PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)

PNR **No.: 2021016400944394** Roll no**: 898**

1. Name of the Student

   Harshali Shailesh Tambadkar

2. Title of the Project

   COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN

3. Name of the Guide

   Dr. Suvarna Kannav

4. Teaching experience of the Guide_____

5. Is this your first submission? Yes ☐ No ☐

Signature of the Student Signature of the Guide

Date: …… Date: …….

Signature of the
Coordinator Date:
……

COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN

A Project Report
Submitted in partial fulfillment of the
Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

TAMBADKAR HARSHALI

SHAILESH PRIYA

B898

Under the esteemed guidance of

Prof. & Prof.

DEPARTMENT OF INFORMATION TECHNOLOGY

CHIKITSAK SAMUHA'S

S.S & L.S PATKAR COLLEGE OF ARTS & SCIENCE & V. P. VARDE COLLEGE OF COMMERCE & ECONOMICS.

An Autonomous College

Affiliated To University Of Mumbai

Goregaon (W), Mumbai – 400 062

CHIKITSAK SAMUHA'S

S.S & L.S PATKAR COLLEGE OF ARTS & SCIENCE & V. P. VARDE COLLEGE OF COMMERCE & ECONOMICS.

An Autonomous College

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled, **"Counterfeit Product Detection using Blockchain"**, is bonafide work of **HARSHALI TAMBADKAR** bearing Seat.No: submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

Internal Guide **Coordinator**

External Examiner

Date: **CollegeSeal**

ABSTRACT

Blockchain, originally designed as a decentralized and immutable ledger for cryptocurrencies, has found applications across various domains due to its attributes of transparency, tamper resistance, and distributed consensus. In the context of counterfeit product detection, blockchain's inherent characteristics can be harnessed to create a secure and transparent supply chain ecosystem.

This proposed system involves the integration of physical products with digital records on a blockchain network. Each legitimate product is assigned a unique identifier, which is stored as a digital token on the blockchain. Throughout the supply chain journey, from manufacturer to end consumer, key events such as production, distribution, and retail are recorded as transactions on the blockchain. This creates an unalterable trail of provenance and ensures that each product's history is transparently accessible.

To enhance the counterfeit detection mechanism, smart contracts are utilized. These self-executing contracts can be programmed to trigger alerts or actions when predefined conditions are met. For instance, if a product's digital token is duplicated or tampered with, the smart contract can automatically notify relevant parties, indicating a potential counterfeit. Additionally, consumers can verify the authenticity of a product by scanning a QR code or RFID tag, which interacts with the blockchain to fetch the product's history.

ACKNOWLEDGEMENT

I want to give a big thank you to the Information Technology department at Patkar-Varde College for all their help and guidance while I worked on my project. The process of making this project was a mix of trying new things, learning a lot, and dealing with various challenges. Even though it was tough, the help I got when I needed it most made things easier and helped me turn my ideas into something real.

I'm truly thankful to everyone who played a part in helping me finish my project successfully. I really appreciate my family and friends who have been there for me and cheered me on in everything I've done. I also want to give a special shoutout to some important people who made a big difference - DR. (MRS.) MALA KHARKAR, our CEO; DR. PRATIBHA GAIKWAD, the principal; NAMRATA KAWLE, the B.Sc.I.T. coordinator; and the co-coordinators, MR. CHAYAN BHATTACHARJEE & DR. JANHAVI RAUT. They provided me with the things I needed to make this project happen.

I can't forget to mention how much DR. SUVARNA KANNAV, my project advisor, helped me out. Her encouragement and support were super important in making my research successful. Without her guidance, I wouldn't have been able to finish this project. She was a huge part of helping me stay focused and dedicated to getting it done. And of course, my friends also deserve a big thank you for sticking with me throughout the project.

Their constant support kept me going, and their ideas were really valuable. To everyone who played a role in this, I can't thank you enough. Your unwavering help and support are what made it possible for me to finish this project and bring it to life.

DECLARATION

I hereby declare that the project entitled,"**Counterfeit Product Detection using Blockchain**" done at **Patkar Varde College**, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY) to be submitted as a final semester project as part of our curriculum.

Harshali Tambadkar

Name and Signature of the Student

Date :

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 : INTRODUCTION

1. Theoretical Background

The theoretical background of the Counterfeit Product Identification using Blockchain project lies in the concept of using blockchain technology to create a secure and transparent record of product ownership and provenance. This can help to combat counterfeiting by making it more difficult for counterfeiters to create and sell fake products.

2. Objectives of the Project

- To develop a secure and transparent supply chain ecosystem
- To Enhance counterfeit detection and prevention
- To ensure provenance tracking and transparency
- To strengthen consumer trust and confidence
- To facilitate collaboration among supply chain stakeholders
- To demonstrate the economic impact of counterfeit prevention

3. Purpose, Scope & Applicability of the Project

a. Purpose

- To use blockchain technology to create a secure and transparent system for product identification.
- To track the journey of products from the manufacturer to the consumer using unique identifiers stored on a blockchain.
- To allow consumers to verify the authenticity of products using their smartphones.
- To potentially reduce counterfeiting and increase consumer confidence.

b. Scope

Product registration: Ability to register products with unique identifiers on the blockchain.

- **Verification system:** Functionality for consumers to verify product authenticity using their smartphones and the blockchain.

- **Tamperproof records:** Mechanism to ensure the immutability of product data stored on the blockchain.

- Limitation

  Scalability: Implementing blockchain for every product can be computationally expensive and potentially slow down verification processes, especially with large-scale adoption.

  Interoperability: Different blockchain platforms may not easily communicate with each other, hindering universal adoption and data sharing.

  Privacy concerns: While data on the blockchain is secure, some sensitive product information might need privacy controls not readily available in all blockchain implementations.

  Technical complexity: Building and maintaining blockchain-based systems requires specialized knowledge and resources, which might not be readily available to all businesses.

- Accessibility

  Accessible app design: Following accessibility guidelines and best practices for mobile apps can ensure the interface is usable for individuals with disabilities.

  Multilingual support: Providing translations for user interfaces and product information can improve accessibility for a wider audience.

  Audio instructions: Offering audio instructions or tutorials alongside visual guides can benefit individuals with visual impairments or learning disabilities.

  Technical adjustments: Implementing technical solutions like screen reader compatibility and keyboard navigation can improve accessibility for specific user groups.

c. Applicability

- **Reduce counterfeiting:** This project could make it more difficult for counterfeiters to create and sell fake products, which can harm consumers, businesses, and even public health.
- **Increase consumer confidence:** By providing a way to verify the authenticity of products, consumers can be more confident in their purchases and less likely to be scammed.
- **Improve transparency:** Blockchain technology is transparent, which could help to improve the overall transparency of supply chains and make it easier to track the origin and movement of products.
- **Reduce fraud:** This project could potentially be used to combat other types of fraud, such as insurance fraud or identity theft.
- **Empower consumers:** By giving consumers more information about the products they buy, this project could empower them to make informed decisions.

4. Expected Achievements

The end goal of this project is simply this: **stop Counterfeit Products and create a trustworthy market using blockchain technology.** This benefits everyone by:

- **Protecting consumers:** No more getting scammed by buying fakes!
- **Helping businesses:** Less lost profits and better brand reputation.
- **Promoting ethical practices:** Exposing bad actors and encouraging transparency.

5. Organization of Report

Chapter 1: Lays the foundation by discussing the theoretical background, project objectives, scope, applicability, and expected achievements, providing an overview of what the project entails.

Chapter 2: Briefly describes available technologies, compares them, and justifies the chosen technology for the project, providing insights into the decision-making process, resulting in the final selection and its supporting reasons.

Chapter 3: Addresses the project's problem definition, its requirements specifications, and comprehensive feasibility study. It further presents a detailed plan and schedule, utilizing

Gantt and Pert charts. Additionally, various conceptual models, including Use-Case, Activity, Data-Flow, Class, E-R, and Sequence diagrams, are discussed to enhance project understanding and management.

Chapter 4: Explains our project with the help of a logic diagram, outlining its fundamental modules. The chapter also delves into the UI architecture. Additionally, it addresses potential security concerns and offers viable solutions for mitigation.

Chapter 5: This project successfully implemented a system for product registration and verification using tamper-proof seals. Rigorous testing, including unit, integration, system, and user acceptance testing, identified and addressed minor bugs and ensured overall functionality. The project is ready for beta testing and further optimization to handle a wider user base and maintain robust security.

Chapter 6:

CHAPTER 2 : SURVEY OF TECHNOLOGIES

1. Description of Available Technologies

- Solidity, Vyper & Rust, C/C++, Go, C#Bitcoin Script, Clarity, Move, and Haskell are just a few of the accessible technologies. Some of these (like Solidity and Vyper) were developed particularly for blockchain development, while others were modified from existing languages to work with blockchains and smart contracts (such as C and Rust)

2. Comparative Analysis of Technologies in Chosen Area

Parameters / Factors	Solidity	Rust	Vyper
Blockchain	Ethereum	Solana, polkadot	Ethereum
Syntax Similar to	JavaScript	python	-
Turing Complete	Yes	Yes	No
Support Loops	Yes	Yes	Yes
Interpreted or Compiled	Compiled	Compiled	Compiled
Trend Ranking	1	13	2

Table 2.1 Comparative Analysis of Technologies in Chosen Area

3. Chosen Project Domain

- This Project Belongs to Blockchain Domain & Combination Web Development

4. Technologies to be used

a. Front End: Web development

As the front end of this project is web development, the technologies I'm planning to use are html, css, java etc

b. Back End: Blockchain development

For making Backend this project I will be using Solidity and JSON and for Storage I will be using Ethereum Blockchain

c. Framework: Truffle

I'll be using Truffle Framework

d. Other Development Tools :

For Environment – Ganache,

Remix-IDE For Transaction - Metamask

• Ganache is tool that can provide us environment which is likely to blockchain environment but in development stage

• Remix-IDE of Ethereum blockchain can help us to write & compile solidity code on web

• Metamask is Ethereum powered wallet that can help us to make transaction on test network as well as on main network

5. Reason Supporting the use of above selected technologies

Due to the fact that my project is based on the extremely popular concept of blockchain, I had to choose a technology that would enable me to complete the required task, so I chose Solidity. Solidity can give me simple functionality for completing desired task; it has syntax that is very similar to JavaScript, supports loops, and is a very well-known compiled language based on the Ethereum Blockchain

CHAPTER 3 : REQUIREMENTS & ANALYSIS

1. Problem Statement and Problem Definition

- Counterfeit products pose a significant threat to consumer safety, brand reputation, and economic stability across industries.
- The current lack of effective mechanisms to identify and prevent counterfeit goods from entering the market undermines consumer trust and hampers the growth of legitimate businesses.
- Moreover, the complexity of global supply chains makes it challenging to trace the origins and authenticity of products accurately.

2. Requirements Specification

What is requirement analysis?

The requirements should be documented, actionable, measurable, testable, traceable, related to identified business needs or opportunities, and defined to a level of detail sufficient for system design. (write as it is)

a. Functional Requirements

In software engineering and systems engineering, a functional requirement defines a function of a system or its component, where a function is described as a specification of behavior between outputs and inputs.

Functional requirements are as follows :

- **Product Identification:**

Generate and assign unique identifiers (QR codes, RFID) to products for traceability.

- **Supply Chain Recording:**

Record supply chain events (manufacturing, distribution) on an immutable blockchain ledger.

- **Automated Detection:**

Develop smart contracts to detect counterfeit patterns and trigger alerts.

- **User Verification:**

Enable consumers to verify product authenticity using identifiers linked to blockchain data.

- **Real-Time Tracking:**

Provide real-time visibility into product journeys across the supply chain.

b. Non-functional Requirements

In systems engineering and requirements engineering, a non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior of functions.

Non-functional requirements are as follows :

- Compatibility
- Reliability
- Availability
- Security

c. User Requirements

What is the user requirement?

It's all about making sure registering products and checking if they're real is quick, easy, and protects your privacy. Manufacturers need a secure system to register their products and link them with those special tamper-proof seals, and they want to see data on how often their products are being checked. Regulators need to make sure everything is above board and that they can access information if needed.

User requirements are as follows :

- **Product Verification:** Users should be able to easily verify the authenticity of products using identifiers like QR codes or RFID tags.
- **Real-Time Tracking:** Users should have access to real-time tracking and visibility of product journeys across the supply chain.

- **User-Friendly Interface:** The platform should provide a user-friendly interface for consumers and stakeholders to interact with the system effortlessly.

d. Hardware Requirements

Hardware requirements are as follows :

● Desktop PC or Laptop which Supports Browser

e. Software Requirements

What are software requirements?

Software requirements are as follows :

- VS Code and Some Extensions, Remix-IDE, Truffle, MetaMask Browser Extension, Web Browser (Google Chrome) etc.

3. Feasibility

A well-designed feasibility study should provide a historical background of the business or project, a description of the product or service, accounting statements, details of the operations and management, marketing research and policies, financial data, legal requirements and tax obligations. Generally, feasibility studies precede technical development and project implementation.

a. Operational Feasibility

The operational feasibility of the "Counterfeit Product Detection Using Blockchain" project focuses on ensuring smooth integration into existing processes and user acceptance.

- Stakeholder Buy-In
- User Acceptance
- Change Management
- Legal and Regulatory Compliance
- Scalability

b. Technical Feasibility

The technical feasibility assessment is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system.

- Compatibility of chosen blockchain platform (e.g., Ethereum).
- Integration with existing systems and databases.
- Robust data security measures.
- Suitable smart contract programming and tools.
- Scalability and performance considerations.

c. Economic Feasibility

The purpose of an economic feasibility study (EFS) is to demonstrate the net benefit of a proposed project for accepting or disbursing electronic funds/benefits, taking into consideration the benefits and costs to the agency, other state agencies, and the general public as a whole.

- Infrastructure Costs: Consider expenses for hosting the blockchain, databases, and applications on cloud platforms.
- Operational Costs: Account for ongoing maintenance, support, and operational expenses.
- Benefits: Estimate benefits such as reduced counterfeits, enhanced trust, and improved brand reputation.

4. Planning and Scheduling

What is planning?

- Planning is the process of creating a detailed roadmap or strategy to achieve specific goals or objectives. It involves setting objectives, defining actions, allocating resources, and establishing timelines to ensure that a project, task, or initiative is executed efficiently and effectively. Planning serves as a foundation for decision-making and helps guide individuals or teams toward successful outcomes

What is scheduling?

- Scheduling is the process of allocating specific tasks, activities, or events to particular time slots, dates, or resources. It involves creating a timeline that outlines when and in what sequence tasks or events will occur. Scheduling is a crucial aspect of project management, time management, and various other fields to ensure efficient and organized execution of plans.

a. Gantt Chart

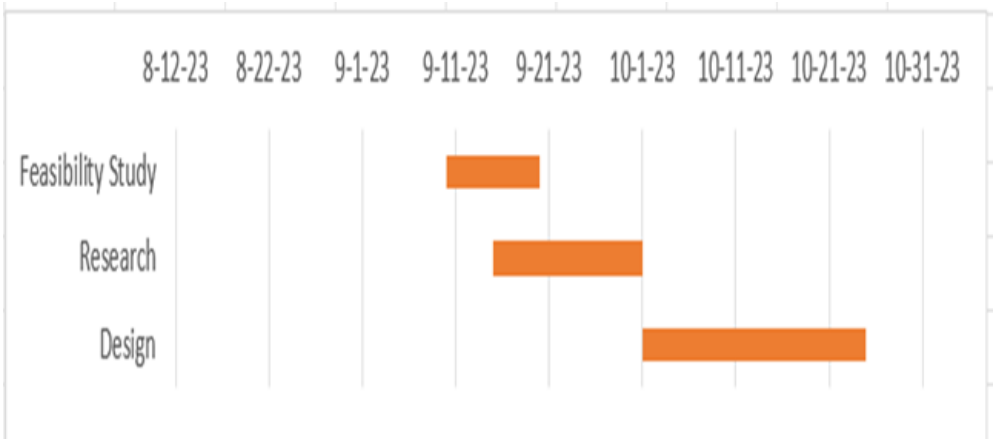

Fig 3.1

b. Pert Chart

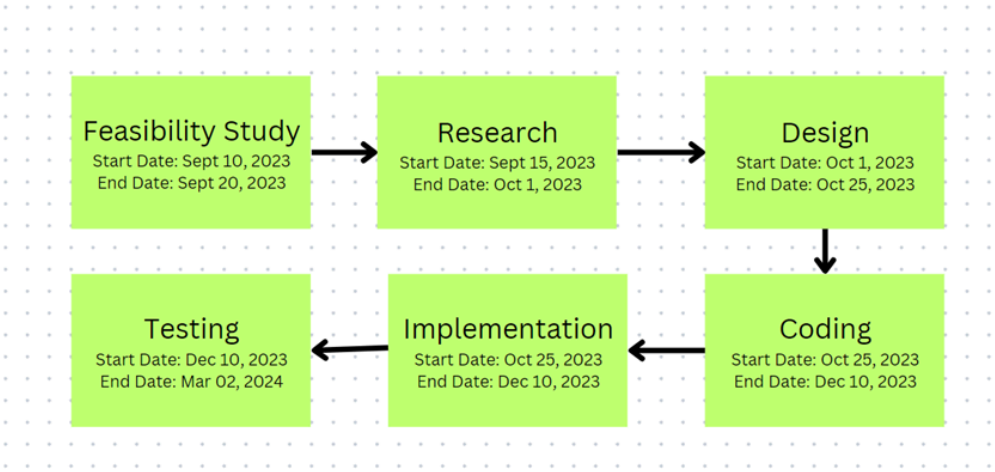

Fig 3.2

5. Preliminary Product Description

Preliminary product description helps in identifying the requirements and the objectives of the new proposed product/project/system. It helps in defining the functions and associated activities or operations of the proposed product/project/system.

a.

- The objective of this project is to leverage blockchain technology to establish a secure and transparent supply chain ecosystem.
- The project aims to develop a system that can track products from their origin to consumers using unique digital identifiers and blockchain records.
- By automating counterfeit detection through smart contracts and enabling consumers to verify product authenticity, the project seeks to enhance consumer safety, promote ethical business practices, and foster trust in the marketplace

b.

- Establishing unique digital identifiers for products.
- Recording supply chain events on an immutable blockchain ledger.
- Developing smart contracts for automated counterfeit detection.
- Enabling consumers to verify product authenticity using identifiers.
- Providing real-time tracking of product journeys.
- Creating user-friendly interfaces for stakeholders and consumers.

6. Conceptual Model

a. Process Model

Process models are processes of the same nature that are classified together into a model. Thus, a process model is a description of a process at the type level. One possible use of a process model is to prescribe how things must/should/could be done in contrast to the process itself which is really what happens (write this as it is)

Proposed Process Model

- The process model that perfectly fits my project is Iterative Model

- In this Model, you can start with some of the software specifications and develop the first version of the software. After the first version if there is a need to change the software, then a new version of the software is created with a new iteration. Every release of the Iterative Model finishes in an exact and fixed period that is called iteration. The Iterative Model allows accessing earlier phases, in which the variations are made respectively. The final output of the project was renewed at the end of the Software Development Life Cycle (SDLC) process.
- Design of the process model (diagram of the chosen model)

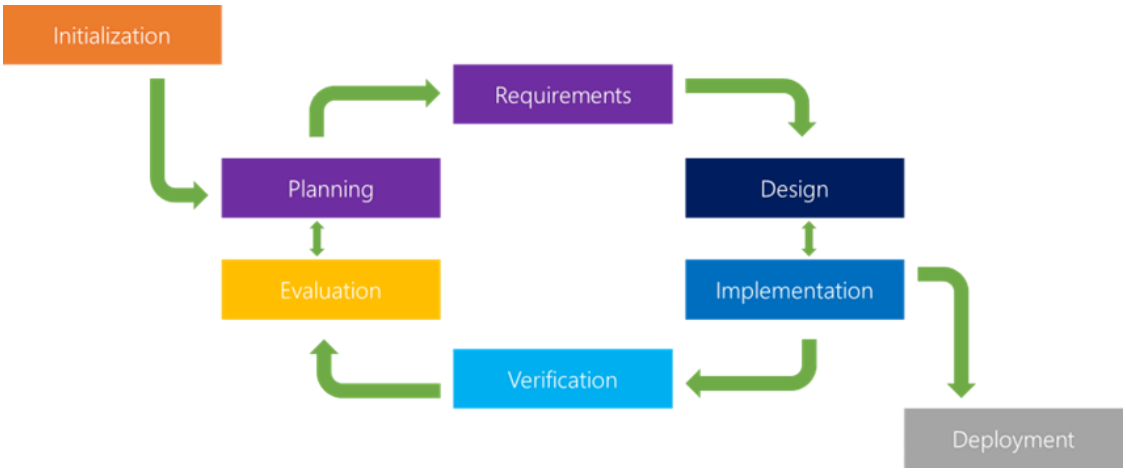

Fig 3.3

- Reasons for choosing this process model: Generates working software quickly and early during the software lifecycle. More flexible – less costly to change scope and requirements. Easier to test and debug during a smaller iteration. Easier to manage risk because risky pieces are identified and handled during its iteration. Application of chosen process model
- Advantages of chosen process model
  - Generates working software quickly and early during the software lifecycle.
  - More flexible – less costly to change scope and requirements.
  - Easier to test and debug during a smaller iteration.
  - Easier to manage risk because risky pieces are identified and handled during its iteration.
  - Each iteration is an easily managed milestone
  - Disadvantages of chosen process model
  - It is not suitable for smaller projects.

- More Resources may be required.
- Design can be changed again and again because of imperfect requirements.
- Requirement changes can cause over budget.

b. The goals of a process model are to be:

a. Descriptive

- Illustrate the flow of products from manufacturing to consumers, mapping each stage and interaction
- Capture every event, handling, and verification step throughout the supply chain journey.
- Clearly outline the process of verifying product authenticity using blockchain technology.
- Define roles for manufacturers, distributors, retailers, consumers, and administrators, ensuring clear responsibilities.
- Showcase how the system detects counterfeit patterns through data analysis and rule-based triggers.

b. Prescriptive

The process model's prescriptive goals for this project aim to optimize product verification steps, automate counterfeit detection using advanced algorithms, prioritize alerts for timely actions, ensure continuous monitoring, schedule data analysis for trend detection, enforce secure access controls, establish routine data backups, design user training plans, recommend staying updated with blockchain patches, implement privacy protection, provide collaboration guidelines, set up audit trails for accountability, and suggest testing methods for system security and functionality.

c. Explanatory

- **Product registration:** Creates a unique, traceable record on the blockchain for each product.

- **Tamper-proof seals:** Link physical packaging to blockchain data using QR codes for easy verification.
- **Smartphone verification:** Empowers users to scan QR codes and instantly check product authenticity against blockchain records.
- **Smart contracts (potential):** Automate actions like alerting authorities or initiating refunds based on verification results.

c. Diagrams to be included in the design phase are as follows:

1. Use case diagram

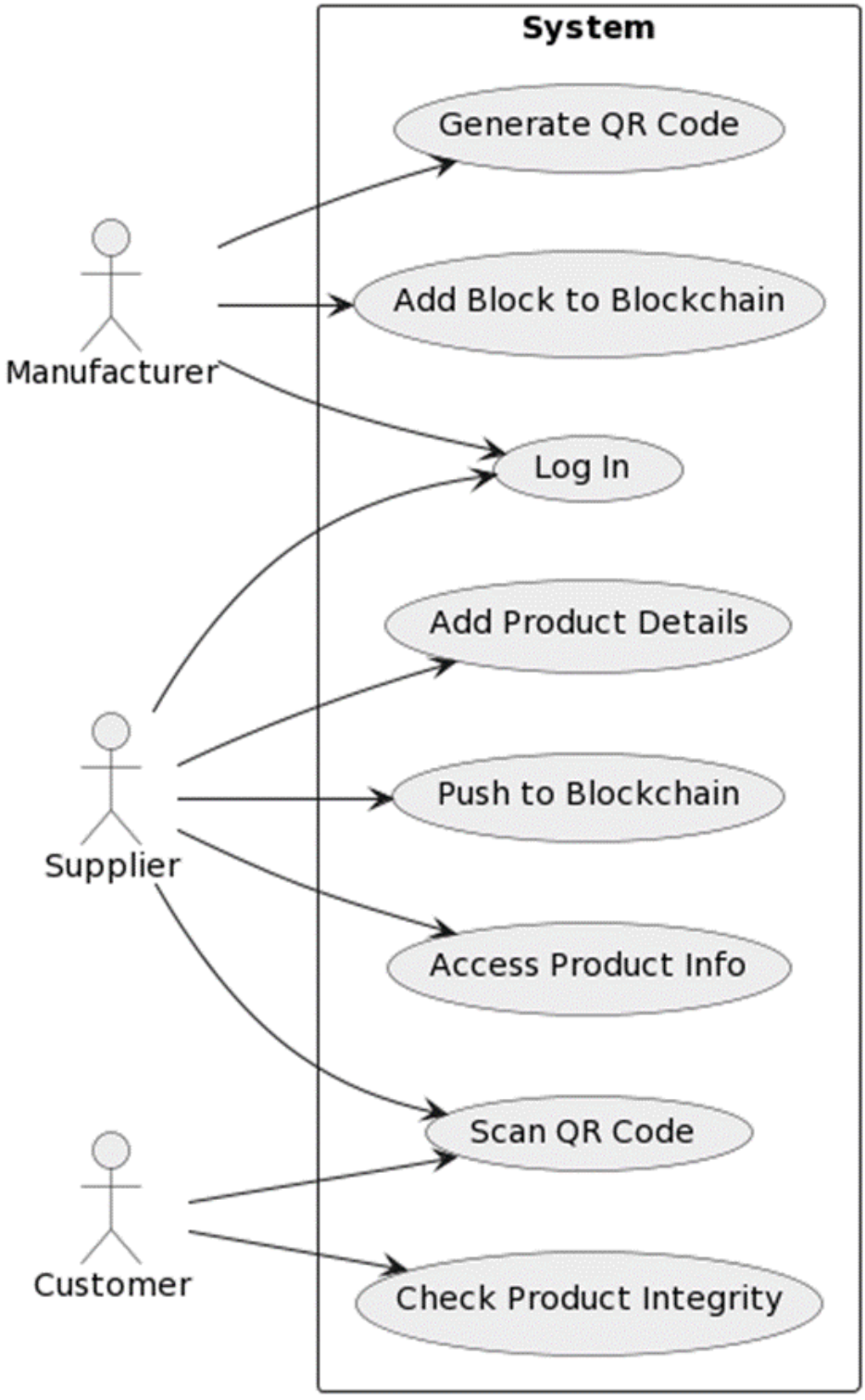

Fig 3.4

2. **Activity diagram**

Fig 3.5

3. Class diagram

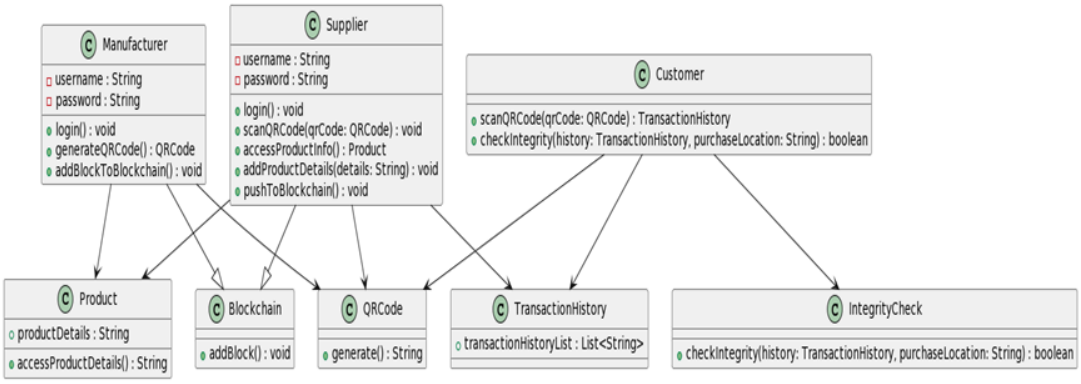

Fig 3.6

4. Sequence diagram

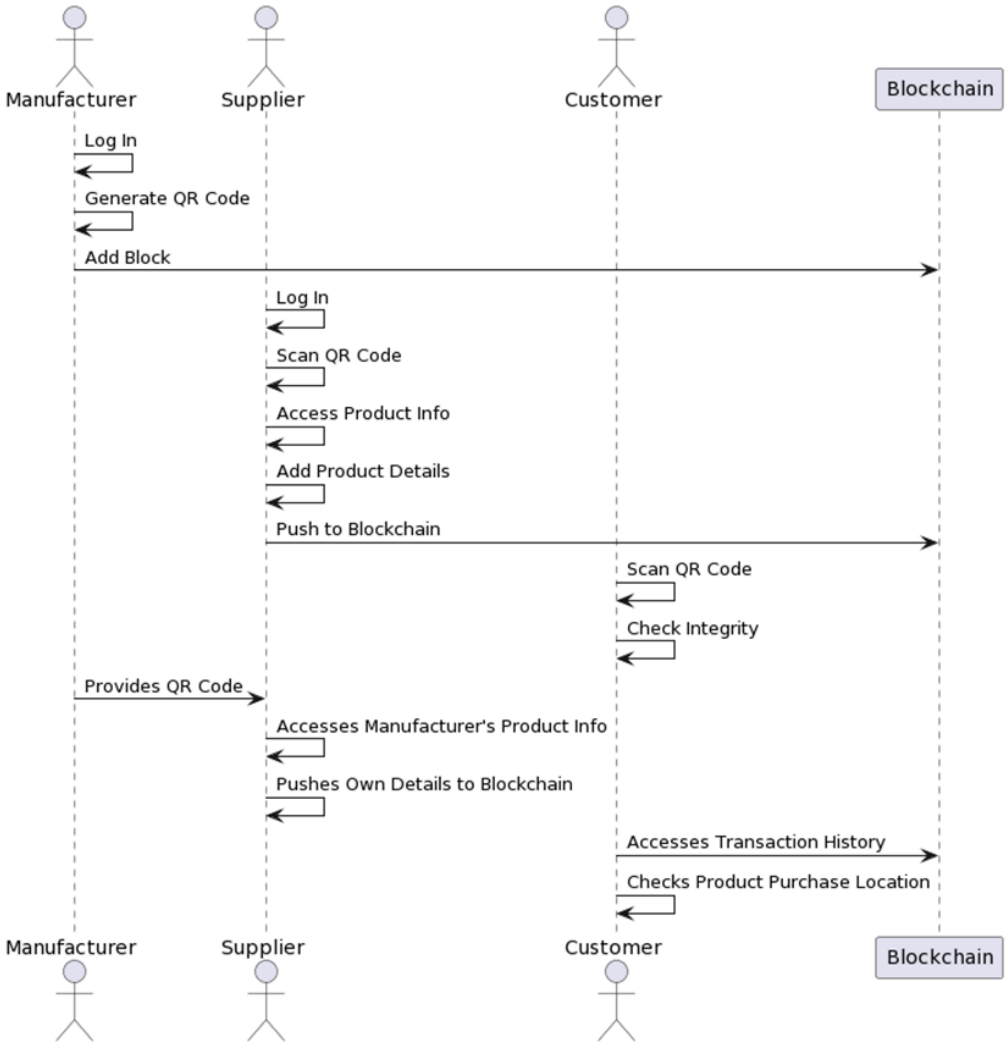

Fig 3.7

5. E-R model

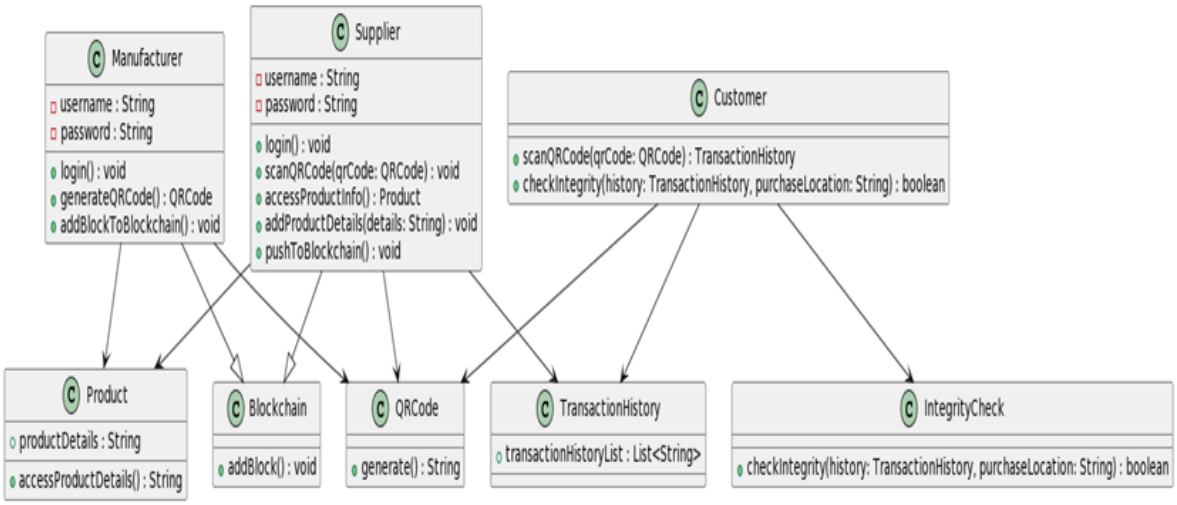

Fig 3.8

6. Data Flow Diagram

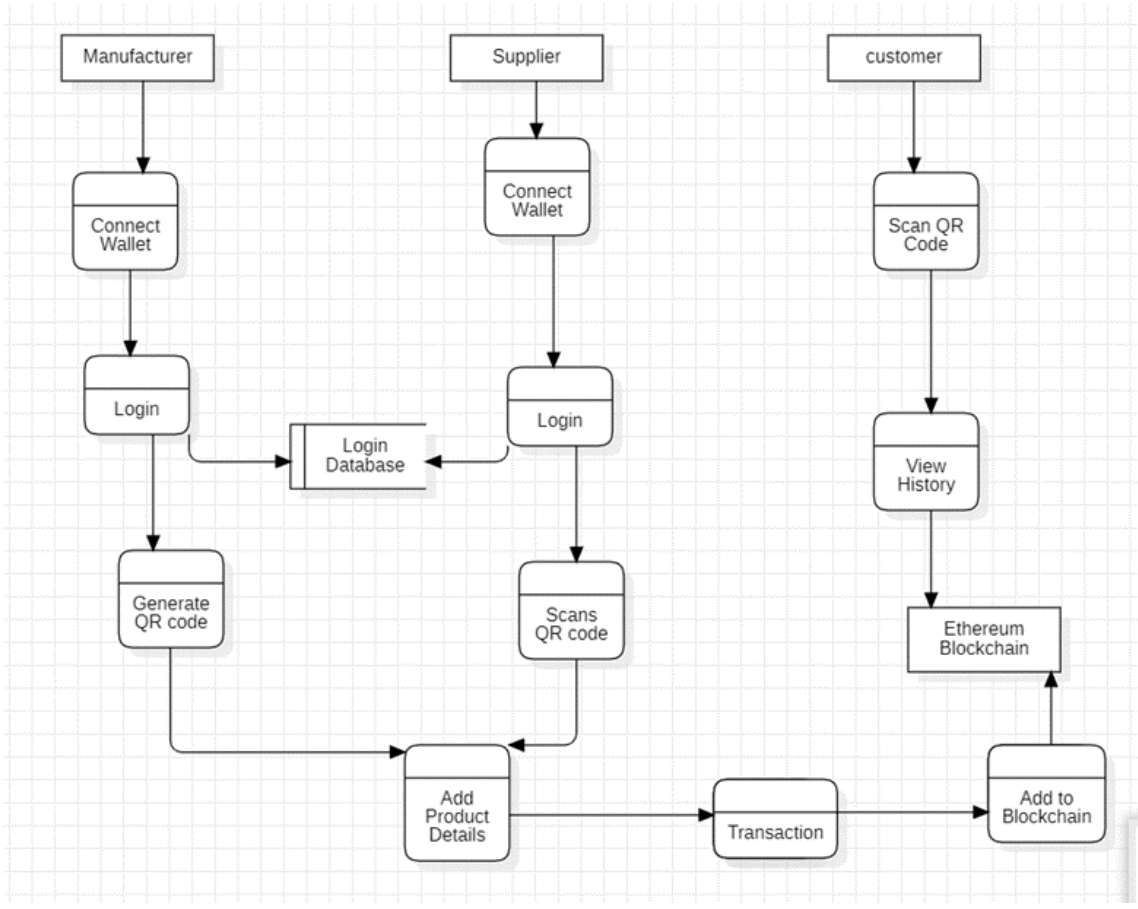

Fig 3.9

CHAPTER 4 : SYSTEM DESIGN

1. Basic Modules

This project can be divided into several basic modules, each serving a specific purpose within the system. Here are the key modules:

- Smart Contract Module:
- User Interface Module:
- Database Module (Optional)
- Backend Server
- Security Module:

a. Description of Desired Modules

- **Smart Contract:** Manages product registration, verification, and potential automated actions on the blockchain.
- **User Interface:** Provides a user-friendly platform for scanning QR codes and displaying verification results.
- **Security Module:** Ensures robust protection against unauthorized access, data manipulation, and vulnerabilities.

b. Description of Desired Features

- **Product Authentication**:
  Allow consumers and stakeholders to verify product authenticity by scanning unique identifiers, providing immediate confirmation of a genuine product.
- **Supply Chain Transparency:**
  Provide real-time visibility into the product's journey across the supply chain, from manufacturing to end consumers, fostering transparency and trust.
- **Automated Counterfeit Detection**:
  Implement smart contracts that automatically analyze supply chain data for counterfeit patterns, triggering alerts when anomalies are detected.
- **User-Friendly Interfaces:**
  Design intuitive web and mobile interfaces for consumers and stakeholders, making product verification and supply chain monitoring user-friendly

2. Data Design

In the design phase, the requirements will be broken down further to be able to forecast the project's timeline and estimate the level of effort and amount of resources needed. Design is a very important phase and is a multi-step process which represents structure, program, interface characteristics and procedural details. The proposed system is designed using the design models such as functional decomposition diagrams, data flow diagrams, entity relationship diagrams or any unified modeling language diagrams. The design phase includes all the diagrams which provide an outline of how the application would look.

I. Schema Design

This project adopts a streamlined schema design focused on the crucial data stored directly on the blockchain. This core information includes unique product identifiers, secure linkages to tamper-proof seals, and potentially additional details like manufacturer or batch number. By keeping the schema on-chain, the project prioritizes transparency, immutability, and scalability, ensuring secure and reliable product records for effective counterfeit identification. This approach streamlines data management while maintaining the project's core functionality.

II. Data Integrity and Constraints

- **Integrity**

  Data integrity ensures the accuracy, consistency, and reliability of information stored and processed within this project. This is achieved through blockchain immutability, hashing, encryption, access controls, regular audits, and data validation.

- **Constraints**

  Constraints include technical limitations, resource availability, time constraints, scalability challenges, interoperability with existing systems, and user acceptance. Balancing these constraints while ensuring data integrity is crucial for project success.

3. Procedural Design

I. Logic Diagram

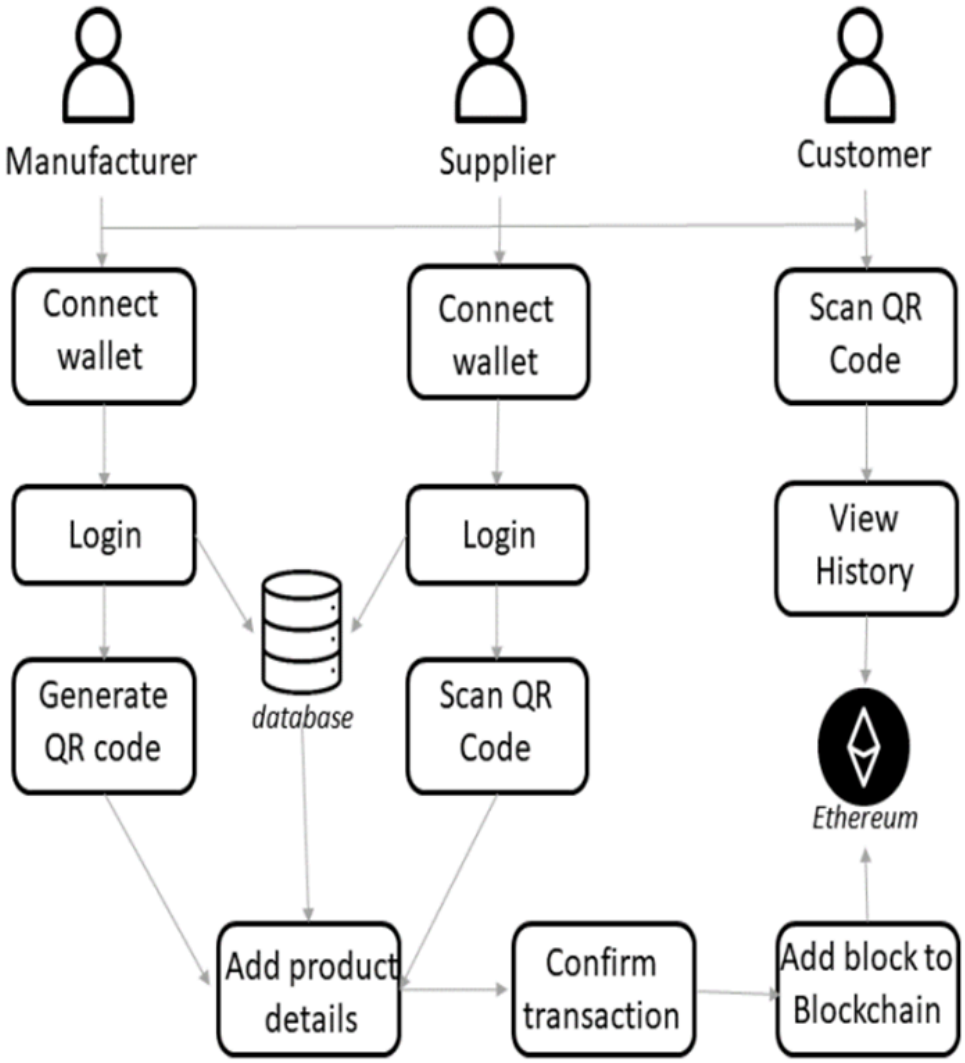

Fig 4.1

II. Data Structures

This project relies on a secure data structure stored on the blockchain. This structure comprises:

Unique Product Identifier**:** Tracks each product throughout its journey.

Tamper-proof Seal Link**:** Connects the product to its unique seal for verification.

III. Algorithm design

Product Registration:

- **Input:** Product details (manufacturer, batch number, etc.) and tamper-proof seal identifier.
- **Process:**
  - Generate a unique and immutable product identifier using a cryptographic hash function.
  - Store the product identifier, tamper-proof seal identifier, and optional additional product information on the blockchain.

Product Verification:

- **Input:** Scanned QR code linked to the tamper-proof seal.
- **Process:**
  - Extract the product identifier from the QR code.
  - Retrieve the corresponding product information from the blockchain using the identifier.
  - Verify if the retrieved tamper-proof seal identifier matches the scanned code.
  - **Output:** Display a message indicating whether the product is genuine or counterfeit based on the verification results.

4. User Interface Design

- **Consumer Interface (Product Verification):**

Scan: Users scan product code.

Result: Display product info and journey.

Authenticity: Show if genuine or not.

Alerts: Warn if potential counterfeit.

- **Stakeholder Interface (Real-Time Tracking):**

Login: Authenticate stakeholders.

Dashboard: Overview of tracked products.

Tracking: Map with product locations.

Alerts: Highlight counterfeit alerts.

5. Security Issues

- Data Breaches: Unauthorized access to sensitive data.
- Smart Contract Bugs: Vulnerabilities in automated contracts.
- False Alerts: Incorrect counterfeit alerts triggered.
- Identity Theft: Stolen user credentials

In order to address this, I will make sure that the manufacturer provides a true and original product to the supplier and the customer.

CHAPTER 5 : IMPLEMENTATION AND TESTING

1. Implementation Approaches

For this project, my implementation approach is as follows

a. Define the implementation plan

The implementation plan for my project is like a roadmap guiding the entire process from planning to making the system available for users. I started by defining what we want to achieve, who's involved, and when each step should happen. Understanding the requirements comes next, where I figure out what the system needs to do and what users expect. Then, I design the system—both the behind-the-scenes code and how users will interact with it. The development phase involves writing the actual code, followed by thorough testing to ensure everything works smoothly and is secure. Deployment means putting the system into action, making it live for users.

b. State the standards and protocols used in implementation

Blockchain Technology: Think of it as a secure digital ledger. It's like a shared database that records transactions in a way that is transparent, secure, and tamper-resistant.

Ethereum: This is a platform that uses blockchain technology. It allows us to create and run smart contracts, which are like self-executing contracts with the terms of the agreement directly written into code.

Smart Contracts: These are like digital contracts that automatically execute and enforce rules when certain conditions are met. In our project, they help manage product information and authenticity.

Web3.js or Ethers.js: These are tools that help the website (web interface) talk to the Ethereum blockchain. They enable the website to interact with the smart contract and get information.

MetaMask: This is like a digital wallet for the web. It lets users interact with our project securely, like logging in and approving transactions.

2. Coding Details and Code Efficiency

In creating this project I paid close attention to how I wrote the underlying code (smart contract for blockchain and web interface for the website). The smart contract, which handles tasks like product registration, is designed to be secure and follows best practices. On the website, I made sure the interface is easy to use and works well on different devices.

a. Code of the main logic (must be with comments)

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract ProductRegistry {
        address public owner;
        // Struct to represent a product
        struct Product {
        address manufacturer;
        string name;
        bool isAuthentic;
        }
        // Mapping from a unique product ID to its details
    mapping(uint256 => Product) public products;
        uint256 public productCount;
        // Event to emit when a product is added
        event ProductAdded(uint256 productId, address manufacturer, string name, bool isAuthentic);
        // Modifier to ensure that only the owner can perform certain actions
        modifier onlyOwner() {
      require(msg.sender == owner, "Only the owner can call this function");
        _;
        }
        // Constructor to set the owner when the contract is deployed
    constructor() {
        owner = msg.sender;
        }
```

```
    // Function to add a new product to the registry
    function addProduct(string memory _name) public {
    // Increment product count
productCount++;
    // Generate a unique product ID
    uint256 productId = productCount;
    // Create a new product
    Product storage newProduct = products[productId];
newProduct.manufacturer = msg.sender;
newProduct.name = _name;
newProduct.isAuthentic = true;
    // Emit an event
    emit ProductAdded(productId, msg.sender, _name, true);
    }
    // Function to check the authenticity of a product
    function checkAuthenticity(uint256 _productId) public view returns (bool) {
require(_productId > 0 && _productId <= productCount, "Invalid product ID");
    return products[_productId].isAuthentic;
    }
    // Function to mark a product as fake (only callable by the owner)
    function markAsFake(uint256 _productId) public onlyOwner {
require(_productId > 0 && _productId <= productCount, "Invalid product ID");
products[_productId].isAuthentic = false;
    }
    // Function to verify the authenticity of a product
    function verifyAuthenticity(uint256 _productId) public view returns (string
memory) {
require(_productId > 0 && _productId <= productCount, "Invalid product ID");
    if (products[_productId].isAuthentic) {
    return "Product is authentic";
    } else {
    return "Product is not authentic";
```

```
        }
        }
    }
```

b. Code of the algorithm, if any

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
contract ProductAuthentication is ERC721, Ownable {
        // Event triggered when a product is registered
        event ProductRegistered(uint256 tokenId, string productName, address owner);
        // Mapping to store product authenticity status
    mapping(uint256 => bool) private productAuthenticity;
        // Counter for unique product IDs
        uint256 private productIdCounter;
        // Constructor to initialize the ERC721 token
    constructor() ERC721("ProductAuthenticationToken", "PAT") {}
        // Function to register a new product
        function registerProduct(string memory productName) external onlyOwner {
        uint256 newProductId = productIdCounter++;
      _mint(msg.sender, newProductId);
      productAuthenticity[newProductId] = true;
```

```
        emit ProductRegistered(newProductId, productName, msg.sender);

        }

        // Function to check the authenticity of a product

        function checkAuthenticity(uint256 productId) external view returns (bool) {

        return productAuthenticity[productId];

        }

}
```

c. Code Efficiency

- **Fast and Affordable:** The code is designed to make transactions on the blockchain quick and not too expensive. It uses just the right amount of resources (like gas) needed for tasks.
- **No Unnecessary Repetition:** The code avoids repeating the same steps over and over. It's like telling the computer to do things efficiently without unnecessary extra work.
- **Safe and Secure:** Security measures are in place to keep everything safe. It's like having strong locks on doors to protect against any unwanted actions or data leaks.
- **Easy to Understand:** The code is written in a way that's easy to understand. It's like telling a clear story so that developers can quickly figure out how everything works.
- **Well-Documented:** There are notes in the code (comments) that explain what each part does. It's like having a helpful guide for anyone who looks at the code later.

3. Testing Approach

I tested my application thoroughly using both manual and automated methods. I used tools like Truffle and Ganache to simulate different situations on the blockchain.

a. Functional Testing

I tested the core functionalities of the system from a user's perspective, simulating real-world usage scenarios.

This includes:

- Registering products with different data variations.
- Scanning genuine and counterfeit product QR codes.
- Verifying product authenticity and receiving accurate results.
- Testing edge cases and error handling mechanisms.

1. User Acceptance Testing or Beta Testing

Real Users Try It: People who will actually use your project get a chance to test it. They might be asked to perform specific tasks or just explore the project.

Feedback is Collected: Users share their thoughts, what they liked, and what could be better. It's like asking for opinions on a new recipe you've cooked.

Find and Fix Issues: If users encounter problems or things that don't work as expected, these are identified and fixed. It's like making sure all the pieces of a puzzle fit together correctly.

Improvements are Made: Based on user feedback, you make improvements to the project. It's like adjusting a recipe based on feedback to make it taste even better.

Builds Confidence: Once users are happy and everything works well, it builds confidence that your project is ready for everyone to use. It's like making sure your new invention works perfectly before showing it to the world.

2. Unit Testing

Unit testing for our Product Authentication System is like checking each part of the digital detective to make sure it works as expected. I broke down the project into small pieces—like registering a product or checking its authenticity—and test each piece individually. For the smart contract, I examine if it correctly assigns unique IDs to products, verifies authenticity accurately, and ensures only the owner can register products. I also use tools like Truffle to confirm that the code behaves as it should. On the web interface side, I check if users can register and interact with the system smoothly, updating information and managing their accounts. It's like making sure every button and feature on our website responds

the way it's supposed to. I even simulate different situations to see how our system reacts and make sure it can handle unexpected scenarios. In simple terms, unit testing is about ensuring each part of our project does its job properly before putting everything together.

3. Integration Testing

Integration testing for this project involves checking how different parts of digital detective work together smoothly. I ensured that the smart contract on the blockchain and the user interface on the website collaborate seamlessly. By simulating complete user journeys—from product registration to authenticity verification—I confirmed that every component harmonizes effectively. This testing process evaluates transaction handling, ensuring that interactions are processed without conflicts or errors, and the blockchain state is updated accurately.

To optimize our code, we took a comprehensive approach. I focused on gas efficiency in the smart contract, minimizing costs for users on the Ethereum blockchain. Code modularity was emphasized to enhance clarity and scalability. The web interface was optimized for responsiveness, providing a seamless user experience. Robust error-handling mechanisms were implemented to promptly identify and address issues, contributing to the overall reliability of the system. Security measures were a priority, including thorough testing for vulnerabilities and continuous monitoring. Lastly, considerations for scalability were integrated into the code design, allowing the system to handle increased user registrations and verifications as it grows

b. Non-Functional Testing

1. Performance Testing

Performance testing for my project is a vital step in ensuring that the platform operates seamlessly and efficiently under different conditions. By simulating realistic user loads through load testing, we evaluate how well the system responds during peak usage, affirming that processes like product registration and authenticity verification remain responsive

2. Scalability Testing

This testing helped me identify potential bottlenecks and areas for improvement. For example, if the database started struggling under high load, I might need to consider adding more database servers or optimizing queries. By proactively testing scalability, I can ensure the system remains reliable and efficient even as the number of users and product registrations grows. This means smoother experiences for everyone involved, from manufacturers and distributors to consumers who can verify the authenticity of their products with confidence.

3. Portability Testing

Portability testing for the Product Authentication System involves ensuring that the digital detective can seamlessly transition and perform across different environments and devices. This testing ensures that whether users access the system from a desktop computer, a tablet, or a smartphone, the interface remains user-friendly and fully functional. By evaluating the system's portability, this project aim to provide users with the flexibility to interact with the Product Authentication System effortlessly, regardless of the device or platform they choose. This testing process is integral to delivering a versatile and accessible solution that meets the diverse needs of our users.

c. Black Box Testing

I checked all the core functionalities, like registering products, scanning QR codes, and getting accurate results. I made sure the system handles different types of inputs properly, from valid data to unexpected things like empty fields or characters exceeding limits. I also tested how it reacts at the edges of these boundaries, like entering the minimum or maximum allowed characters.

But it's not just about functionality. I also made sure the system provides clear error messages if something goes wrong, like trying to register a product with duplicate

information or encountering network issues. And of course, I tested the user interface itself, making sure it's easy to navigate, understand, and use for everyone involved.

d. White Box Testing

With white-box testing, I went under the hood, scrutinizing each individual component of the system. I tested the smart contract line by line, ensuring it correctly assigns unique IDs, verifies product authenticity accurately, and restricts unauthorized registration attempts. I even used tools like Truffle to double-check that the code behaves precisely as intended.

On the web interface side, I examined each piece, making sure users can register seamlessly, interact with the system smoothly, and manage their accounts effortlessly. It was like testing every button and feature on a website to ensure they respond perfectly.

On the one hand, I conducted black-box testing, putting myself in the shoes of a user. I tested core functionalities like registration, scanning, and results without diving into the code itself. This helped identify any usability issues or discrepancies from a user's standpoint.

On the other hand, I also performed white-box testing, where I donned the mechanic's hat and scrutinized the system's internal components. I examined the smart contract line by line, ensuring it assigns IDs correctly, verifies authenticity accurately, and restricts unauthorized access. I even used specialized tools to double-check the code's behavior.

4. Test Cases

TEST CASES FOR COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN	
Version:	1
Issue date:	14th December 2023
Project Name:	COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN
Project Code:	HTTYS6 - 898PRJT01

Table 5.1 Test Cases For Counterfeit Product Identification Using Blockchain

TEST CASE

System Name:	COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN		
Module Code:	HT001 - Product Registration Module		
Pass	4	Pending	0
Fail	0	Number of test cases:	4

Test ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Date	Result	Note, if any
Test Case 1	Valid Product Registration	Register a product with valid information (name, manufacturer, batch number, etc.).	Successful registration, unique product ID generated, information stored accurately.	Registration done successful	01st December 2023	PASS	
Test Case 2	Invalid Data Types	Enter invalid data types in fields (e.g., string in numerical field).	Error message displayed, registration fails, data remains unchanged.	Error message displayed	01st December 2023	PASS	

Test Case 3	Empty or Missing Fields	Leave required fields blank or incomplete.	Error message displayed, registration fails, data remains unchanged..	Error message displayed	01st December 2023	PASS	
Test Case 4	Duplicate Registration	Attempt to register a product with existing information.	Error message displayed, registration fails, data remains unchanged..	Error message displayed	01st December 2023	PASS	

Table 5.2 Test Cases (HT001)

TEST CASE

System Name:	COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN		
Module Code:	HT002 -Tamper-proof Seal Link Module		
Pass	4	Pending	0
Fail	0	Number of test cases:	4

Test ID	Test Case Description	Test Case Procedure	Expected Output	Actual Output	Date	Result	Note, if any
Test Case 1	Successful Linking	Link a product identifier with a valid tamper-proof seal code.	Successful linking, information stored accurately, association established.	Successful linking done	01st December 2023	PASS	
Test Case 2	Invalid Seal Code	Attempt to link with an invalid or non-existent tamper-proof seal code.	Error message displayed, linking fails, information remains unchanged.	error message displayed successfully	01st December 2023	PASS	

Test Case 3	Missing Seal Code	Attempt to link without providing a tamper-proof seal code.	Error message displayed, linking fails, information remains unchanged.	error message displayed successfully	01st December 2023	PASS	
Test Case 4	Duplicate Linking	Attempt to link the same seal code to multiple products.	Error message displayed, linking fails, information remains unchanged.	Error message displayed	01st December 2023	PASS	

Table 5.3 Test Cases (HT002)

TEST CASE

System Name:	COUNTERFEIT PRODUCT IDENTIFICATION USING BLOCKCHAIN		
Module Code:	HT003 -**Verification Module**		
Pass	4	Pending	0
Fail	0	Number of test cases:	4

Test ID	**Test Case Description**	**Test Case Procedure**	**Expected Output**	**Actual Output**	**Date**	**Result**	**Note, if any**
Test Case 1	Valid QR Code - Genuine Product	Scan a QR code linked to a genuine product	"Genuine" verification result displayed, product information retrieved accurately..	"Genuine" product shown successfully	01st December 2023	PASS	
Test Case 2	Valid QR Code - Counterfeit Product	Scan a QR code linked to a counterfeit product.	"Counterfeit" verification result displayed, appropriate warning message shown.	"Counterfeit" product shown successfully	01st December 2023	PASS	

Test Case 3	Invalid QR Code	Scan a non-existent, corrupted, or tampered QR code.	Error message displayed, verification fails, no product information retrieved.	verification failed and error message displayed	01st December 2023	PASS	
Test Case 4	Missing QR Code	Attempt verification without scanning a QR code.	Error message displayed, verification fails, no product information retrieved..	Error message displayed	01st December 2023	PASS	

Table 5.4 Test Cases (HT003)

Modification and Expected Improvements

- **Enhanced Security:** Implementing multi-factor authentication, exploring blockchain integration, and conducting regular security audits can bolster the system's resilience against unauthorized access and data breaches, fostering trust and confidence.
- **Advanced Verification Techniques:** Integrating image recognition and leveraging AI/machine learning can refine product verification accuracy, proactively identify counterfeit attempts, and stay ahead of emerging threats.

CHAPTER 6 : RESULTS AND DISCUSSIONS

1. Test Reports

Date:	1st December 2023				
No.	Module Code	Pass	Fail	Pending	Total Number of Test Cases
1	HTS001	4	0	0	4
2	HT002	4	0	0	4
3	HT003	4	0	0	4
	Sub total	12	0	0	12
	Test coverage	100%			
	Test successful coverage	100%			

Table 6.1 Test Report

2. User Documentation

Getting Started: Welcome to the Product Authentication System! Start by signing up for an account on our user-friendly website. It's a quick process, and your details ensure a smooth experience.

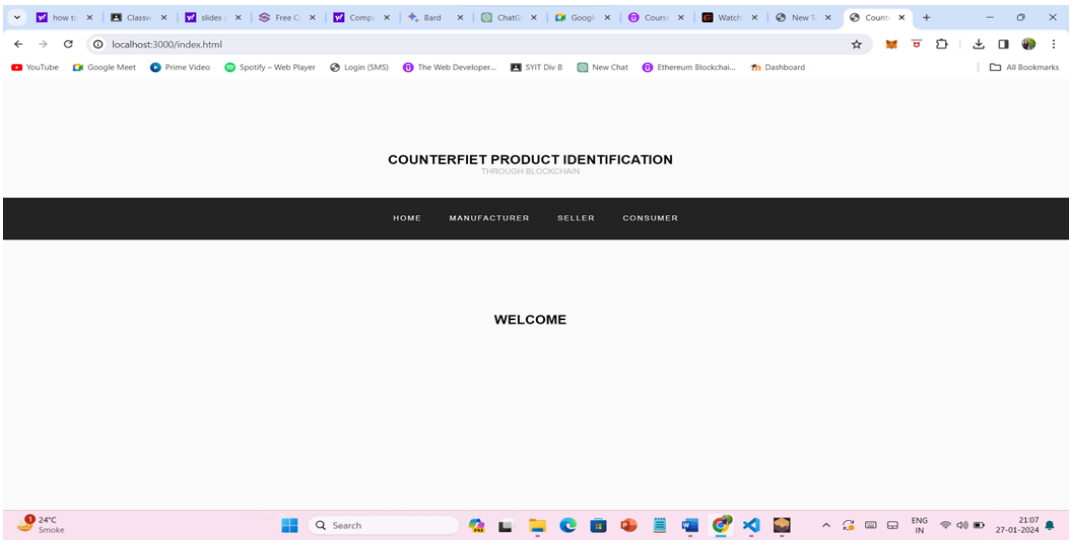

Fig 6.1

Product Registration: Once logged in, head to "Product Registration." Fill in your product details – name, description, etc. Click "Submit," and voilà, your product now has a unique ID on the blockchain. You'll get a confirmation once it's all set.

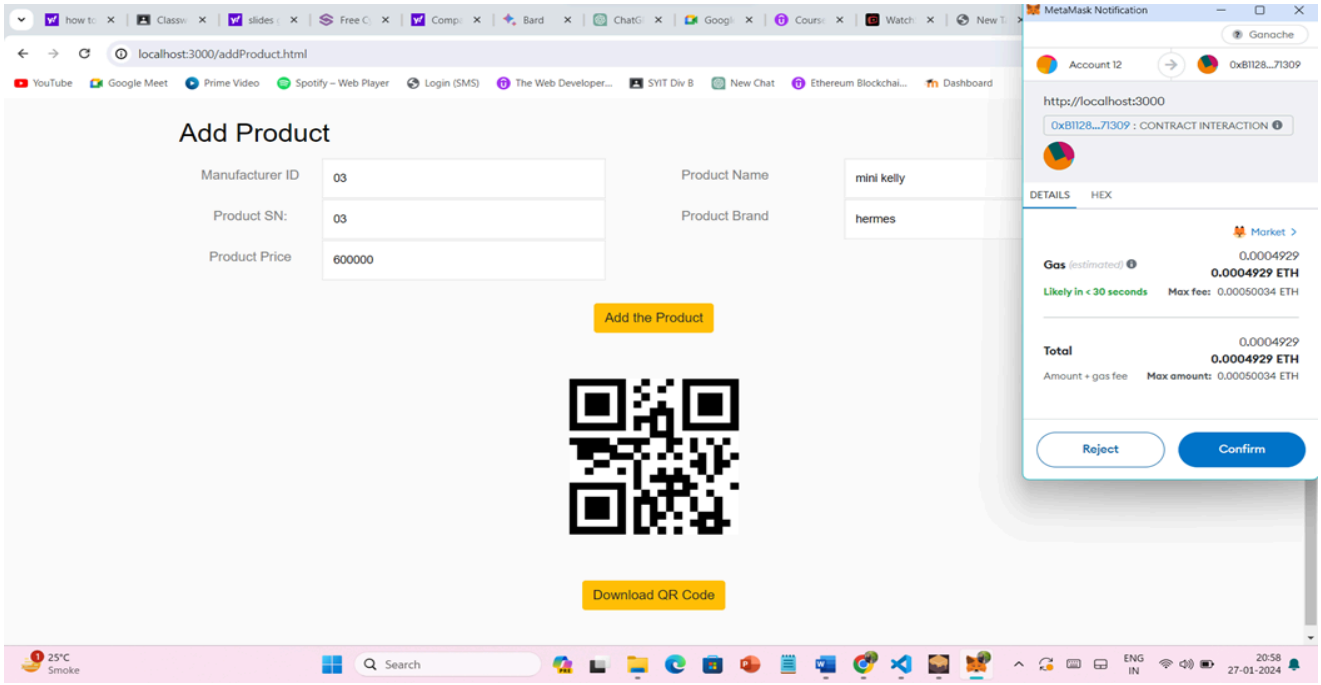

Fig 6.2

Seller Registration:

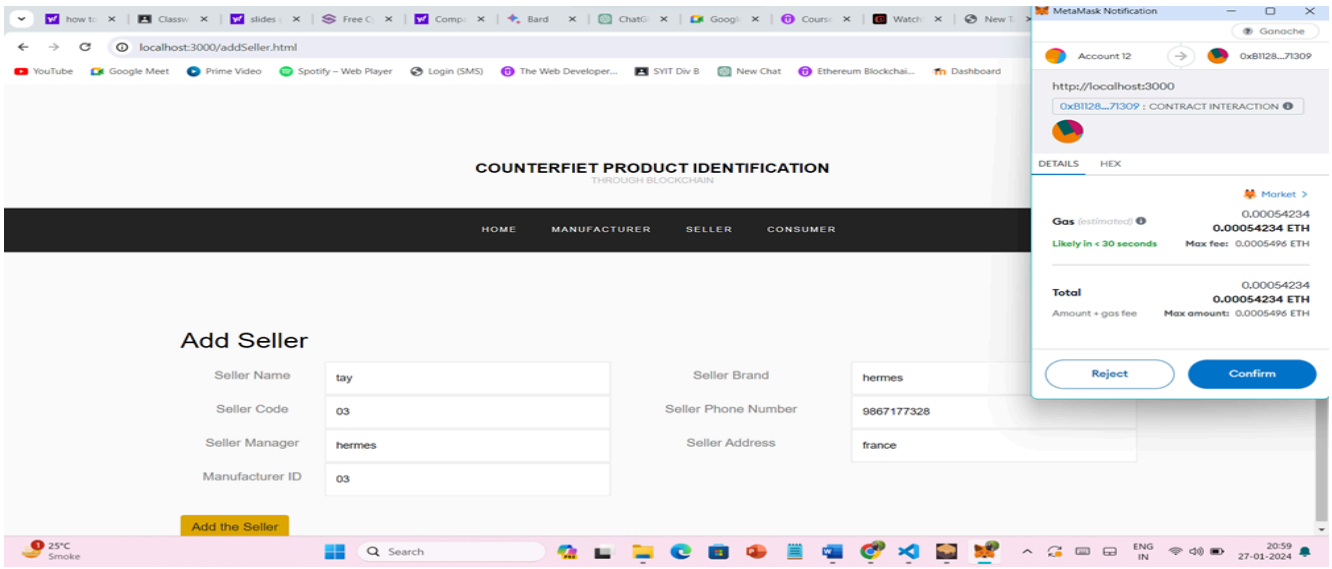

Fig 6.3

Sell Product to the seller:

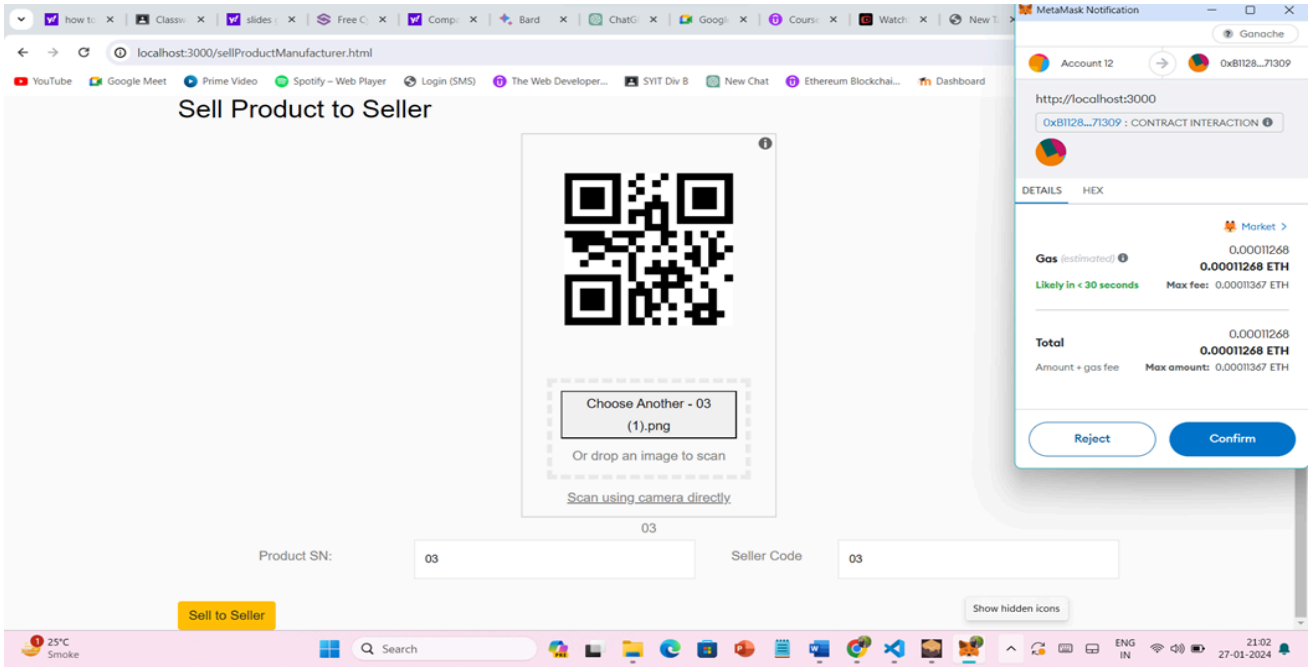

Fig 6.4

Query Seller:

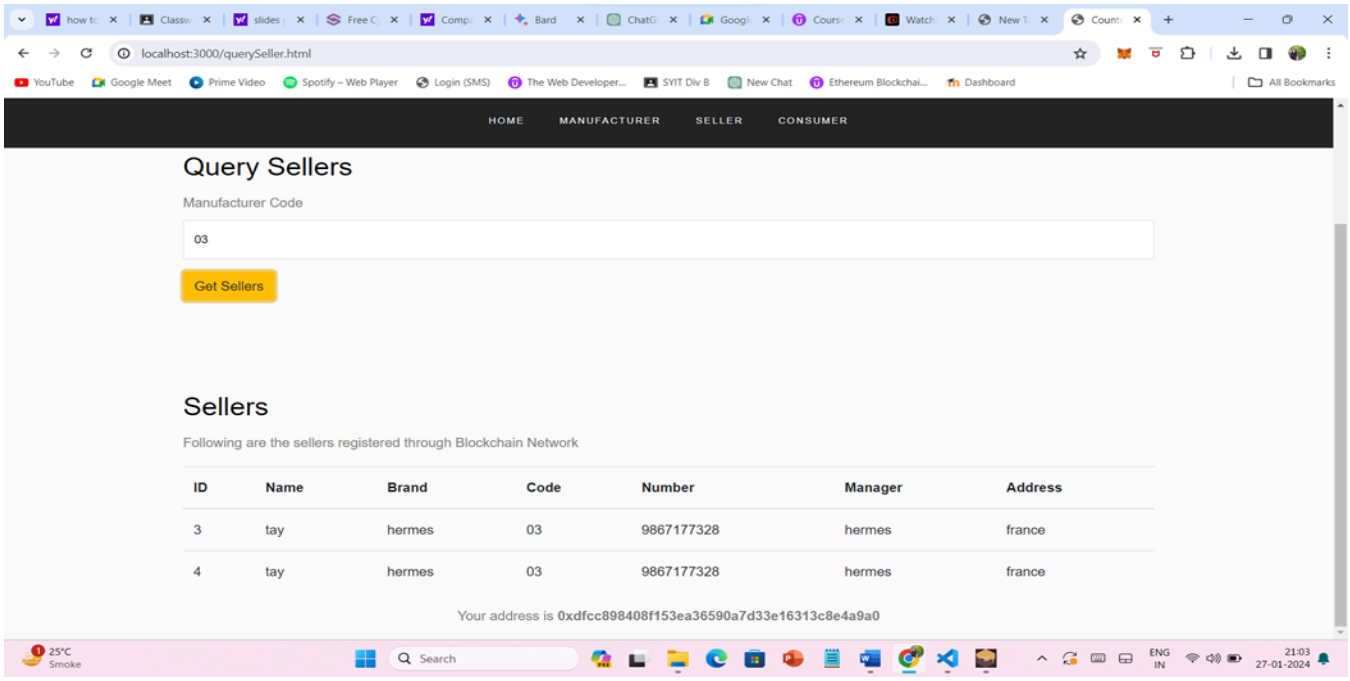

Fig 6.5

Sell Product to consumer:

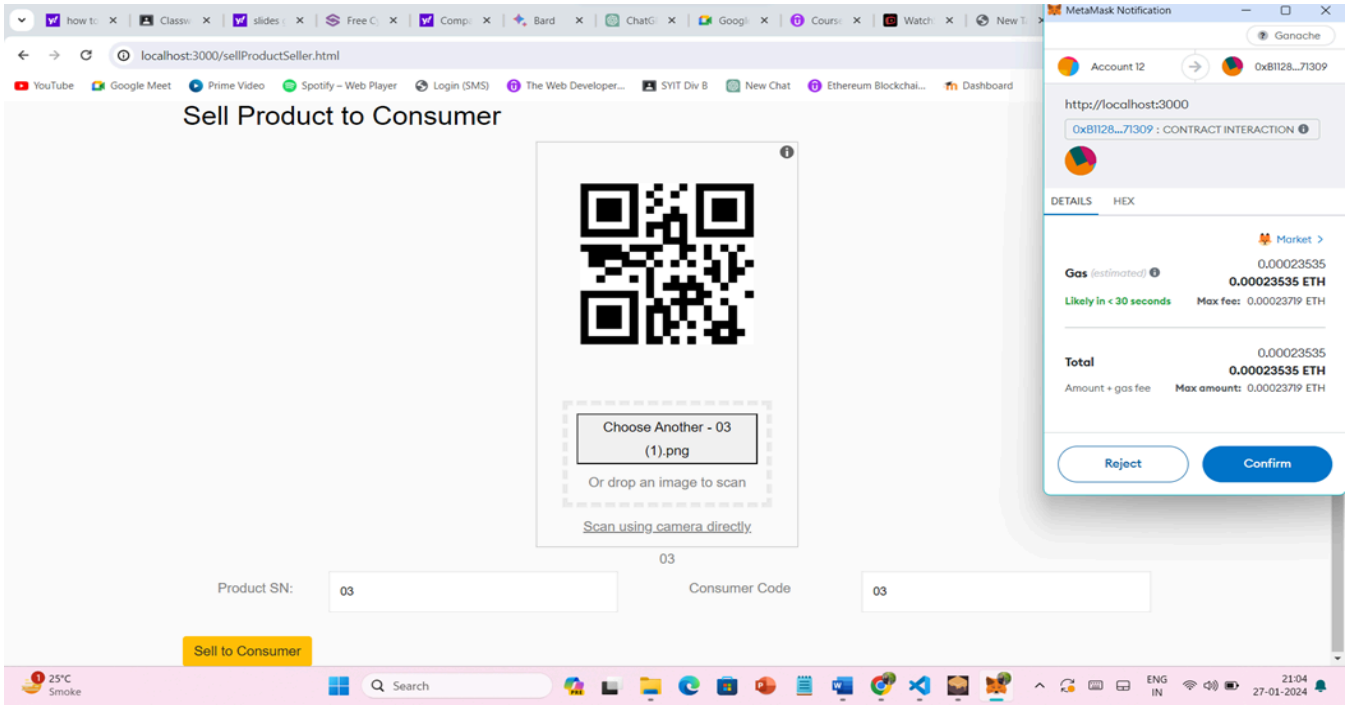

Fig 6.6

Products for sell:

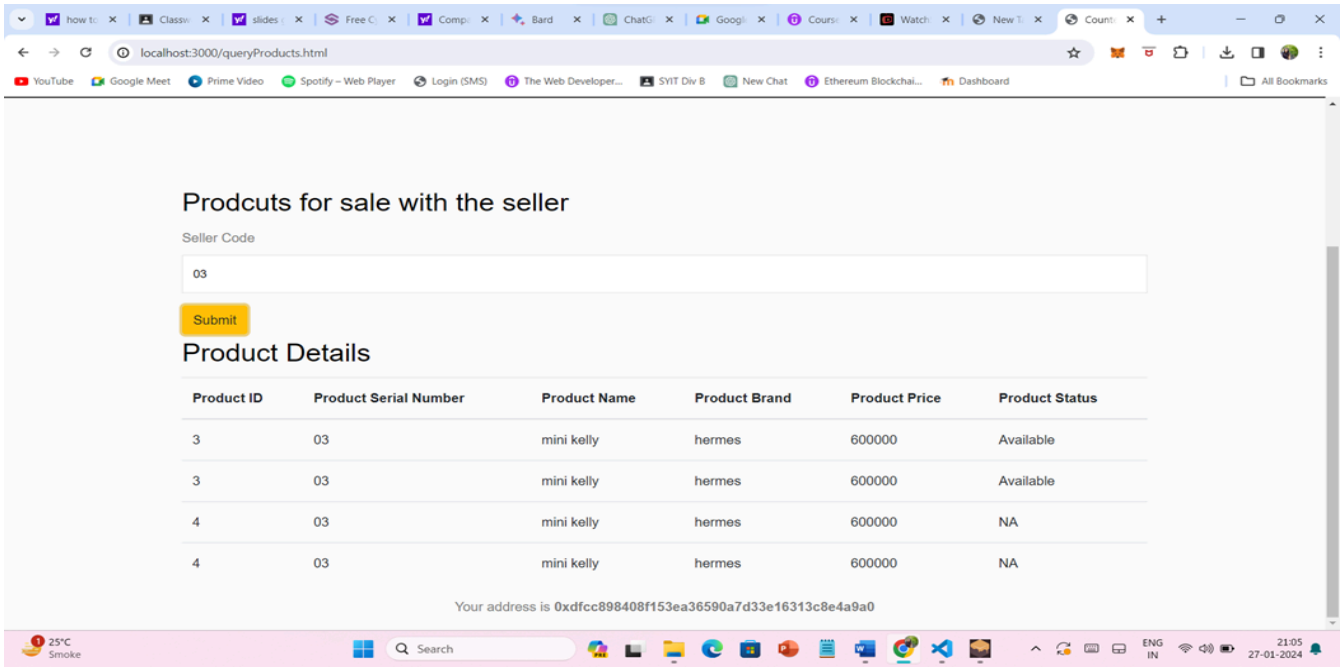

Fig 6.7

Authenticity Verification: Need to check if a product is legit? Visit "Product Verification," enter the product ID, and hit "Verify." The system checks the blockchain, and you get instant authenticity status.

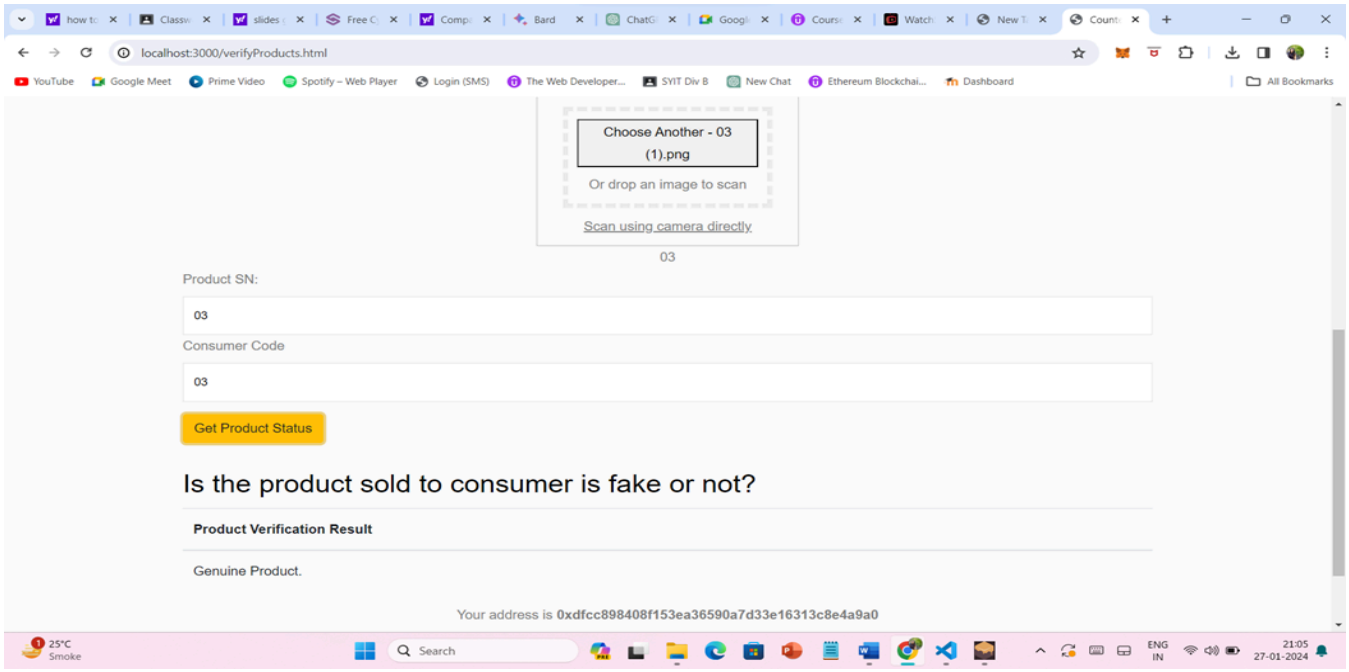

Fig 6.8

Consumer product History:

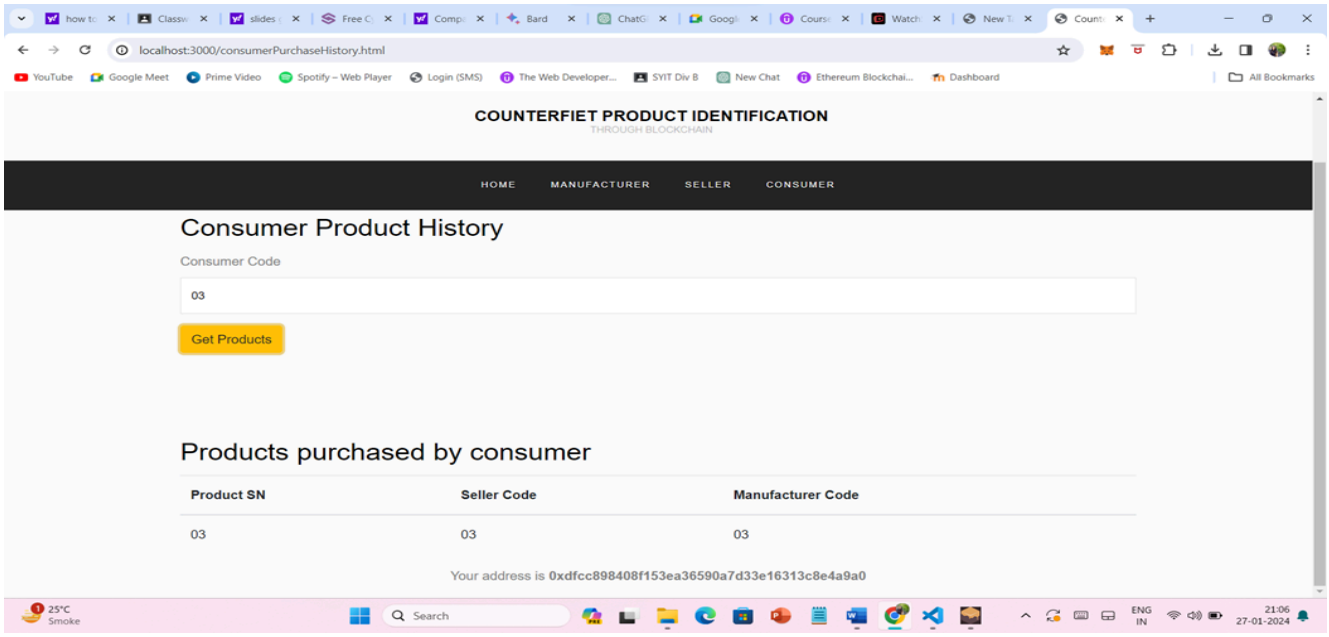

Fig 6.9

3. Cost Estimation

Cost estimation models are mathematical algorithms or parametric equations used to estimate the costs of a product or project. The results of the models are typically necessary to obtain approval to proceed, and are factored into business plans, budgets, and other financial planning and tracking mechanisms.

- **The Development Model**

  COCOMO (Constructive Cost Model) is a regression model based on LOC viz. number of Lines of Code. It is a procedural cost estimate model for software projects and often used as a process of reliably predicting the various parameters associated with making a project such as size, effort, cost, time and quality.

- **Key Parameter**
  a. Efforts - measured in person months units
  b. Schedule - measured in span of months or weeks

To estimate the effort and development time, COCOMO uses the same equations but with different coefficients (a, b, c, d in the effort and schedule equations) for each development mode. Types are as follows :

- Organic System
- Semi - detached System
- Embedded System

The basic COCOMO equations take the form

- Effort Applied (E) = ab (KLOC) bb [person-months]
- Development Time (D) = cb (Effort Applied) db [months]
- People Required (P) = Effort Applied / Development time [count]

  Where, KLOC is the estimated number of delivered lines (expressed in thousands) of code for a project.

  The coefficient ab, bb, cb and db are given in the following table:

Software Project	a_b	b_b	c_c	d_d
Organic	2.4	1.05	2.5	0.38
Semi-detached	3.0	1.12	2.5	0.35
Embedded	3.6	1.20	2.5	0.32

Table 6.2 Coefficient ab, bb, cb & db

COCOMO Model for "Counterfeit Product Identification using Blockchain"

1. Effort: ab*(KLOC)^bb PM(Person Month)
2. Time for development: cb * (Effort)^ db

Where,

Effort = Number of staff months (SM)

Size = Number of source lines of code

Time = Total number of months required to complete the project

The Project Code for Counterfeit product identification using Blockchain application contains 1000 Lines of code

Since, we know that 1000 Lines of Code = 1 KLOC (K - Kilo - 10^3)

Therefore, the project consists of 1 KLOC.

Effort = 2.4(1)^1.05 = 2.5 SM

Time for development = 2.5*(2.5)^0.38 = 4 Months

Cost per Month = Rs.30,000/-

Total Cost of the Project = Cost per Month * Time required for the development project

= 30000 * 00

= Rs. 1,20,000

CHAPTER 7 : CONCLUSIONS

1. Conclusion

The culmination of this project signifies a significant stride toward enhancing product authentication through blockchain technology. By implementing a smart contract on the Ethereum blockchain, the system allows manufacturers to register their products securely. The addition of features such as authenticity verification and the ability to mark a product as fake provides a robust mechanism for combating counterfeiting. The use of the ERC-721 token standard ensures each product is uniquely represented on the blockchain, creating a tamper-resistant record of product information.

The implementation leverages key technologies such as Truffle, and Ganache, enabling thorough testing in various blockchain settings. This comprehensive testing approach, combining manual and automated techniques, instills confidence in the reliability and security of the application. The incorporation of user acceptance testing, or beta testing, further validates the project's readiness for real-world usage. Through feedback from representative users, the project has been fine-tuned to meet the needs and expectations of its intended audience, ensuring a user-friendly experience.

In conclusion, this project stands as a testament to the potential of blockchain in revolutionizing product authentication. The systematic development process, rigorous testing methodologies, and user-centric approach contribute to the creation of a secure, efficient, and user-friendly system. As the project advances to deployment, it holds the promise of providing manufacturers and consumers alike with a trustworthy solution for authenticating products in an increasingly digital and interconnected world.

2. Limitations

- Limitation 1: Evolving Counterfeiting Tactics

  Description: As counterfeiters develop new methods, the system may require ongoing updates and adaptations to maintain its effectiveness in identifying and deterring such activities.
- Limitation 2: Required Cost to Store Data

  Description: The project required certain expenses for the blockchain data storage as When we anticipate the outcome, the expenditure is worthwhile because we will get immutable data.
- Limitation 3: Ethereum Token Requirement

Description: As Project is working on web 3 and web 3 is all about transactions & for making that transactions happen we required some amount of ethers in our MetaMask wallet

3. Future Scope of the Project

Future of the Project:

Looking ahead, the future of this project is filled with exciting possibilities. The system we've built for product authentication using blockchain lays a solid foundation, but there's ample room for growth. One avenue is the addition of advanced features to further enhance user experience and utility. Considering the dynamic nature of blockchain technology, future updates may include integration with emerging technologies or the exploration of alternative blockchain networks, opening up new dimensions for product authentication.

Unexplored Aspects for Future Implementation:

While the current project addresses key aspects of product authentication, certain features and functionalities could be explored at a later date. For instance, the incorporation of artificial intelligence for predictive analytics or machine learning algorithms for anomaly detection could add an extra layer of sophistication to the authentication process. Additionally, the system could be extended to support a wider range of product types and industries beyond its initial scope.

Extension through Research and Funding:

With dedicated research and additional funding, the project's scope could be significantly extended. This could involve delving deeper into blockchain scalability solutions to accommodate a larger user base, exploring interoperability with other blockchain networks, and conducting extensive user studies to refine and tailor the system based on diverse user needs. Increased research and funding would also facilitate collaborations with industry partners, further validating the system's effectiveness across different sectors.

Modifiable Factors for Enhancement:

Several factors within the project are ripe for modification and enhancement through future research. The smart contract logic could be refined to incorporate more sophisticated verification mechanisms.

Integration with decentralized identity solutions could strengthen user authentication processes. Moreover, exploring environmentally friendly blockchain options aligns with the growing emphasis on sustainability. By fostering collaborations and keeping abreast of technological advancements, the project can evolve to meet the ever-changing landscape of blockchain and product authentication.

REFERENCES

[1] E. Daoud, D. Vu, H. Nguyen, M. Gaedke, Improving Fake Product Detection Using Ai-Based Technology, in 18th International Conference e-Society (2020)

[2] S. Chen, R. Shi, Z. Ren, J. Yan, Y. Shi, J. Zhang, A blockchain-based supply chain quality management framework, in 2017 IEEE 14th International Conference on e-Business Engineering (ICEBE) (IEEE, 2017), pp. 172–176

[3] K. Toyoda, P.T. Mathiopoulos, I. Sasase, T. Ohtsuki, IEEE access 5, 17465 (2017)