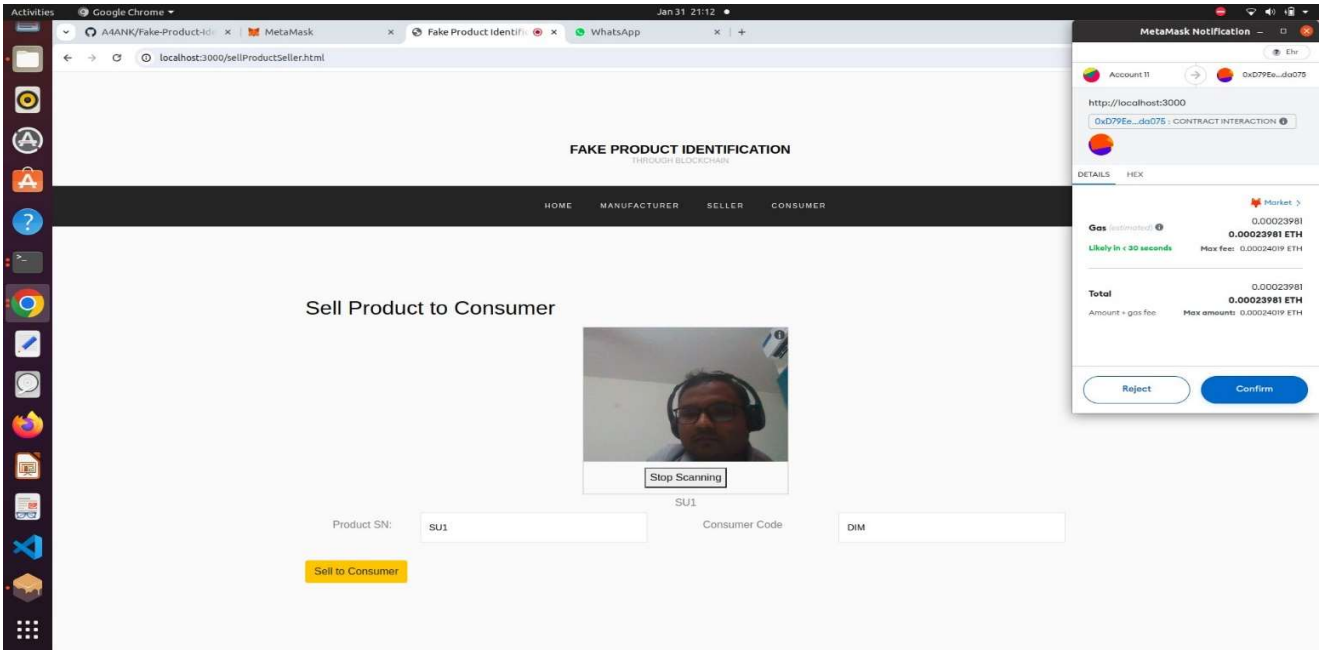

Fig 5.4 :- Sell Product to Consumer

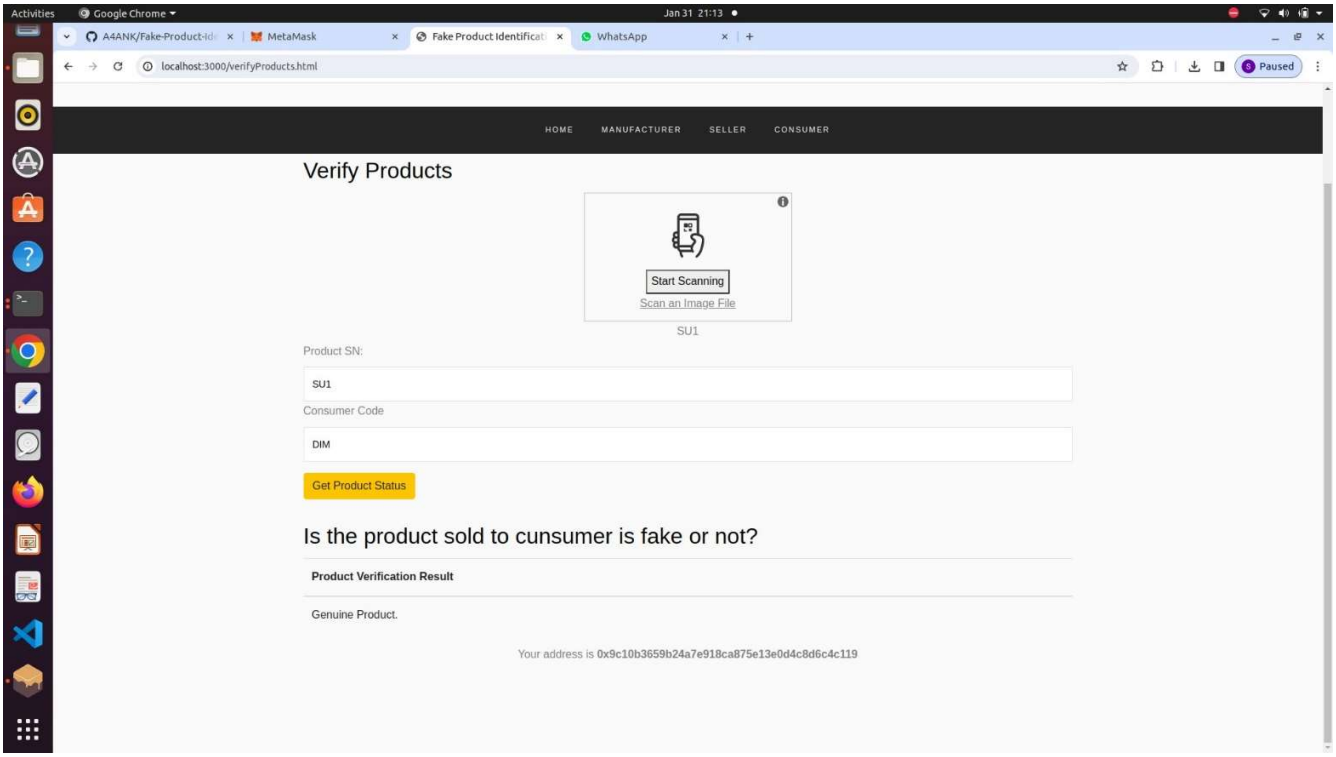

Fig 5.5 :- Verify Product (Genuine)

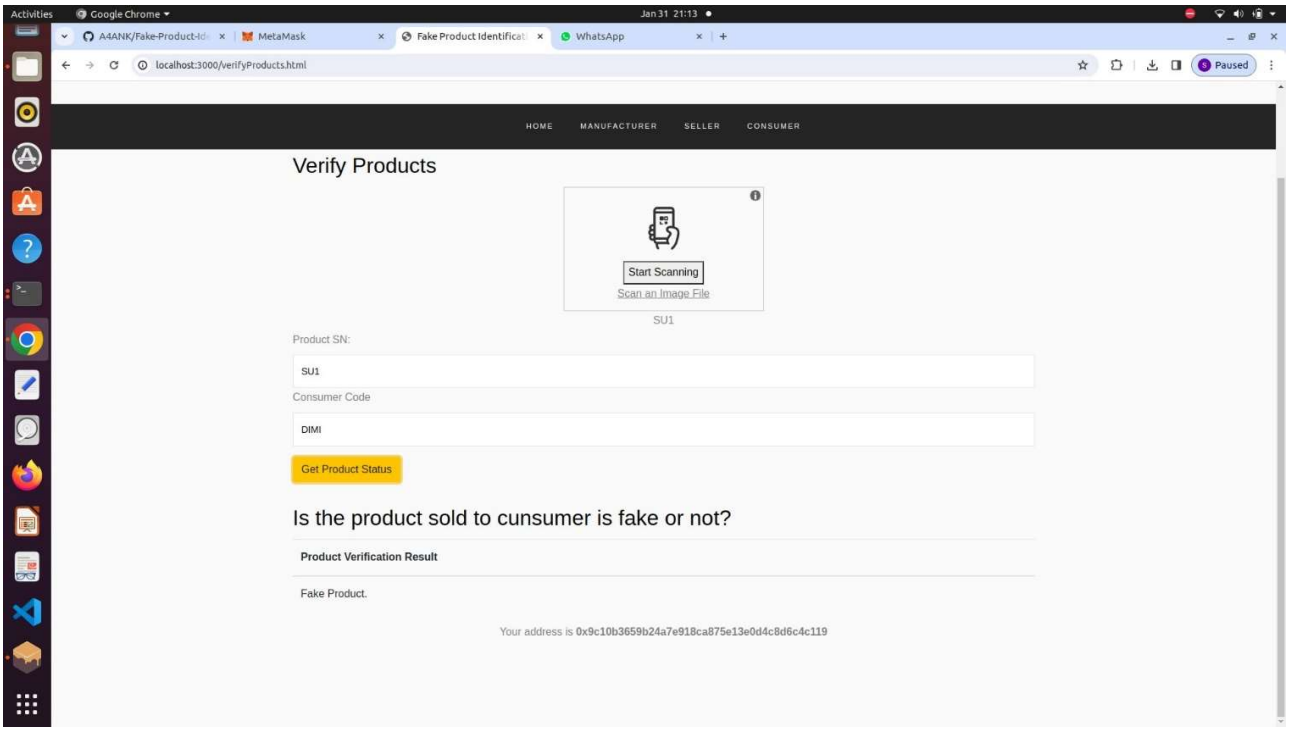

Fig 5.6 :- Verify Product (Fake)

Chapter 6

CONCLUSION AND FUTURE SCOPE

Counterfeiting products are growing exponentially with the enormous amount online. So, there is a strong need to detecting counterfeit products and blockchain technology is used to detect fake products. Furthermore, the information is encoded into a QR code. Customers or users scan the QR code and then they can detect the fake product. Digital information of product can be stored in the form of blocks in blockchain technology. . End user can scan QR code assigned to a product and can get all the information that has been put up throughout the supply chain in the blockchain on which end to end user can check whether the product is genuine or not.

Blockchain technology is still in its general early stages in terms of usage, therefore more research is required. Future work on this framework may be proof of the simplicity of the code. The customer can believe that the proper applications and due to the simplicity of the code, without excessive code, will have additional use. With the experience gained in this project, it is clear that when assigning a chain of blocks that is counterfeit-resistant and traceable, then platform improvement is warranted by a move toward genuine product warranties. This would improve the customer experience by making the entire supply chain framework more open and transparent. In addition, it is hoped that a buyers' intellectual framework can be built on cross-line product tracing, providing more accurate and far-reaching product and seller tracing information to administrative authorities around the world. Obtaining data of such quality will be conceivable with estimates of coordinated efforts, for example, administrative support, framework creation, and data sharing.

REFERENCES

[1] Muhammad Nasir Mumtaz Bhutta, Amir A. Khwaja, Adnan Nadeem, Hafiz Farooq Ahmad , Muhammad Khurram Khan, Moataz A. Hanif, Houbing Song, Majed Alshamari , and Yue Cao , "A Survey on Blockchain Technology: Evolution, Architecture and Security", IEEE special section on intelligent big data analytics for internet of things, services and people,2021, pp. 61048 – 61073.

[2] Rishabh Sushil Bhatnagar, Sneha Manoj Jha , Shrey Surendra Singh, Rajkumar Shende "Product Traceability using Blockchain", 2020 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN

[3] Si Chen , Rui Shi , Zhuangyu Ren , Jiaqi Yan , Yani shi , Jinyu Zhang," A Blockchainbased Supply Chain Quality Management Framework", 2017 IEEE 14th International Conference on e-Business Engineering (ICEBE).

[4] M.C.Jayaprasanna, .V.A.Soundharya , M.Suhana, S.Sujatha," A Block Chain based Management System for Detecting Counterfeit Product in Supply Chain" ,IEEE 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV).

5] Jinhua Ma , Shih-Ya Lin , Xin Chen , Hung-Min Sun,A Blockchain-Based Application System for Product Anti-Counterfeiting" International Journal Of Scientific & Technology Research Volume 8, Issue 12, December 2019 issn 2277-8616.

[6] B. M. A. L. Basnayake, C. Rajapakse," A Blockchain-based decentralized system to ensure the transparency of organic food supply chain" ,IEEE 2019 International Research Conference on Smart Computing and Systems Engineering (SCSE)

[7] Atima Tharatipyakul and Suporn Pongnumkul, "User Interface of Blockchain-Based Agri-Food Traceability Applications", IEEE vol 9, 2019,pp.82909-82929.