

# **PRODUCT IDENTIFICATION SYTEM USING BLOCKCHAIN**

Submitted in partial fulfillment of the requirements for the award of  
Bachelor of Engineering degree in Computer Science and Engineering

By

**ROHITH.M (Reg. No - 39110858)**  
**SRIKANTH.B(Reg. No - 39110966)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**  
**SCHOOL OF COMPUTING**

## **SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY**  
**(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,**  
**CHENNAI - 600119**

**APRIL - 2023**



# **SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY  
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

[www.sathyabama.ac.in](http://www.sathyabama.ac.in)

## **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

### **BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the bonafide work of **Rohith.M(39110858)** and **Srikanth.B(39110966)** who carried out the Project Phase-2 entitled "**PRODUCT IDENTIFICATION SYTEM USING BLOCKCHAIN**" under my supervision from January 2023 to April 2023.

**Internal Guide**

**MS GRACELIN SHEENA . B**

**Head of the Department**

**Dr. L. LAKSHMANAN, M.E., Ph.D.,**



Submitted for Viva voce Examination held on 27.4.2023

**Internal Examiner**

**External Examiner**

## **DECLARATION**

I, **ROHITH.M(Reg.No-39110858)**, hereby declare that the Project Phase-2 Report entitled “**PRODUCT IDENTIFICATION SYTEM USING BLOCKCHAIN**” done by me under the guidance of **MS GRACELIN SHEENA.B.** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

**DATE:27.04.2023**

**PLACE: Chennai**

A small, square image showing a handwritten signature in blue ink on a white background. The signature appears to be 'Rohith'.

**SIGNATURE OF THE CANDIDATE**

## **ACKNOWLEDGEMENT**

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D, Dean**, School of Computing, **Dr. L. Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **MS.GRACELIN SHEEN.B** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-2 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

## **ABSTRACT**

Blockchain usually consists of collection of records called “Blocks”, These blocks are cryptographically linked to each other by a chain. Each block consists of current hash of the data, Cryptographic hash of previous block and timestamp. It follows the technique of back linking blocks and its representation can be shown using Merkle Tree. In recent years, block chain has received increasing attention and numerous applications have emerged from this technology. In this idea we are using the Decentralized Blockchain technology approach to ensure products are genuine. we describe a decentralized blockchain system with products anti-counterfeiting, in that way manufacturers can use this system to provide genuine products without having to manage direct-operated stores, which can significantly reduce the cost of product quality assurance. Therefore the contents of its data are tamper-proof before it is used in blockchain. The customer thinks a thousand times before purchasing the product. He has questions in his mind whether the product that he is purchasing is fake or not. So in order to stop the making of fake products ,

It is important to bring transparency about the products to the notice of the consumers. The developing presence of fake and risky products in the world is a purpose for the situation and blockchain technology has taken the following step toward its entire demolition. This technique is used to watch the supply chain in the market. So In this paper , we create a system that delivers the original product to the customer and helps to identify if the product is fake or real to the customer. This proposed system not only stops the making of fake products but also spreads awareness about this. For the creation and packaging of the right items each of these items needs to be provided with a digital code of its own. This research paper proposes a form of identification of fake products using blockchain technology. Discusses the software implementation process where the product code is scanned using this application and confirms whether the given product is fake or not.

**Keywords: Blockchain, Fake Product, Supply Chain**

## TABLE OF CONTENTS

Chapter No.	TITLE	Page No.
	<b>ABSTRACT</b>	v
	<b>LIST OF FIGURES</b>	vii
1	<b>INTRODUCTION</b>	1
2	<b>LITERATURE SURVEY</b>	4
	2.1 Inferences from Literature Survey	6
	2.2 Open problems in Existing System	7
3	<b>REQUIREMENTS ANALYSIS</b>	9
	3.1 Feasibility Studies/Risk Analysis of the Project	9
	3.2 Software Requirements Specification Document	11
	3.3 System Use Case	13
4	<b>DESCRIPTION OF PROPOSED SYSTEM</b>	15
	4.1 Selected Methodology or process model	15
	4.2 Architecture / Overall Design of Proposed System	22
	4.3 Financial Report	23
	4.4 Description of Software for Implementation and Testing plan of the Proposed Model/System	23
	4.4 Project Management Plan	34
5	<b>IMPLEMENTATION DETAILS</b>	37
	5.1 Development and Deployment setup	36
	5.2 Algorithms Used	38
6	<b>RESULTS AND DISCUSSION</b>	44
7	<b>CONCLUSION</b>	45
	7.1 Conclusion	45
	7.2 Future Work	46
	7.3 Research Issues	46
	7.4 Implementation Issues	46
	<b>REFERENCES</b>	47
	<b>APPENDIX</b>	47
	<b>A. SOURCE CODE</b>	47
	<b>B. SCREENSHOTS</b>	62
	<b>C. RESEARCH PAPER</b>	67

## LIST OF FIGURES

<b>Figure No.</b>	<b>FIGURE NAME</b>	<b>Page No.</b>
3.2	MetaMask UI	12
4.2	System Architecture	23
4.5.1	Upload Product Details	35
4.5.2	Generate QR Code	36
4.5.3	Add User	36





# **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 maintain the database. In a distributed ledger system, a block will not be recognized as authentic until it has been validated and sanctioned by a number of distinct entities. Additionally, the data that is included within blocks is immutable, meaning that it cannot be changed in any way. For instance, the utilization of a blockchain to facilitate the execution of smart contracts results in a system that is more reliable due to the fact that it eradicates any concerns regarding the precision of the information.

The blockchain technology makes available a database that is both decentralized and incorruptible, and it has the potential to be used for a wide variety of applications. It is a decentralized network that has the potential to have millions of members located in different regions all over the world. Users of the peer-to-peer (P2P) network have the ability to generate records that cannot be altered and to engage in financial transactions with one another. This technology is also known as the write-once, append-many technology, and the only way it can be updated is once it has obtained the agreement of all of the members that are part of that network. This is the only way it can be updated. Every user has the ability to add new information to the Blockchain, and all of the data that is saved on the Blockchain is encrypted to protect it from these

generated after it as a result of the usage of a cryptographic hashing algorithm. When a block is completed, it generates a unique hash code that is then connected to the block that follows after it, therefore establishing a chain of blocks. This process continues until the chain is complete. Following the completion of the validation process for a piece of data, the data is then compiled into a block, which must then be added to the chain. The information that it conveys is there permanently on the network, which means that it cannot be modified or wiped in any way. The information that it carries cannot be changed or erased in any way. In order to make it easier for users to share and exchange data with one another, a public blockchain will feature ledgers that are managed in an unattended fashion. A private or permissioned ledger that can be utilized by the firm to centrally administer their own transactions and that can be utilized either within the company or between companies. There is no way to get around the security measures. It is necessary for the transaction to receive consensus from all of the nodes in the network before it can be regarded valid. If someone is able to break into one computer system, then it is imperative that they break into the rest of the computer systems as well. As a direct consequence of the rapid progress that has been made in information technology over the course of the most recent few years, the security of one's data is currently of greater significance than it has ever been. The manipulation of data has become a key worry for national security in the modern world. a substantial number of events involving falsified certificates have occurred in recent years. Some of these incidents occurred in recent years. Within the scope of this research project, a Blockchain-based certification system was developed with the intention of resolving the issue in question. The Blockchain is an ideal system for the storage of data, regardless of whether the data is meant to be static, such as in the case of a registry, or dynamic, such as in the cases of transactions, registration, and distribution. This is because the Blockchain can store both types of data with equal ease. Because these data are recorded in an array sequence to one another, modifying one data set will require modifying all of the history records on every single node, which will result in the data set becoming immutable. Because these data are recorded in an array sequence to one another, modifying one data set will require modifying all of the

Our team came up with the idea for a decentralized application as well as a blockchain-based Product identification system, both of which have been put into

action. The selection of this technology was based on its properties of being indestructible, encrypted, and trackable in addition to having the ability to synchronize data. The implementation of the fundamental characteristics of Blockchain technology into the solution results in an increase in the overall efficiency of the processes at each level.

Satoshi Nakamoto is credited as being the first person to publicly propose the concept of a blockchain in the year 2008. The blockchain is a distributed online ledger that makes it possible to share data in a manner that is both decentralized and open to public scrutiny. Before being added to the various blocks of a distributed recording system, each and every piece of transaction data that is saved in the nodes of that system undergoes compression. As a result of the fact that data of various types have been partitioned off into their own blocks, it is possible to conduct verifications without the participation of any intermediaries. After that, each of the nodes contributes a timestamp to the blockchain that is being created by the others. The data that is stored in each block can be validated concurrently once it has been entered, after which it will no longer be possible to alter it in any way. The public can view the entire process in its entirety, and the procedure's security and reliability will not be compromised in the process. The introduction of Ethereum Smart Contracts in 2013 gave a huge push to the development of blockchain technology, which eventually evolved into blockchain 2.0. This was due to the fact that Ethereum Smart Contracts provided a platform for smart contracts. As can be seen in Figure 1, the necessity to address difficulties connected to decentralized payments and currency was the key impetus behind Bitcoin's adoption of blockchain 1.0. This requirement was met by the development of Bitcoin. Blockchain 2.0 was developed with the intention of decentralizing the whole market; it does this through the utilization of smart contracts. This contributes to the creation of value by enabling the development of blockchains that are complementary to Bitcoins.

## **CHAPTER 2**

## LITERATURE SURVEY

[1] One of them demonstrated a solution for identifying potentially counterfeit products using an Android app that does a search for the goods in question over the Blockchain network.

[2] Prabhu Shankar and R. Jayavadivel. A Review of Methods for Identifying Fake Goods. The vast quantities available online and on the illicit market are fueling a meteoric rise in counterfeit goods. So, Dealing with the challenges of recognising counterfeit goods and developing the proper technology to enhance detection are urgently needed. accuracy. One of the current active study areas being studied is this.. In this article, we'll look at a few different methods for spotting fakes.

[3] Marko Steven's Smart Tags: Counterfeiting Prevention and Brand Security in the wine business. The authors of this study provide a brand protection and anti-counterfeiting system designed exclusively for the wine industry using smart tags and Cloud-enabled technology. RRCs and functional inks hosted on the cloud are the core idea of smart tags, enabling two-way communication between the vineyard and the end user.

[4] Si Chen and Rui Shi's Supply Chain Quality Management in a Blockchain Environment. Here, we suggest an architecture built on the blockchain. Intelligent supply chain quality control using blockchain technology will have a theoretical foundation thanks to the architecture. It also serves as a springboard for research into information resource management in decentralized, online businesses.

[5] Another article presented a blockchain-based method for detecting counterfeit goods by utilizing the SHA-256 algorithm.

[6] Researchers have developed a fully operational anti-product counterfeiting system that relies on digital signature verification.

[7] A Product Ownership Management System built on the blockchain was suggested in one of the publications. It demonstrated the advantages of the blockchain technology over the more conventional RFID system.

[8] In another publication, researchers introduced a blockchain-based Internet of Things food tracking system. Fuzzy logic was employed to make judgments about the quality of the meal in this model.

[9] A study presented a solution that used RFID and blockchain to do away with restrictions in the last stage of the supply chain.

[10] One paper used blockchain and the Internet of Things to improve upon the standard supply chain method

## **2.1 INFERENCES FROM LITERATURE SURVEY**

1. Blockchain technology can offer a secure and decentralized approach to product identification eliminating the need for intermediaries and manual verification processes.
2. Smart contracts, public key infrastructure, and consensus mechanisms are commonly used in blockchain-based product identification systems to ensure the integrity and authenticity of the products.
3. Hybrid blockchain structures, combining public and permissioned blockchains, can offer a balance between security and scalability in product identification systems.
4. While blockchain-based product identification shows promise, there are still challenges that need to be addressed, such as scalability, interoperability, and legal and regulatory issues.
5. A systematic review of the existing literature can provide insights into the different approaches and architectures proposed for certificate verification using blockchain technology, highlighting the benefits and challenges of each approach.

From the above-mentioned literature works, it is clear that the research work in the blockchain field to hire people is new and relatively few. Blockchain technology can be used in all contexts where a decentralized system is necessary in order to ensure the involvement of many people in the same network and guarantee full transparency and reliability between people who do not know each other. Therefore, blockchain technology is not only useful for creating digital currencies or new financial technologies but can be applied for a wide variety of applications, such as protection systems of digital identity, the provenance of documents, organizational data management, and digital and physical assets.

## 2.2 OPEN PROBLEMS IN EXISTING SYSTEM

While product identification using blockchain technology offers many benefits, there are still some open problems in the existing system that need to be addressed. Here are some of the challenges that researchers and practitioners are currently working on.

**Scalability:** As the number of product being verified on the blockchain network grows, the system needs to be able to handle the increased volume of transactions. Currently, blockchain technology has limited scalability, which may result in slower verification times and higher transaction fees.

**Interoperability:** There are several different blockchain networks currently in use, and they are not always compatible with each other. This can make it difficult to transfer products between different blockchain networks or to verify products across multiple networks.

**Privacy:** While blockchain technology is inherently secure and transparent, there are concerns around privacy when it comes to product verification. For example, some individuals may not want their product information to be publicly available on the blockchain.

**Regulation:** There is currently a lack of regulatory framework around blockchain-based product identification. This may create legal and regulatory challenges, particularly in industries where certifications and licenses are required.

**Adoption:** While blockchain-based product identification has the potential to revolutionize the way we verify credentials, there is still a lack of widespread adoption. This may be due to a lack of awareness or understanding of the technology, or resistance to change from traditional verification methods.

**Governance:** There is a need for a governance model for blockchain-based product verification systems. The model should address issues such as decision-making,

dispute resolution, and updating of the system.

**Implementation complexity:** Implementing a blockchain-based product identification system can be complex and require significant technical expertise. It may be difficult for smaller organizations or brands to develop and maintain such a system.

**Adoption barriers:** Adoption of blockchain-based product identification may be slow due to resistance from traditional brands or a lack of awareness among potential users.

**High transaction fees:** Blockchain transactions can be expensive, and the cost of product identification on the blockchain may be prohibitive for some organizations.

**Secrecy concerns:** While blockchain technology provides a high level of transparency and immutability, it can also be problematic for privacy. Personal information stored on the blockchain may be visible to all participants, potentially compromising confidentiality.

**Legal recognition:** The legal recognition of product identification on the blockchain may still be uncertain or underdeveloped in some jurisdictions, making it difficult for them to be widely accepted.

In summary, while product identification using blockchain technology is a promising solution for secure and efficient verification of credentials, there are still several open problems that need to be addressed. These include scalability, interoperability, privacy, regulation, adoption, and governance. Addressing these challenges will be crucial for the widespread adoption of blockchain-based product verification systems. While blockchain technology offers significant potential for improving the security and efficiency of product verification, these potential problems must be carefully considered and addressed in order to ensure the successful adoption and implementation of such systems.



## **CHAPTER 3**

### **REQUIREMENT ANALYSIS**

#### **3.1 FEASIBILITY STUDIES/RISK ANALYSIS OF THE PROJECT**

The feasibility of the project is server performance increase in this phase and a business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis, the feasibility study of the proposed system is to be carried out. A feasibility study and risk analysis are essential steps in evaluating the viability and potential risks of implementing a product identification system using blockchain technology.

##### **ECONOMICAL FEASIBILITY**

The economic feasibility of a blockchain-based product identification system involves analyzing the costs associated with developing and maintaining the system. This includes the cost of hardware, software, development, maintenance, and operation. It is also important to evaluate the potential return on investment and determine whether the benefits outweigh the costs.

##### **TECHNICAL FEASIBILITY**

The technical feasibility of a blockchain-based product identification system involves assessing whether the necessary technical resources and expertise are available to develop and maintain such a system. This includes evaluating the compatibility of existing systems with blockchain technology and determining whether there are any technical limitations that need to be addressed.

##### **OPERATIONAL FEASIBILITY**

The operational feasibility of a blockchain-based product identification system involves assessing whether the system can be integrated into the existing processes and workflows of the organization. It also involves evaluating the usability and user-friendliness of the system and determining whether it can be easily adopted by end-users.

## **TECHNICAL RISKS**

The technical risks associated with a blockchain-based product identification system include potential security vulnerabilities, data loss, and system downtime. It is important to evaluate the security features of the blockchain technology being used and ensure that the system is regularly updated and maintained.

## **LEGAL RISKS**

The legal risks associated with a blockchain-based product identification system include uncertainties regarding legal recognition and compliance with data protection laws. It is important to consult legal experts and ensure that the system complies with relevant laws and regulations.

## **OPERATIONAL RISKS**

The operational risks associated with a blockchain-based product identification system include user adoption and usability issues. It is important to involve stakeholders and end-users in the development process and ensure that the system is user-friendly and easy to use.

## **ECONOMIC RISKS**

The economic risks associated with a blockchain-based product identification system include the cost of implementation and the potential for the system to become obsolete in the future. It is important to evaluate the potential return on investment and ensure that the system is scalable and adaptable to future changes.

## 3.2 SYSTEM SPECIFICATION

### Hardware Environment

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the systems do and not how it should be implemented.

- Hard disk : 512 GB
- RAM : 4 GB
- PROCESSOR : CORE i3/ i5

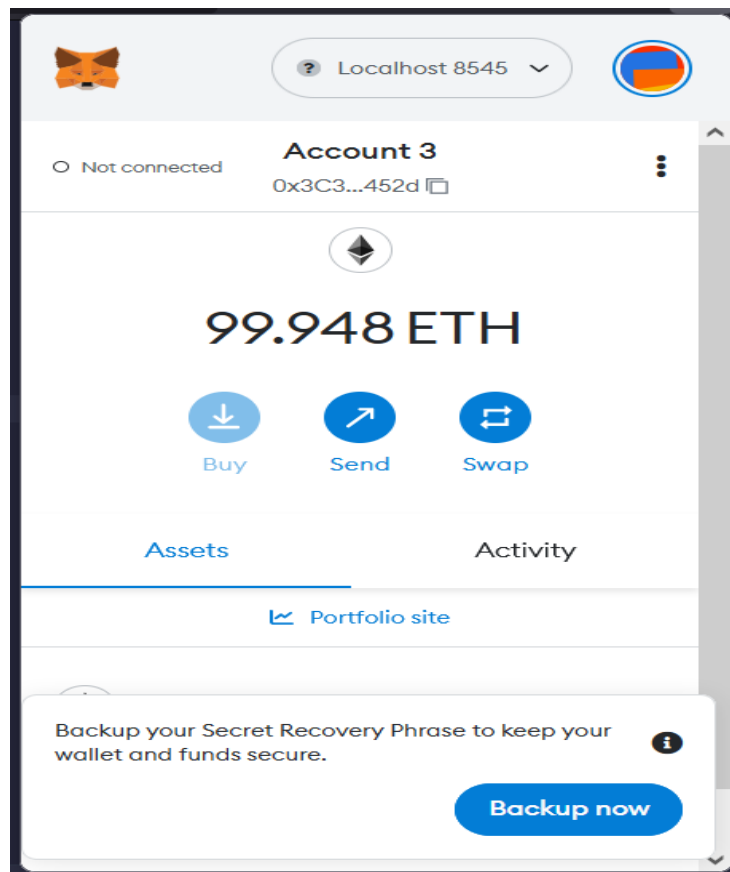
### Software Environment

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team's and tracking the team's progress throughout the development activity.

- Operating system : Windows 10/11
- Languages : Python, Solidity, HTML, CSS.
- Data Base : JSON
- Tools : Ganache, Solidity, Ethereum
- Wallet : MetaMask
- IDE : REMIX IDE

### MetaMask:

Metamask is a popular cryptocurrency wallet that allows users to store, send, and receive digital assets on the Ethereum blockchain. It is a browser extension that can be used with popular web browsers like Google Chrome, Mozilla Firefox, and Brave.



**Fig 3.2: MetaMask UI**

### 3.3 SYSTEM USE CASE

A system use case for product identification using Blockchain Technology could involve the following steps:

**Manufacturing of Products:** The manufacturer delivers a product to the buyer, which contains details about the product, such as the name of the product, the date of manufacturing, and the expiry date.

**Blockchain Integration:** The product is then integrated into a blockchain network, which creates a tamper-proof and secure record of the product. This ensures that the product cannot be duplicated or falsified in any way.

**Verification Process:** When a third party, such as an buyer or an retailer, needs to verify the authenticity of the product, they can access the blockchain network and retrieve the product's details. The verification process can be automated, making it quick and efficient.

**Decentralization:** Since the blockchain network is decentralized, no central authority has control over the manufacturing process, and the products can be verified by anyone with access to the blockchain network.

**Cost-effective:** The use of blockchain technology for product identification can also reduce costs associated with manual verification processes.

**Product details on Blockchain:** The product details are stored on the blockchain in a tamper-proof manner. Each certificate is uniquely identified by a hash value, which ensures that the product cannot be counterfiet or duplicated.

**Verification Request:** A third party requests verification of a product. The request is sent to the blockchain network.

**Verification Result:** The verification result is sent back to the requesting party, indicating whether the product is authentic or not.

**Product Verification:** The blockchain network validates the request and verifies the authenticity of the product. The verification process involves checking the hash value of the product details stored on the blockchain against the hash value of the product provided in the verification request. If the hash values match, the product is deemed authentic.

**Revocation of product:** If a product is found to be fraudulent or counterfeit, the delete option can be used to remove the product from the blockchain. This ensures that the product cannot be used in the future for verification purposes.

**Consent withdrawal:** In some cases, individuals may have given their consent to have their product data stored on the blockchain but later change their mind. In such cases, the delete option can be used to remove their product data from the blockchain.

**User Management:** The admin tab can be used by authorized administrators to manage users and their access privileges. They can add new users, delete existing ones, and modify user roles and permissions.

**Audit Trail Management:** Blockchain technology provides an immutable and transparent audit trail that can be used to verify the authenticity of products. The admin tab can be used to manage the audit trail by ensuring that all product-related transactions are recorded and can be traced back to their source.

**Efficient Manufacturing and Management of Products:** The product details storage tab can streamline the manufacturing and management of products. Using the blockchain, products can be manufactured and recorded in real-time, reducing the administrative burden of product manufacturing and management.

## **CHAPTER 4**

### **DESCRIPTION OF PROPOSED SYSTEM**

#### **4.1 SELECTED METHODOLOGY OR PROCESS MODEL**

System is maintaining Status of product i.e., Manufacturer of product, current owner of product, and history of owners, time stamp i.e., at what time product was updated and a QR code.

Stage 1: Product Enrollment Process:

Initially manufacturer will be the first owner of product. So, manufacturer will request administrator to add product on the network, at that time QR code will be generated. Administrator will enroll product and manufacturer on the network, and QR code is taken.

Stage 2: Ship Product to Distributor:

In the next step manufacturer will ship the product to distributor. When distributor receives product will scan the QR code and update his details on the network, about product ownership, time Stamp and date.

Stage 3: Ship Product to Retailer:

At this Stage, the retailer receives product from Distributor and scan QR code assigned to product using QR code scanner, and will update owner details of the product on the network.

Stage 4: End User Authentication Process:

At the end of the chain, customer will take the product, go to website and upload QR code over there, and customer will be able to get all detail about product from manufacture to last retailer. And after getting details, it is his question where to buy the product or not.

#### **WORKING PROCESS**

Blockchain is a decentralized distributed database. In this research to design and

develop a system for dynamic and secure supplychain system using smart contract in blockchain environment. In this work we also illustrate own blockchain in open-source environment with custom mining strategy as well as smart contract. Finally validate and explore system performance using consensus algorithm for proof of validation. Dynamic QR-code and unique product identification for each product's detail in proposed system.

- The working processes of the system developed in this study are as follows  
Brand manufactures a product and enter the product's data into the system.
- Next, the system automatically records the serial number of the product in a blockchain.
- Each buyer also receives an inquiry number and electronic file of their product.
- When buying for a product online, a manufacturer simply sends the serial number or product with a QR code to the buyer. The companies send inquiries to the system and are informed if the serial numbers are validated.
- The QR code enables them to recognize if the product has been duplicated or forged.

### **Module 1: Uploading a Product details**

The uploading page provides the user with an option to upload the Product details. Once user presses the upload button the data will be pushed on to the webpage and the MetaMask Wallet. Once the MetaMask Wallet receives the request from uploading page it generates a unique hash and a QR code for the product & the data is pushed into the Ethereum blockchain. We are using "Ganache" to perform transaction because original ether cost 2.4lakh INR. When the data is pushed the transaction log will be automatically pushed into the Ethereum testnet.

### **Module 2: Verification and Validation**



The verification page takes one parameter i.e., unique hash. Whenever the user wants to validate the product, he uploads the product details along with the generated hash address. The GET request goes to Verify.js and a JSON is returned. We process the JSON and will check whether the hash is present in the transaction log or not. If found in the transaction log FOUND is returned else NOT FOUND is returned.

### **ALGORITHM:**

- Product Verifier: It allows the user to verify the Product.
- Procedure: CONTRACT Product verifier.

### **FUNCTIONS:**

- **Modifiers**

Modifiers are like check constraints. for validating inputs. The modifiers in this algorithm checks whether the product is not expired.

### **4.1.1 Libraries and Frameworks Used**

The right blockchain framework to run our project is one of the most basic and important things to think about.

#### **1) VISUAL STUDIO CODE:**

Visual Studio Code is a streamlined code editor with support for development operations like debugging, task running, and version control. It aims to provide just the tools a developer needs for a quick code-build-debug cycle and leaves more complex workflows to fuller featured IDEs, such as Visual Studio IDE. The code of the editor is completely Open Source, and there's no payment required to use it. It uses Electron as its base, which enables it to be cross platform and work on Mac, Windows and Linux. It's built using Node.js, and you can extend it using JavaScript (which makes it a win for all us JavaScript developers). Visual Studio Code provides a range of features that make it popular among developers, including syntax highlighting, code completion, debugging tools, and Git integration. It also offers a marketplace of extensions that can be used to customize and enhance the editor's functionality. One notable feature of Visual Studio Code is its IntelliSense feature, which provides code completion, parameter suggestions, and other intelligent features based on the programming language being used. It also supports debugging of various programming languages through the use of debuggers and breakpoints.

In addition, Visual Studio Code has a built-in terminal, allowing developers to execute commands and scripts directly within the editor. It also supports version control systems like Git, providing developers with the ability to manage code changes and collaborate with other developers.

#### **2) REACT JS:**

React is a library for building composable user interfaces. It encourages the creation of

reusable UI components, which present data that changes over time. Lots of people use React as the V in MVC. React abstracts away the DOM from you, offering a simpler programming model and better performance. React can also render on the server using Node, and it can power native apps using React Native. React implements one-way reactive data flow, which reduces the boilerplate and is easier to reason about than traditional data binding. React JS is an open-source JavaScript library developed by Facebook for building user interfaces. It allows developers to create reusable UI components and manage the state of the application efficiently. React uses a declarative approach to building UIs, where developers describe the desired outcome of a component and let React handle the rendering. One of the key benefits of React is its virtual DOM, which allows for efficient updates and rendering of the UI. Instead of updating the entire UI, React only updates the components that have changed, resulting in faster and more efficient performance.

React also provides a component-based architecture, which makes it easier to break down complex UIs into smaller, reusable components. This allows developers to build UIs that are easy to maintain, test, and debug.

### **3) ETHEREUM:**

Ethereum is a blockchain platform with its own cryptocurrency, called Ether (ETH) or Ethereum, and its own programming language, called Solidity. Ethereum is an open-source, decentralized blockchain platform that allows developers to build decentralized applications (dapps). It was first proposed in 2013 by Vitalik Buterin, and the Ethereum network went live in 2015. Ethereum uses smart contracts, which are self-executing programs that automatically enforce the rules and conditions of a contract. These smart contracts run on the Ethereum Virtual Machine (EVM), which is a runtime environment for executing smart contracts. The EVM ensures that smart contracts are executed in a secure and deterministic manner. Ethereum also has its own cryptocurrency, called Ether (ETH), which is used to pay transaction fees and incentivize miners to secure the network. Ether can also be used as a currency for buying and selling goods and services. One of the key benefits of Ethereum is its ability to create and execute decentralized applications. This means that applications built on Ethereum are not

controlled by any central authority or single entity. Instead, they are powered by a network of nodes that work together to maintain the integrity of the network. Ethereum has become a popular platform for building decentralized finance (DeFi) applications, such as lending and borrowing platforms, decentralized exchanges, and stablecoins. It is also used for building other types of dapps, such as games, social networks, and supply chain management systems. As a blockchain network, Ethereum is a decentralized public ledger for verifying and recording transactions. Its cryptocurrency is now second only to Bitcoin in market value. It is the fuel that runs the network. It is used to pay for the computational resources and the transaction fees for any transaction executed on the Ethereum network. Like Bitcoins, ether is a peer-to-peer currency. Apart from being used to pay for transactions, ether is also used to buy gas, which is used to pay for the computation of any transaction made on the Ethereum network.

#### **4) METAMASK:**

Metamask acts as a bridge between websites and the Ethereum blockchain. The smart contracts in the blockchain check to see if the node in the network can access that data or not. When a local blockchain network is built, Metamask wallet can take care of the nodes. Metamask is a cryptocurrency wallet that can be added to Google Chrome. It is a way to communicate with the Ethereum blockchain. Metamask is used to store Ethereum accounts. It is safe and easy to use, and can also be used to deploy to the main Ethereum networks. Metamask is a popular cryptocurrency wallet that allows users to store, send, and receive digital assets on the Ethereum blockchain. It is a browser extension that can be used with popular web browsers like Google Chrome, Mozilla Firefox, and Brave. Metamask not only provides a secure way to store and manage cryptocurrencies but also acts as a gateway to decentralized applications (dApps) built on the Ethereum blockchain. It enables users to interact with dApps without the need to give them direct access to their private keys or funds. Some of the key features of Metamask include the ability to manage multiple Ethereum accounts, view transaction history, set transaction fees, and connect to different Ethereum networks like the mainnet, testnet, and custom networks. Metamask has gained popularity among cryptocurrency enthusiasts due to its ease of use and the

convenience it provides in interacting with the Ethereum blockchain and dApps. However, users should always exercise caution when using any cryptocurrency wallet and take appropriate measures to protect their private keys and funds. MetaMask allows users to store, send, and receive Ethereum and other ERC-20 tokens, as well as interact with Ethereum-based dApps without the need for a separate wallet or browser extension. It provides a secure and user-friendly interface for managing digital assets, including the ability to view transaction history, set transaction fees, and manage multiple Ethereum accounts.

### **5) REMIX IDE:**

Remix is a popular integrated development environment (IDE) for Ethereum smart contract development. It is an open-source tool developed by the Ethereum Foundation and is available in a web-based version as well as a desktop application.

Remix provides a range of features that make it easy for developers to create, test, and deploy smart contracts on the Ethereum blockchain. Some of its key features include:

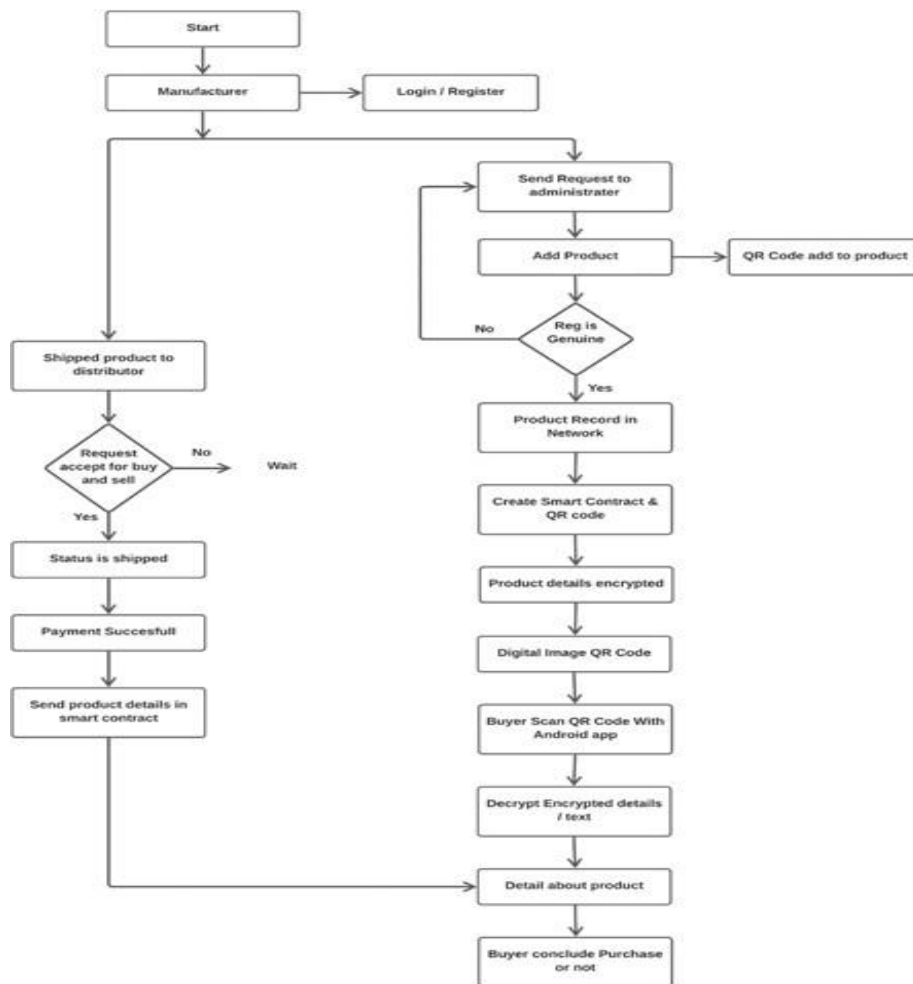
1. **Solidity Compiler:** Remix comes with a built-in Solidity compiler that allows developers to write and compile smart contracts in the Solidity programming language. It also supports other programming languages like Vyper.
2. **Code Editor:** The code editor in Remix provides syntax highlighting, code completion, and error highlighting to make coding easier and more efficient.
3. **Debugger:** Remix also comes with a built-in debugger that allows developers to step through their code and identify any issues or bugs.
4. **Gas Estimator:** The gas estimator in Remix allows developers to estimate the amount of gas required for a particular transaction, helping them optimize the cost of their smart contracts.

5. Deployment: Remix also provides an easy-to-use deployment interface that allows developers to deploy their smart contracts directly from the IDE.

6. Plugin System: Remix has a plugin system that allows developers to extend its functionality with custom plugins.

Overall, Remix is a powerful and user-friendly IDE that has become a popular choice among Ethereum developers. Its range of features and intuitive interface make it an essential tool for creating and testing smart contracts on the Ethereum blockchain.

## **4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM**



**Fig 4.2: System Architecture**

### **4.3 FINANCIAL REPORT**

The required software and frameworks are acquired. Hardware systems and devices are already owned ones. The premium version of Ganache is bought at a minimal price. Proper wifi and broadband internet is used. More sophisticated methods can be applied in real time usage.

### **4.4 DESCRIPTION OF SOFTWARE FOR IMPLEMENTATION AND TESTING PLAN OF THE PROPOSED MODEL/SYSTEM**

JavaScript is a high-level, interpreted programming language that is widely used for creating interactive web pages and web-based applications. It was first developed in the mid-1990s by Brendan Eich while working at Netscape Communications, and has since become one of the most popular programming languages in the world.

JavaScript is primarily used to add functionality to web pages, such as form validation, interactivity, and dynamic content updates without requiring a page refresh. It is also commonly used for developing web-based applications and games. JavaScript is a client-side scripting language, which means that it runs on the user's web browser rather than on the server hosting the web page. This makes it a powerful tool for creating responsive and dynamic user interfaces.

In recent years, JavaScript has expanded beyond the browser and is now used for server-side programming, desktop application development, and even mobile app development. Several popular frameworks and libraries, such as React, Angular, and Vue.js, have also been developed to simplify the process of building complex web applications.

JavaScript is a high-level, interpreted programming language that is widely used for creating interactive web pages and web-based applications. It was first introduced by Brendan Eich at Netscape in the mid-1990s and has since become one of the most



popular programming languages in the world.

One of the main features of JavaScript is its ability to add dynamic functionality to web pages, such as form validation, interactivity, and the ability to update content without requiring a page refresh. This makes it a powerful tool for creating responsive and engaging user interfaces.

JavaScript is a client-side scripting language, which means that it runs on the user's web browser rather than on the server hosting the web page. This allows for more immediate feedback to user interactions and enables the creation of more dynamic user interfaces.

In recent years, JavaScript has expanded beyond its traditional use in the browser and is now being used for server-side programming with platforms like Node.js. Additionally, many popular frameworks and libraries have been developed to simplify the process of building complex web applications, such as React, Angular, and Vue.js.

JavaScript is a versatile and powerful programming language that has played a crucial role in the evolution of the modern web. Its continued popularity and growth make it an essential skill for web developers and programmers to master.

JavaScript is a versatile programming language that has many uses in web development and beyond. One of the primary uses of JavaScript is to add interactivity and dynamic functionality to web pages. For example, JavaScript can be used to validate form data, create animations, and update content without requiring a page refresh.

JavaScript is also commonly used for web-based application development, such as online games and productivity tools. It is a popular choice for building single-page applications, which provide a more seamless user experience by updating the page dynamically rather than navigating to a new page. Beyond web development, JavaScript is being used for server-side programming with platforms like Node.js.

In summary, JavaScript is a powerful and versatile programming language that is essential for modern web development and has many other potential uses. Its

continued popularity and growth make it an important skill for developers and programmers to learn

JavaScript is a powerful programming language with many functionalities. Here are some of the most important functionalities of JavaScript:

1. **Interactivity:** JavaScript allows you to add interactivity to your web pages, making them more engaging and user-friendly. You can use JavaScript to create animations, pop-ups, and other interactive elements.

2. **DOM manipulation:** JavaScript allows you to manipulate the Document Object Model (DOM) of your web page. This means you can dynamically change the content of your web page, update styles, and even add or remove HTML elements.

3. **Form validation:** JavaScript can be used to validate user input on forms. This can help prevent errors and improve the user experience.

4. **Event handling:** JavaScript allows you to handle events such as clicks, mouseovers, and keyboard inputs. You can use event handlers to trigger functions and perform actions based on user interactions.

5. **Asynchronous programming:** JavaScript is a single-threaded language, but it allows for asynchronous programming using callbacks, promises, and async/await functions. This allows you to perform operations that take time without blocking the user interface.

**6. APIs and Ajax:** JavaScript can be used to make API calls and retrieve data from servers using Ajax. This allows you to create dynamic web applications that interact with external data sources.

**7. Object-oriented programming:** JavaScript supports object-oriented programming, allowing you to create classes, objects, and methods. This can help you organize and structure your code.

### **Functions:**

In JavaScript, a function is a block of code that performs a specific task or set of tasks. Functions can be called or invoked multiple times with different arguments, making them a useful tool for code reuse and organization. Here are some important things to know about functions in JavaScript:

**1. Defining a function:** You can define a function using the function keyword followed by a name and a set of parentheses, which can contain parameters. The function body is contained within curly braces {}. Here's an example of a function that adds two numbers:

```
function addNumbers(num1, num2) {  
  return num1 + num2;  
}
```

**2. Calling a function:** To call or invoke a function, you simply write the function name followed by a set of parentheses that contain any arguments you want to pass in. Here's an example of calling the addNumbers() function defined above:

```
var sum = addNumbers(5, 7);console.log(sum); // Output: 12
```

**3. Calling a function:** To call or invoke a function, you simply write the function name followed by a set of parentheses that contain any arguments you want to pass in. Here's an example of calling the `addNumbers()` function defined above:

```
var sum = addNumbers(5, 7);console.log(sum); // Output: 12
```

**4. Function expressions:** Functions can also be defined using function expressions, which are anonymous functions that can be assigned to a variable. Here's an example of a function expression that adds two numbers:

```
var addNumbers = function(num1, num2) {  
    return num1 + num2;  
};  
var sum = addNumbers(5, 7);console.log(sum); // Output: 12
```

**5. Arrow functions:** Arrow functions provide a concise syntax for defining functions. They are similar to function expressions but use an arrow (`=>`) instead of the function keyword. Here's an example of an arrow function that adds two numbers:

```
var addNumbers = (num1, num2) => num1 + num2;  
var sum = addNumbers(5, 7);console.log(sum); // Output: 12
```

#### **4.4.1 EXISTING SYSTEM:**

The existing system for product verification using blockchain technology involves the use of decentralized applications (dApps) built on top of the Ethereum blockchain. These dApps provide a secure, transparent, and tamper-proof system for verifying the authenticity of products. One example of such a dApp is the Blockcerts Wallet, which allows users to store and verify details of the products on the blockchain. The Blockcerts system uses the Bitcoin and Ethereum blockchains to store and verify products, with each product represented as a unique digital asset on the blockchain. To verify a product, the verifier enters the product ID or hash into the Blockcerts Wallet, which retrieves the relevant data from the blockchain, including the hashed product and any other relevant information. The wallet then compares the hashed product retrieved from the blockchain with the hashed product provided by the verifier, and if the two match, the product is considered authentic and valid.

In this case, the use of blockchain technology provides a secure and tamper-proof system for verifying the authenticity of products, eliminating the need for manual verification processes and reducing the risk of fraud and tampering. The traditional system for product identification involves a central authority. With the advent of blockchain technology, there has been a shift towards a decentralized system for product identification. In this system, products details are stored and verified using blockchain technology, making the process more secure, transparent, and tamper-proof. One of the most prominent examples of a blockchain-based product verification system is the Blockcerts project, developed by Learning Machine in collaboration with the MIT Media Lab.

#### **Drawbacks of the existing system:**

While blockchain technology has the potential to revolutionize the product verification process, there are still some drawbacks and challenges associated with the existing system. Some of the main drawbacks include.

**Complexity:** Blockchain technology can be complex and difficult to understand for

those who are not familiar with it. This can make it challenging for manufacturers and individuals to adopt and implement blockchain-based product identification systems.

**Limited Adoption:** While blockchain-based product identification systems are gaining popularity, they are still not widely adopted by product manufacturer and other organizations. This can limit their usefulness and impact.

**Dependence on Public Key Infrastructure:** Most blockchain-based product verification systems rely on public key infrastructure (PKI) for secure communication and verification. However, PKI can be complex and difficult to manage, which can limit the scalability and effectiveness of blockchain-based product identification systems.

**Performance Issues:** The performance of blockchain-based product identification systems can be slower than traditional verification systems due to the time required for transactions to be processed and verified on the blockchain.

**Cost:** Blockchain-based product verification systems can be expensive to develop and maintain, which can limit their adoption by smaller brands and organizations.

**Dependence on Blockchain Technology:** The existing system for product verification using blockchain technology relies heavily on the reliability and security of the underlying blockchain technology. If the blockchain is compromised or fails, it can potentially compromise the integrity of the entire system.

Overall, while blockchain technology offers many benefits for product verification, there are still challenges that need to be addressed to ensure that these systems are practical, scalable, and accessible for all users.

#### **4.4.2 PROPOSED SYSTEM**

A proposed system for product identification using blockchain technology would aim to leverage the advantages of blockchain technology to create a more secure, efficient, and

user-friendly way to verify products. The system would be decentralized, meaning it would not rely on a central authority to verify products, which would make it more secure and transparent. The system would also be designed with a user-friendly interface to make it accessible to individuals with little or no technical expertise.

The proposed system would support multiple blockchain platforms to provide users with a range of options based on their specific needs and requirements. It would also use smart contracts to automate the verification process, which would reduce the need for manual intervention and help ensure that products are verified accurately and efficiently. The system would aim to reduce transaction fees and automate the verification process using smart contracts, providing a more efficient and reliable way to verify products while addressing the limitations of the existing system.

Additionally, the proposed system would be scalable to handle a large volume of product and users, making it suitable for use by e-shopping apps, governments, and other organizations that need to verify a large number of products.

The proposed system would also aim to address some of the drawbacks of the existing system, such as high transaction fees and technical complexity. The system would optimize the verification process to reduce transaction fees and use more efficient blockchain protocols to make the system more cost-effective. It would also be designed to be user-friendly, with a simple and intuitive interface to make it accessible to individuals with little or no technical expertise.

In summary, a proposed system for product verification using blockchain technology would aim to provide a more efficient, secure, and user-friendly way to verify products. By leveraging the benefits of blockchain technology and addressing the limitations of the existing system, a proposed system could help to improve the verification process and increase the adoption of products.

## **MODIFICATION SYSTEM**

A modified system for product verification using blockchain technology would aim to

address the limitations of the existing system while leveraging the benefits of blockchain technology. The modified system could be designed to integrate with existing product verification systems used by e-shopping apps, government agencies, and other organizations. This would provide a seamless experience for users and enable a smoother transition to the new system.

The modified system could also use blockchain interoperability to enable communication between different blockchain platforms. This would increase the system's flexibility and utility, allowing users to verify products across multiple blockchain networks.

Consensus mechanisms could also be used in the modified system to ensure the accuracy of the verification process. This could involve using a network of nodes to verify each product and reach a consensus on its authenticity. The modified system could also be designed with a user-friendly interface that is easy to navigate and understand, reducing the learning curve for users and making the verification process more accessible.

The modified system would be scalable, with the ability to handle a large volume of products and users. This would make it suitable for use by e-shopping, government agencies, and other organizations that need to verify a large number of products. The modified system could also aim to reduce transaction fees by using more efficient blockchain protocols and optimizing the verification process.

Overall, a modified system for product verification using blockchain technology would aim to improve upon the existing system by incorporating additional features to enhance security, usability, and scalability. By leveraging the benefits of blockchain technology a modified system could provide a more efficient and reliable way to verify products.

### **Advantages:**



**Increased Security:** Blockchain technology provides a secure and tamper-proof way to store and verify products. By using cryptographic algorithms and distributed ledger technology, it is difficult for fraudsters to manipulate or falsify products.

**Decentralization:** Blockchain technology operates in a decentralized manner, meaning that there is no central authority controlling the system. This makes it more resilient to hacking or other forms of attack, as there is no single point of failure.

**Transparency:** The use of blockchain technology for product verification enables transparency in the verification process. Anyone can view the blockchain and verify the authenticity of products, which helps to increase trust in the system.

**Efficiency:** The verification process is automated and streamlined when using blockchain technology. This reduces the time and effort required to verify products, which can be particularly beneficial for organizations that need to verify a large volume of products.

**Lower Costs:** By using blockchain technology for product verification, costs can be reduced compared to traditional methods of verification. This is because blockchain technology eliminates the need for intermediaries and reduces the amount of paperwork involved in the verification process.

**Portable:** Blockchain technology enables products to be stored and verified globally, meaning that they can be easily accessed and verified from anywhere in the world. This is particularly useful for individuals who have earned qualifications in different countries or who work remotely.

**Immune to Forgery:** Blockchain technology provides a tamper-proof and immutable record of products, which cannot be falsified or manipulated. This helps to eliminate the risk of fraud and ensures the authenticity of products.

## **4.5 PROJECT MANAGEMENT PLAN**

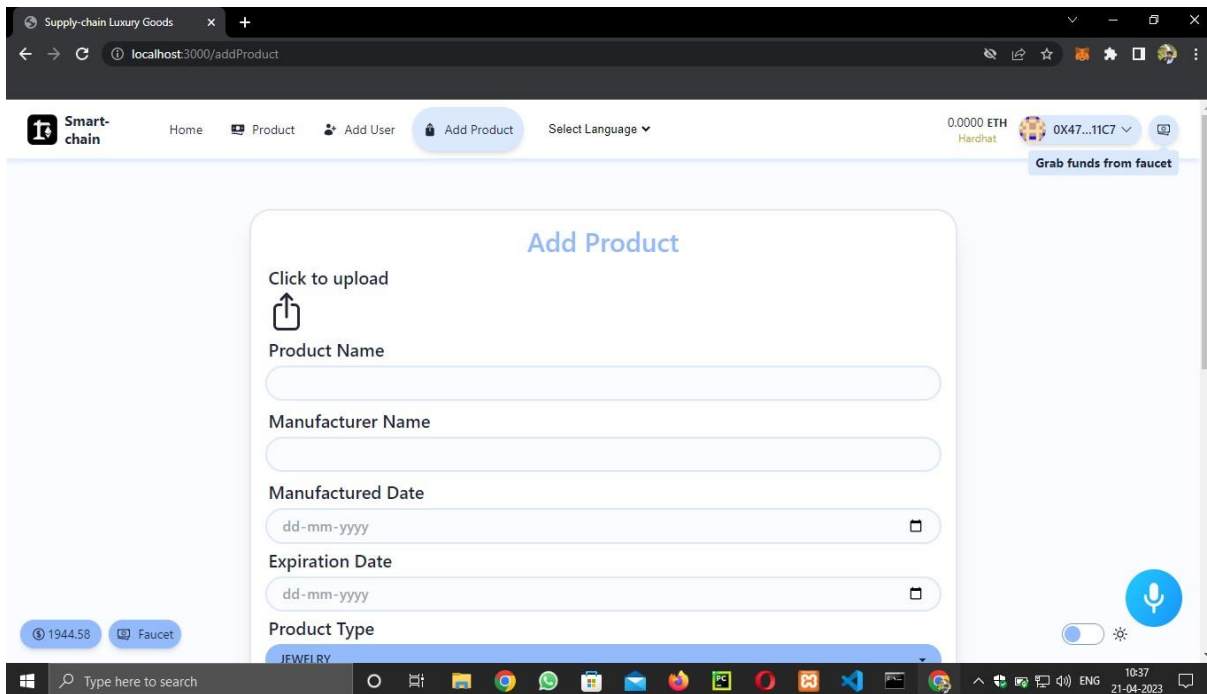
The following are the modules of the project, which is planned in aid to complete the project with respect to the proposed system, while overcoming existing system and also providing the support for the future enhancement.

### **MODULE LIST:**

1. Uploading Products
2. Generating Qr Code
3. Add Users
4. Consensus Mechanism

### **1) Uploading Products:**

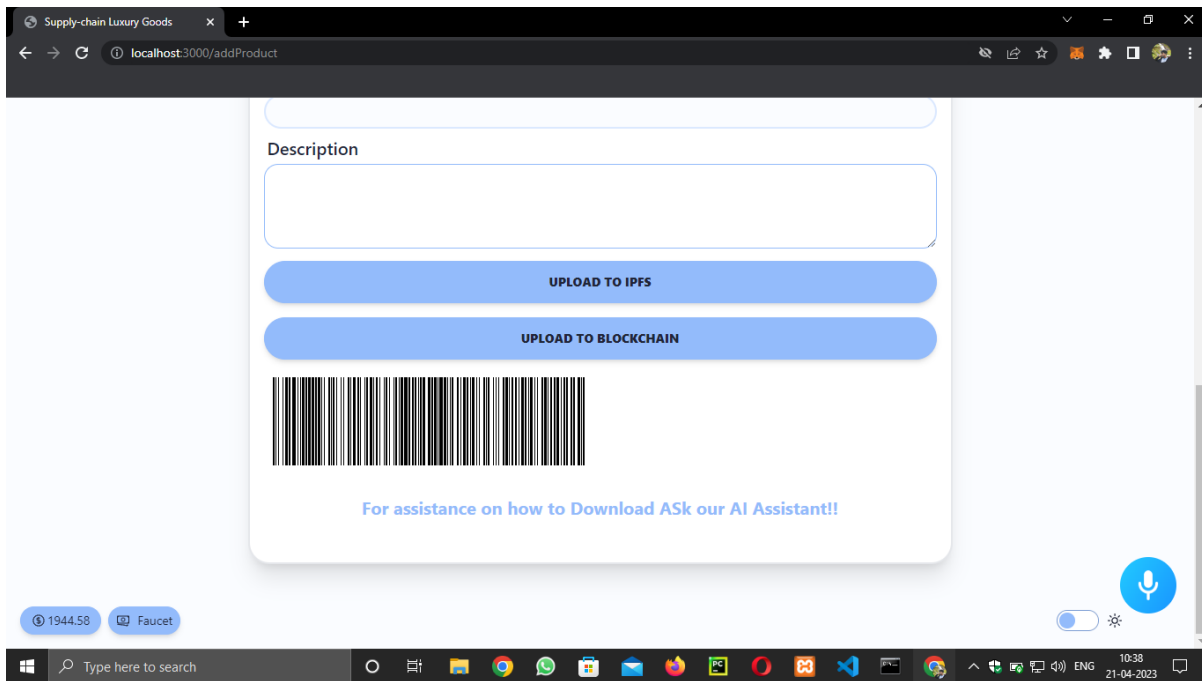
This module will allow brands to manufacture products on the blockchain. The module will enable product manufacturers to input product data, verify the identity of the product manufacturer, and issue the product details on the blockchain. The module will ensure that only valid and authenticated products are issued



**Fig 4.5.1: Upload product details**

## **2) Generating QR Code:**

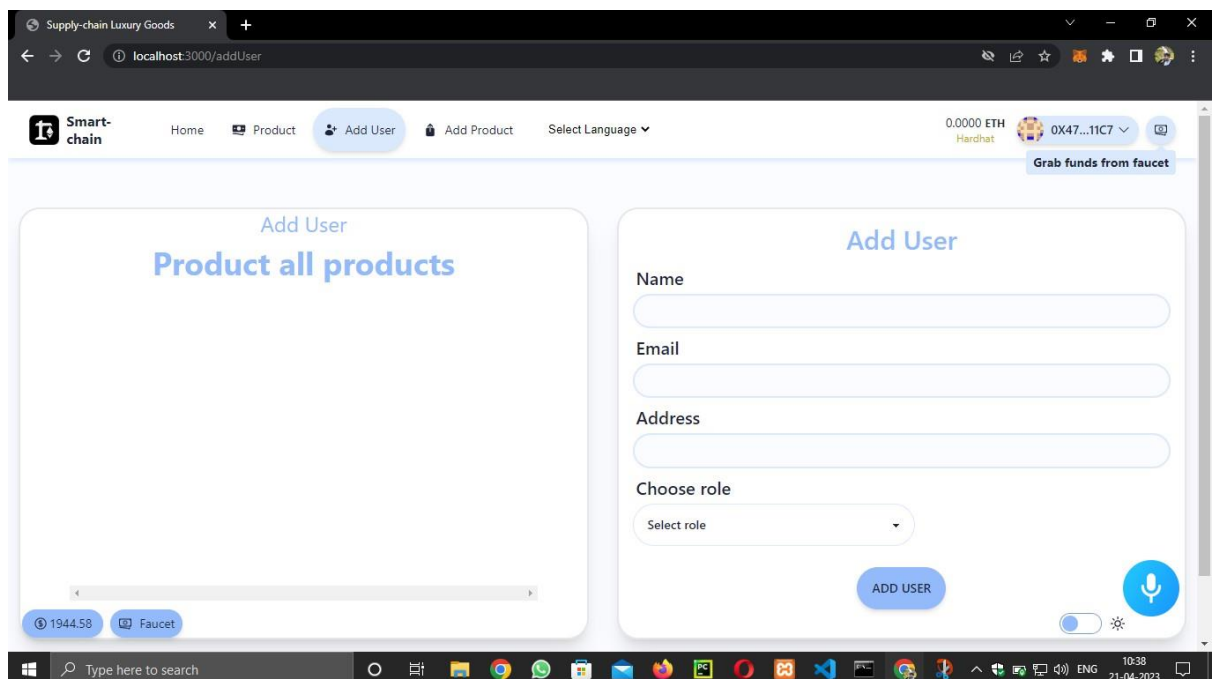
This module will allow manufacturers to generate the qr code for products by accessing the blockchain. The module will enable manufactureres to generates the qr code and verify its authenticity on the blockchain. The module will ensure that only valid and authenticated products are accepted.



**Fig 4.5.2: Generate QR Code**

### 3)Add Users :

This module will allows users to add users to the web page.In this module the manufacturers can add their staffs like managers or retailers.So that they can update or modify the product details.



**Fig 4.5.3: Add user**

#### **4) Consensus Mechanism Module:**

This module will ensure the accuracy of the verification process by using consensus mechanisms to reach a consensus on the validity of products. The module will ensure that the verification process is secure and tamper-proof.

## **CHAPTER 5 IMPLEMENTATION DETAILS**

### **1.1 DEVELOPMENT AND DEPLOYMENT SETUP**

The development and deployment setup for the proposed system for product verification using blockchain technology is a complex process that involves several key steps. The first step is to select the appropriate blockchain platform for the project, which will depend on the specific requirements of the project, including scalability, performance, and security. Once the blockchain platform has been selected, the next step is to develop the smart contracts that will be used to manage the product details storing and verification processes. Smart contracts are self-executing programs that run on the blockchain, and can be used to automate the verification and storing processes. The user interface (UI) is an important component of the system, as it will be the primary interface for users to interact with the blockchain. The UI can be developed using web-based technologies such as HTML, CSS, and JavaScript, and can be integrated with the blockchain platform using tools such as Web3.js or Truffle. The backend of the system will be responsible for managing the interactions between the UI and the blockchain platform. This can be achieved using server-side technologies such as Node.js, Python, or Java. Once the development is complete, the system should be thoroughly tested to ensure that it meets the requirements and is free of bugs and vulnerabilities. Finally, the system can be deployed to a production environment, which will involve deploying the smart contracts to the blockchain and hosting the UI and backend on a server. The process can be streamlined by using blockchain development frameworks such as Truffle, which provide pre-built templates

and tools for developing and deploying dApps on the blockchain. Overall, the development and deployment setup for the proposed system for product verification using blockchain technology requires expertise in blockchain development, UI development, and backend development, and must be approached with careful planning and attention to detail to ensure a successful deployment.

### **1) Anaconda:**

Anaconda is an open-source package manager for Python and R. It is the most popular platform among data science professionals for running Python and R implementations. There are over 300 libraries in data science, so having a robust distribution system for them is a must for any professional in this field. Anaconda simplifies package deployment and management. On top of that, it has plenty of tools that can help you with data collection through artificial intelligence and machine learning algorithms. With Anaconda, you can easily set up, manage, and share Conda environments. Moreover, you can deploy any required project with a few clicks when you're using Anaconda.

### **2) Solidity:**

Solidity is an object-oriented programming language for implementing smart contracts on various blockchain platforms, most notably, Ethereum. It was developed by Christian Reitwiessner. Solidity is user-friendly for anyone with basic familiarity with other coding languages. Programs in Solidity run on Ethereum Virtual Machine. It is influenced by C++, Python and JavaScript. Solidity is compiled into bytecode, which can be executed by the EVM. Solidity has become one of the most popular languages for blockchain development, and it is used by many apps and blockchain platforms.

### **3) Ethereum:**

Ethereum is an open-source blockchain platform that was launched in 2015. It was

created by Vitalik Buterin, a cryptocurrency researcher and programmer, with the goal of enabling developers to build decentralized applications (dApps) that execute smart contracts. Ethereum is built on a decentralized, distributed network of computers that work together to process transactions and execute smart contracts. One of the key features of Ethereum is its support for smart contracts, which are self-executing contracts that run on the Ethereum blockchain. Smart contracts are written in programming languages like Solidity and are used to automate the terms of a contract.

## **1.2 ALGORITHMS USED**

### **1) Asymmetric Key Encryption**

Asymmetric key encryption, also known as public-key encryption, is a type of encryption where two separate keys, one public and one private, are used to encrypt and decrypt data. The public key can be shared with anyone, while the private key is kept secret. In this type of encryption, the sender uses the recipient's public key to encrypt the message. The recipient then uses their private key to decrypt the message. This process is secure because only the recipient has access to their private key, which is required to decrypt the message.

```
C:\Windows\System32\cmd.e X + v
EthereumJS TestRPC v6.0.3 (ganache-core: 2.0.2)

Available Accounts
=====
(0) 0xd77ccdf564e347bdc7a262ed75c7aa0b0ad0a4ee
(1) 0x6f1eb72c582ef7450277fd62f23d04c535694f43
(2) 0x1c7332a73bad133bb575b1d45c3cd5eccdf7b89
(3) 0x729f117121962bae223020839bf5b6012103d185
(4) 0xe78758ee254a96fec719d658314ae96241258a93
(5) 0x2433a07def4b7def599412a1f2b0a509e653db66
(6) 0xf3a7329559328e3bf8851be69b63f5efa1099d2
(7) 0xd36711bce8b5294108ea3d808333f5e08b660e62
(8) 0xb7be8ba45c4bfe2a5896aa41c60c95df3ae2cb7c
(9) 0x811b30c324b9a16a0c8ff86b85eb7a07899b8914

Private Keys
=====
(0) 1c6baa3ceede678f70dda7fd632dbccb4ca1169770b9ba98c85c3e0afe901490
(1) e6a6a123532e1beff89f905a89ef86cbf98bb89caa415b085e2bb90c518aac97
(2) 20d3ed7eae9ff5c0453800213704e03501e29a81cd89a55877876aadcbcbdae4
(3) 65c911b313385ae61c4e6a645a4fe7198dc01c8a828cabf824a99d693816f554
(4) 6901e610e49518025e02e047625230abf8324f1d48a9fdf8acbf185740dd55d9
(5) 04bcbccc2995e8f709d1009ec1cfe5c4f3c190c43da27a17dd4839460bfa8871
(6) 0ac6817bdd4f4115adaaf193f1a1fe0b7954e3dcdec9c33873149ad9cdd7f8c2
(7) 5c9521ec4c2619ce43a393b5038097ee65087d253ae8afd71c0db9d2e9845cdb
(8) 6e24f695e91ab8b5570bdf8fdfee7a8b2309bce338c1fcf15f9b9c599607a719
(9) 9b82c27aa27dba9f227ffda5b0124c5ea909a8f7bdda3d96d12148ea1e3e0618
```

**Fig 5.1: Asymmetric key encryption**

Asymmetric key encryption is commonly used for secure communication over the internet, such as in HTTPS, SSL, and TLS protocols. It is also used in digital signatures and for securing electronic transactions. Some of the advantages of asymmetric key encryption include:

**Security:** Asymmetric key encryption provides a high level of security because the private key is never shared, making it difficult for unauthorized parties to decrypt the message.

**Convenience:** Asymmetric key encryption is convenient because the recipient's public key can be shared widely, making it easy for anyone to send a secure message to the recipient without needing to know their private key.

**Authentication:** Asymmetric key encryption can be used for authentication because the private key can be used to sign a message, proving that the message was sent by the owner of the private key.



One potential of asymmetric key encryption is that it is generally slower than symmetric key encryption, which uses the same key for both encryption and decryption. As a result, asymmetric key encryption is often used for securing small amounts of data or for key exchange in symmetric key encryption.

Asymmetric key encryption is commonly used for various security purposes, including secure data transmission, digital signatures, and authentication. It provides a way to secure communication and protect data confidentiality, integrity, and authenticity in various applications, including online transactions, secure email communication, secure file transfer, and secure access to online services.

## **2) Digital Signature:**

A digital signature is a mathematical technique used to verify the authenticity and integrity of a digital document or message. It is a type of electronic signature that uses encryption and decryption to ensure that the document or message has not been altered or tampered with during transmission. The process of creating a digital signature involves the following steps: The sender generates a private key and a corresponding public key using a digital signature algorithm, such as RSA or DSA.

The sender uses the private key to create a digital signature of the document or message, which is essentially a hash of the data that has been encrypted using the sender's private key. The sender sends the document or message, along with the digital signature, to the recipient. The recipient uses the sender's public key to decrypt the digital signature and verify that the document or message has not been altered during transmission. If the digital signature is valid, it provides assurance that the document or message is authentic and has not been tampered with. If the digital signature is not valid, it indicates that the document or message has been altered or may have been sent by an unauthorized party.

Digital signatures are commonly used in electronic transactions, such as online banking, e-commerce, and legal contracts. They provide a high level of security and

help to prevent fraud and identity theft.

### **3) SHA-256:**

Secure Hash Algorithm 256 (SHA-256) is a cryptographic hash function that is used to generate a fixed-length, 256-bit (32-byte) hash value. It is a widely used algorithm for data security and is considered to be a secure and robust method for protecting sensitive data. The SHA-256 algorithm takes an input message of any length and generates a fixed-length output, which is called a hash value.

This hash value is unique to the input message, which means that even a small change in the input message will result in a completely different hash value.

This property of the SHA-256 algorithm makes it ideal for verifying the integrity of data and ensuring that it has not been tampered with.

SHA-256 is widely used in digital signatures, digital certificates, and secure communications protocols such as HTTPS, SSL, and TLS. It is considered to be a secure hash function and is widely adopted in many industries for its robustness and efficiency. The SHA-256 algorithm takes an input message, processes it in fixed-size blocks, and applies a series of bitwise operations, logical operations, and modular arithmetic operations to generate a 256-bit hash value.

The output hash is deterministic, meaning that the same input data will always produce the same hash value, and any small change in the input data will produce a significantly different hash value, due to the avalanche effect. It is also used in crypto currencies such as Bitcoin to secure transactions and to protect the integrity of the blockchain. Some of the key features of SHA-256 include:

**Collision resistance:** It is extremely difficult to find two different input messages that produce the same hash value.

**Deterministic:** The hash value is always the same for a given input message, which makes it useful for verifying data integrity.

**One-way function:** It is virtually impossible to reverse-engineer the input message from the hash value.

Overall, SHA-256 is a powerful tool for securing data and ensuring its integrity. However, it should be used in conjunction with other security measures, such as encryption and access controls, to provide a comprehensive approach to data security.

#### **4) Blockchain Consensus Algorithm:**

Blockchain consensus algorithm is a method used by a distributed network of nodes to agree on the state of a shared ledger without the need for a central authority.

There are various consensus algorithms used in blockchain, such as Proof of Work (PoW), Proof of Stake (PoS), and Delegated Proof of Stake (DPoS).

In PoW, nodes compete to solve complex mathematical puzzles, and the first node to solve the puzzle is rewarded with a new block and the right to add it to the chain. This process requires significant computational power, which makes it resource-intensive.

In PoS, nodes are chosen to validate transactions based on their stake in the network. The more tokens a node holds, the more likely it is to be selected to validate transactions. This process is less resource-intensive than PoW.

In DPoS, stakeholders vote for a group of nodes, known as delegates, who are responsible for validating transactions and creating new blocks. This process is faster than PoW and PoS and is used by some blockchain platforms.

Overall, consensus algorithms play a crucial role in maintaining the integrity and security of blockchain networks by ensuring that all nodes agree on the state of the ledger.

## CHAPTER 6

### RESULTS AND DISCUSSION

Product verification using blockchain technology involves storing details of products on a blockchain to ensure their authenticity and prevent fraud. When a product details is issued, it is cryptographically signed and added to the blockchain, creating an immutable record of the product details.

To verify a product, the recipient can access the blockchain and verify the signature on the product. If the signature matches the one stored on the blockchain, the product is considered authentic. The use of blockchain technology for product verification can improve the efficiency and security of the verification process. It eliminates the need for intermediaries such as notaries, reduces the risk of fraud, and provides a tamper-proof record of product.

product verification using blockchain technology involves storing product details on a blockchain to ensure their authenticity and prevent tampering.

When a product details is stored, it is hashed and stored on the blockchain, creating a permanent and unalterable record. Recipients of the product can then verify its authenticity by checking the hash on the blockchain.

This process provides several benefits, including increased security and transparency, as well as reduced costs and time for verifying products. Additionally, blockchain-based product verification can help prevent fraud and improve trust between parties.

By storing details of products on a blockchain, they can be accessed and verified by anyone, at any time, without the need for a centralized authority. The blockchain network provides a secure and transparent platform for issuing, storing, and verifying products, which can help reduce fraud and improve efficiency

As a result, product verification using blockchain technology can improve the credibility

and trust of products, increase the transparency of the verification process, and reduce the cost and time required for verification

## **CHAPTER 7**

### **CONCLUSION**

#### **7.1 CONCLUSION**

In conclusion, the use of blockchain technology for product identification/verification has significant potential to streamline the verification process, increase efficiency, and reduce fraud in various industries. The decentralized, tamper-proof nature of blockchain technology ensures that products are authentic, secure, and immutable, and can be accessed and verified by anyone at any time, without the need for a central authority.

Blockchain-based product identification systems offer numerous benefits, including increased transparency, trust, and security, while reducing the administrative burden associated with traditional paper-based product delivery systems. This technology has the potential to revolutionize the way products are manufactured, stored, and verified, ultimately resulting in a more efficient and trustworthy system that can benefit individuals, organizations, and society as a whole

#### **7.2 FUTURE WORK**

There is significant potential for future work on product verification using blockchain technology. One area for future exploration is the integration of blockchain-based product verification systems with existing systems and processes to increase interoperability and adoption. As usage of blockchain-based product verification systems increases, there may be a need to scale the underlying blockchain network to accommodate higher transaction volumes and improve performance. Improving the user experience of blockchain-based product verification systems, such as developing user-friendly interfaces and ensuring compatibility with mobile devices, could increase

adoption and usage. Finally, ensuring compliance with data protection laws, such as GDPR, and developing best practices for data privacy and security, will be critical to the success of blockchain-based product verification systems. By addressing these areas of future work, the potential benefits of blockchain technology for product verification can be fully realized, resulting in a more efficient and trustworthy system that benefits individuals, organizations, and society as a whole.

### **7.3 RESEARCH ISSUES:**

There are several research issues related to product verification using blockchain technology that merit further exploration. One important research issue is the scalability of blockchain networks for product verification systems, as increased transaction volumes could lead to potential performance issues. Another research issue is the development of more robust consensus mechanisms that can withstand attacks and ensure the integrity of the blockchain network. Finally, there is a need for more research on the legal and regulatory implications of blockchain-based product verification systems, particularly regarding data privacy and security. Addressing these research issues will be critical to the successful development and implementation of blockchain-based product verification systems, ultimately resulting in a more efficient, secure, and trustworthy system.

### **7.4 IMPLEMENTATION ISSUES**

There are several implementation issues related to product verification using blockchain technology that should be considered. One implementation issue is the cost associated with implementing and maintaining blockchain-based product verification systems, including the cost of network infrastructure and ongoing maintenance and upgrades. Another implementation issue is the need for interoperability between different blockchain networks to ensure that product details stored on one blockchain network can be verified on another. Additionally, there may be challenges related to integrating blockchain-based product verification systems with existing systems and

processes, particularly regarding data migration and system compatibility. Another implementation issue is the need to ensure data privacy and security, particularly in instances where sensitive personal information is stored on the blockchain network.

## **REFERENCES: -**

[1] Si Chen, Rui Shi, Zhuangyu Ren, Jiaqi Yan, Yani Shi, Jinyu Zhang, “A Blockchain-based Supply Chain Quality Management Framework”, 14th, IEEE International Conference on e-Business Engineering, 2017.

[2] Mitsuaki Nakasumi, “Information Sharing for Supply Chain Management based on Block Chain Technology”, 19th Conference on Business Informatic, IEEE, 2017.

[3] Daniel Tse, Bowen Zhang, Yuchen Yang, Chenli Cheng, Haoran Mu, “Blockchain Application in Food Supply Information Security”, 2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM).

[4] Feng Tian, “A supply chain traceability system for food safety based on HACCP, blockchain & Internet of things”, 2017 International Conference on Service Systems and Service Management

[5] Freya Sheer Hardwick, Apostolos Gioulis, Raja Naeem Akram, Konstantinos Markantonakis, “E-Voting with Blockchain: An E-Voting Protocol with Decentralisation and Voter Privacy”, 2018.

[6] Neo C.K. Yiu, Member, “Toward Blockchain-Enabled Supply Chain Anti-Counterfeiting and Traceability” IEEE Department of Computer Science, University of Oxford.

[7] A Blockchain-Based Application System for Product Anti-Counterfeiting Jinhua Ma, Shih-Ya Lin, Xin Chen, Hung-Min Sun, Yeh-Cheng Chen, (Graduate Student Member, IEEE) and Huaxiong Wang

[8] A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain Kentaroh Toyoda, (Member, IEEE), P. Takis Mathiopoulos, (Senior Member, IEEE), Iwao Sasase, (Senior Member, IEEE), And Tomoaki Ohtsuki, (Senior Member, IEEE).

## APPENDIX

### A. SOURCE CODE

#### *BLOCKCHAIN SERVER*

```
{
  "_format": "hh-sol-artifact-1",
  "contractName": "SupplyChain",
  "sourceName": "contracts/SupplyChain.sol",
  "abi": [
    {
      "inputs": [
        {
          "internalType": "string",
          "name": "name_",
          "type": "string"
        },
        {
          "internalType": "string",
          "name": "email_",
          "type": "string"
        }
      ],
      "stateMutability": "nonpayable",
      "type": "constructor"
    },
    {
      "anonymous": false,
      "inputs": [
        {
          "indexed": false,
          "internalType": "string",
          "name": "name",
          "type": "string"
        }
      ],
    }
  ]
}
```



```

        "indexed": false,
        "internalType": "string",
        "name": "email",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "enum Types.UserRole",
        "name": "role",
        "type": "uint8"
    }
],
"name": "LostUser",
"type": "event"
},
{
    "anonymous": false,
    "inputs": [
        {
            "indexed": false,
            "internalType": "string",
            "name": "name",
            "type": "string"
        },
        {
            "indexed": false,
            "internalType": "string",
            "name": "manufacturerName",
            "type": "string"
        },
        {
            "indexed": false,
            "internalType": "string",
            "name": "barcodeId",
            "type": "string"
        }
    ],
    {
        "indexed": false,
        "internalType": "uint256",
        "name": "manDateEpoch",
        "type": "uint256"
    },
    {
        "indexed": false,
        "internalType": "uint256",
        "name": "expDateEpoch",
        "type": "uint256"
    }
]

```

```

    }
  ],
  "name": "NewProduct",
  "type": "event"
},
{
  "anonymous": false,
  "inputs": [
    {
      "indexed": false,
      "internalType": "string",
      "name": "name",
      "type": "string"
    },
    {
      "indexed": false,
      "internalType": "string",
      "name": "email",
      "type": "string"
    },
    {
      "indexed": false,
      "internalType": "enum Types.UserRole",
      "name": "role",
      "type": "uint8"
    }
  ],
  "name": "NewUser",
  "type": "event"
},
{
  "anonymous": false,
  "inputs": [
    {
      "indexed": false,
      "internalType": "string",
      "name": "name",
      "type": "string"
    },
    {
      "indexed": false,
      "internalType": "string",
      "name": "manufacturerName",
      "type": "string"
    },
    {
      "indexed": false,

```

```

        "internalType": "string",
        "name": "barcodeId",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "string",
        "name": "buyerName",
        "type": "string"
    },
    {
        "indexed": false,
        "internalType": "string",
        "name": "buyerEmail",
        "type": "string"
    }
],
"name": "ProductOwnershipTransfer",
"type": "event"
},
{
    "inputs": [
        {
            "components": [
                {
                    "internalType": "enum Types.UserRole",
                    "name": "role",
                    "type": "uint8"
                },
                {
                    "internalType": "address",
                    "name": "id_",
                    "type": "address"
                }
            ],
            "internalType": "string",
            "name": "name",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "email",
            "type": "string"
        }
    ],
    "internalType": "struct Types.UserDetails",
    "name": "user_",

```

```

        "type": "tuple"
    }
],
"name": "addParty",
"outputs": [],
"stateMutability": "nonpayable",
"type": "function"
},
{
    "inputs": [
        {
            "components": [
                {
                    "internalType": "string",
                    "name": "name",
                    "type": "string"
                },
                {
                    "internalType": "string",
                    "name": "manufacturerName",
                    "type": "string"
                },
                {
                    "internalType": "address",
                    "name": "manufacturer",
                    "type": "address"
                },
                {
                    "internalType": "uint256",
                    "name": "manDateEpoch",
                    "type": "uint256"
                },
                {
                    "internalType": "uint256",
                    "name": "expDateEpoch",
                    "type": "uint256"
                },
                {
                    "internalType": "bool",
                    "name": "isInBatch",
                    "type": "bool"
                },
                {
                    "internalType": "uint256",
                    "name": "batchCount",
                    "type": "uint256"
                }
            ]
        }
    ]
}

```

```

        {
            "internalType": "string",
            "name": "barcodeId",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "imageURI",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "productMetaDataURI",
            "type": "string"
        },
        {
            "internalType": "enum Types.ProductType",
            "name": "productType",
            "type": "uint8"
        }
    ],
    "internalType": "struct Types.Product",
    "name": "product_",
    "type": "tuple"
},
{
    "internalType": "uint256",
    "name": "currentTime_",
    "type": "uint256"
}
],
"name": "addProduct",
"outputs": [],
"stateMutability": "nonpayable",
"type": "function"
},
{
    "inputs": [],
    "name": "getAllProducts",
    "outputs": [
        {
            "components": [
                {
                    "internalType": "string",
                    "name": "name",
                    "type": "string"
                }
            ]
        }
    ]
},

```

```

{
  "internalType": "string",
  "name": "manufacturerName",
  "type": "string"
},
{
  "internalType": "address",
  "name": "manufacturer",
  "type": "address"
},
{
  "internalType": "uint256",
  "name": "manDateEpoch",
  "type": "uint256"
},
{
  "internalType": "uint256",
  "name": "expDateEpoch",
  "type": "uint256"
},
{
  "internalType": "bool",
  "name": "isInBatch",
  "type": "bool"
},
{
  "internalType": "uint256",
  "name": "batchCount",
  "type": "uint256"
},
{
  "internalType": "string",
  "name": "barcodeId",
  "type": "string"
},
{
  "internalType": "string",
  "name": "imageURI",
  "type": "string"
},
{
  "internalType": "string",
  "name": "productMetaDataURI",
  "type": "string"
},
{
  "internalType": "enum Types.ProductType",

```

```

        "name": "productType",
        "type": "uint8"
    }
],
"internalType": "struct Types.Product[]",
"name": "",
"type": "tuple[]"
}
],
"stateMutability": "view",
"type": "function"
},
{
    "inputs": [
        {
            "internalType": "address",
            "name": "id_",
            "type": "address"
        }
    ],
    "name": "getMyDetails",
    "outputs": [
        {
            "components": [
                {
                    "internalType": "enum Types.UserRole",
                    "name": "role",
                    "type": "uint8"
                },
                {
                    "internalType": "address",
                    "name": "id_",
                    "type": "address"
                },
                {
                    "internalType": "string",
                    "name": "name",
                    "type": "string"
                },
                {
                    "internalType": "string",
                    "name": "email",
                    "type": "string"
                }
            ]
        },
        {
            "internalType": "struct Types.UserDetails",
            "name": "",

```

```

        "type": "tuple"
    }
],
"stateMutability": "view",
"type": "function"
},
{
    "inputs": [],
    "name": "getMyProducts",
    "outputs": [
        {
            "components": [
                {
                    "internalType": "string",
                    "name": "name",
                    "type": "string"
                },
                {
                    "internalType": "string",
                    "name": "manufacturerName",
                    "type": "string"
                },
                {
                    "internalType": "address",
                    "name": "manufacturer",
                    "type": "address"
                },
                {
                    "internalType": "uint256",
                    "name": "manDateEpoch",
                    "type": "uint256"
                },
                {
                    "internalType": "uint256",
                    "name": "expDateEpoch",
                    "type": "uint256"
                },
                {
                    "internalType": "bool",
                    "name": "isInBatch",
                    "type": "bool"
                },
                {
                    "internalType": "uint256",
                    "name": "batchCount",
                    "type": "uint256"
                }
            ]
        }
    ]
}

```



```

        {
            "internalType": "string",
            "name": "barcodeId",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "imageURI",
            "type": "string"
        },
        {
            "internalType": "string",
            "name": "productMetaDataURI",
            "type": "string"
        },
        {
            "internalType": "enum Types.ProductType",
            "name": "productType",
            "type": "uint8"
        }
    ],
    "internalType": "struct Types.Product[]",
    "name": "",
    "type": "tuple[]"
}
],
"stateMutability": "view",
"type": "function"
},
{
    "inputs": [
        {
            "internalType": "address",
            "name": "id_",
            "type": "address"
        }
    ],
    "name": "getMyUsersList",
    "outputs": [
        {
            "components": [
                {
                    "internalType": "enum Types.UserRole",
                    "name": "role",
                    "type": "uint8"
                }
            ],
            "type": "uint8"
        }
    ]
}

```

```

        "internalType": "address",
        "name": "id_",
        "type": "address"
    },
    {
        "internalType": "string",
        "name": "name",
        "type": "string"
    },
    {
        "internalType": "string",
        "name": "email",
        "type": "string"
    }
],
"internalType": "struct Types.UserDetails[]",
"name": "usersList_",
"type": "tuple[]"
}
],
"stateMutability": "view",
"type": "function"
},
{
    "inputs": [
        {
            "internalType": "string",
            "name": "barcodeId_",
            "type": "string"
        }
    ],
    "name": "getSingleProduct",
    "outputs": [
        {
            "components": [
                {
                    "internalType": "string",
                    "name": "name",
                    "type": "string"
                },
                {
                    "internalType": "string",
                    "name": "manufacturerName",
                    "type": "string"
                }
            ],
            "internalType": "address",

```

```

        "name": "manufacturer",
        "type": "address"
    },
    {
        "internalType": "uint256",
        "name": "manDateEpoch",
        "type": "uint256"
    },
    {
        "internalType": "uint256",
        "name": "expDateEpoch",
        "type": "uint256"
    },
    {
        "internalType": "bool",
        "name": "isInBatch",
        "type": "bool"
    },
    {
        "internalType": "uint256",
        "name": "batchCount",
        "type": "uint256"
    },
    {
        "internalType": "string",
        "name": "barcodeId",
        "type": "string"
    },
    {
        "internalType": "string",
        "name": "imageURI",
        "type": "string"
    },
    {
        "internalType": "string",
        "name": "productMetaDataURI",
        "type": "string"
    },
    {
        "internalType": "enum Types.ProductType",
        "name": "productType",
        "type": "uint8"
    }
],
"internalType": "struct Types.Product",
"name": "",
"type": "tuple"

```

```

    },
    {
      "components": [
        {
          "components": [
            {
              "internalType": "address",
              "name": "id_",
              "type": "address"
            },
            {
              "internalType": "uint256",
              "name": "date",
              "type": "uint256"
            }
          ],
          "internalType": "struct Types.UserHistory",
          "name": "manufacturer",
          "type": "tuple"
        },
        {
          "components": [
            {
              "internalType": "address",
              "name": "id_",
              "type": "address"
            },
            {
              "internalType": "uint256",
              "name": "date",
              "type": "uint256"
            }
          ],
          "internalType": "struct Types.UserHistory",
          "name": "supplier",
          "type": "tuple"
        },
        {
          "components": [
            {
              "internalType": "address",
              "name": "id_",
              "type": "address"
            },
            {
              "internalType": "uint256",
              "name": "date",

```

```

        "type": "uint256"
    }
],
"internalType": "struct Types.UserHistory",
"name": "vendor",
"type": "tuple"
},
{
    "components": [
        {
            "internalType": "address",
            "name": "id_",
            "type": "address"
        },
        {
            "internalType": "uint256",
            "name": "date",
            "type": "uint256"
        }
    ],
    "internalType": "struct Types.UserHistory[]",
    "name": "customers",
    "type": "tuple[]"
}
],
"internalType": "struct Types.ProductHistory",
"name": "",
"type": "tuple"
}
],
"stateMutability": "view",
"type": "function"
},
{
    "inputs": [
        {
            "internalType": "address",
            "name": "id_",
            "type": "address"
        }
    ],
    "name": "getUserDetails",
    "outputs": [
        {
            "components": [
                {
                    "internalType": "enum Types.UserRole",

```

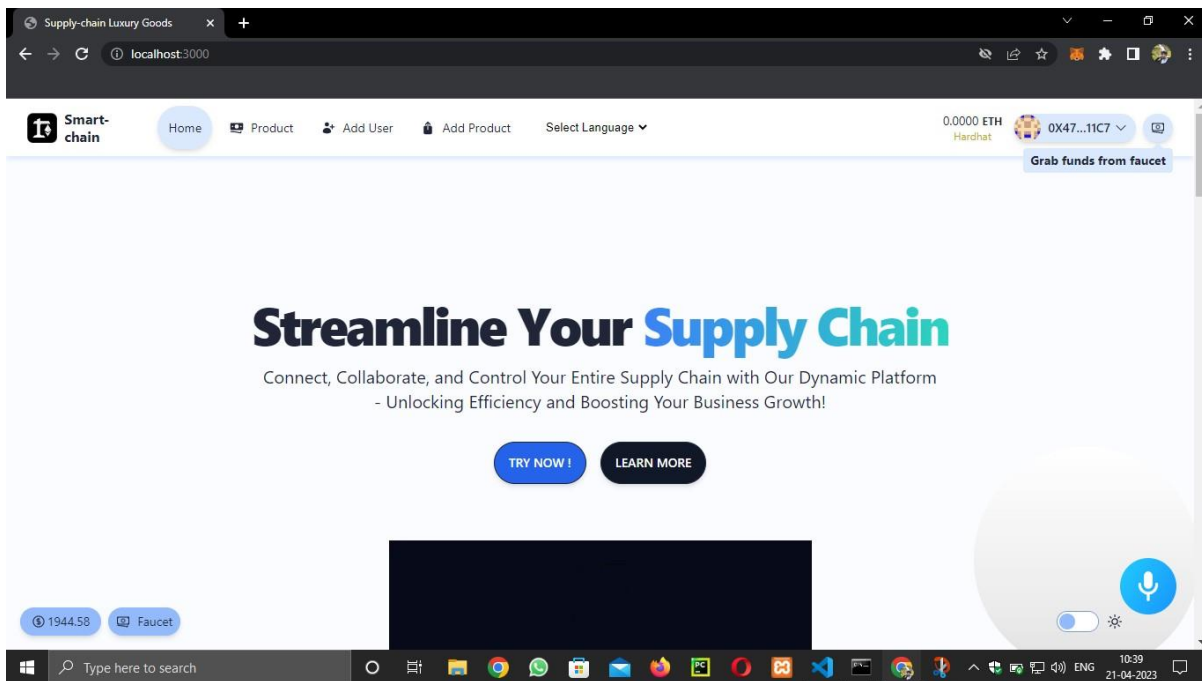
```

        "name": "role",
        "type": "uint8"
    },
    {
        "internalType": "address",
        "name": "id_",
        "type": "address"
    },
    {
        "internalType": "string",
        "name": "name",
        "type": "string"
    },
    {
        "internalType": "string",
        "name": "email",
        "type": "string"
    }
],
"internalType": "struct Types.UserDetails",
"name": "",
"type": "tuple"
}
],
"stateMutability": "view",
"type": "function"
},
{
    "inputs": [
        {
            "internalType": "address",
            "name": "partyId_",
            "type": "address"
        },
        {
            "internalType": "string",
            "name": "barcodeId_",
            "type": "string"
        },
        {
            "internalType": "uint256",
            "name": "currentTime_",
            "type": "uint256"
        }
    ],
    "name": "sellProduct",
    "outputs": [],

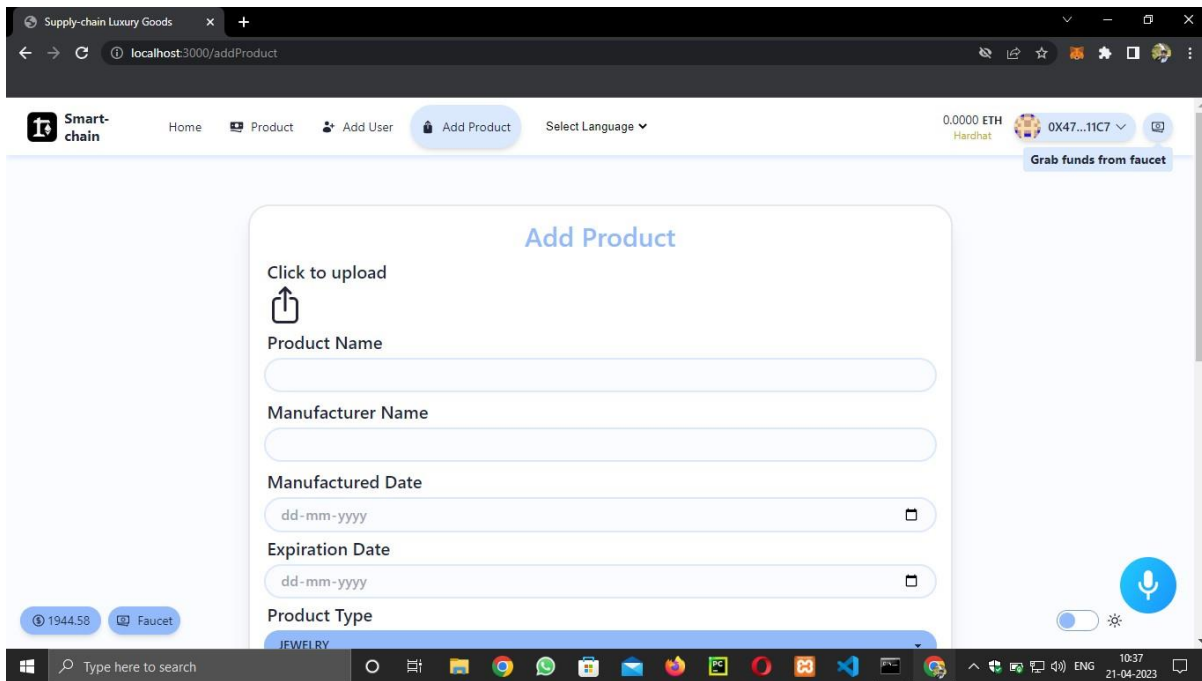
```

```
    "stateMutability": "nonpayable",  
    "type": "function"  
  }  
],
```

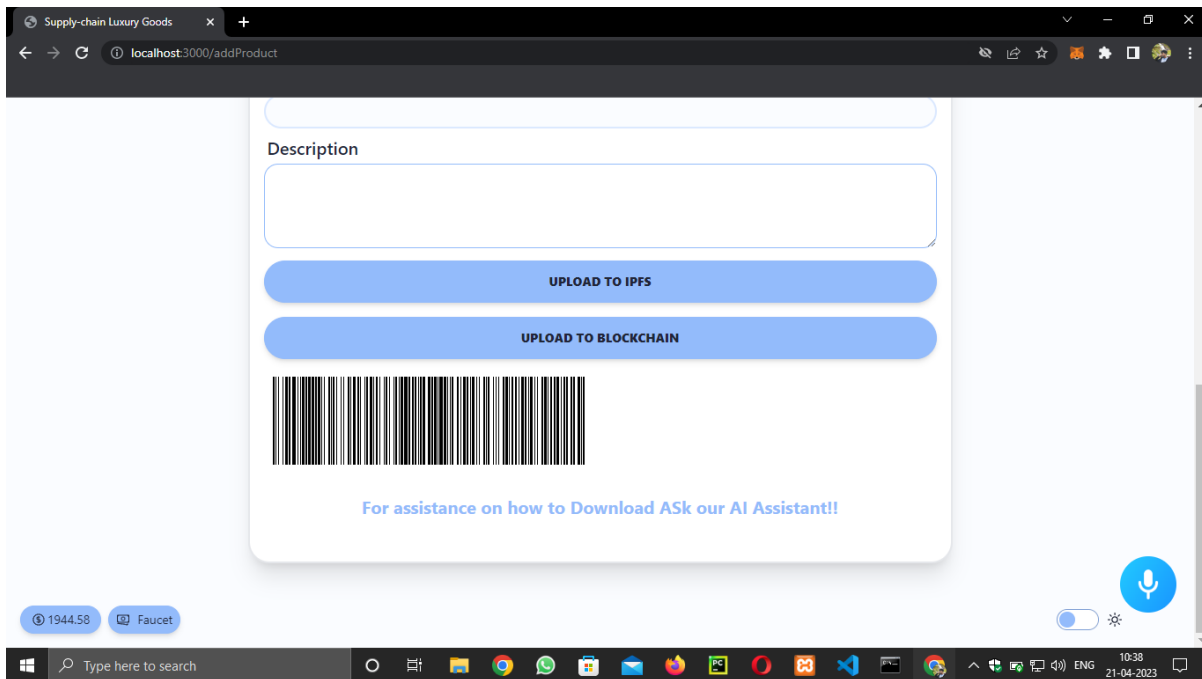
## B. SCREENSHOTS



Home Page

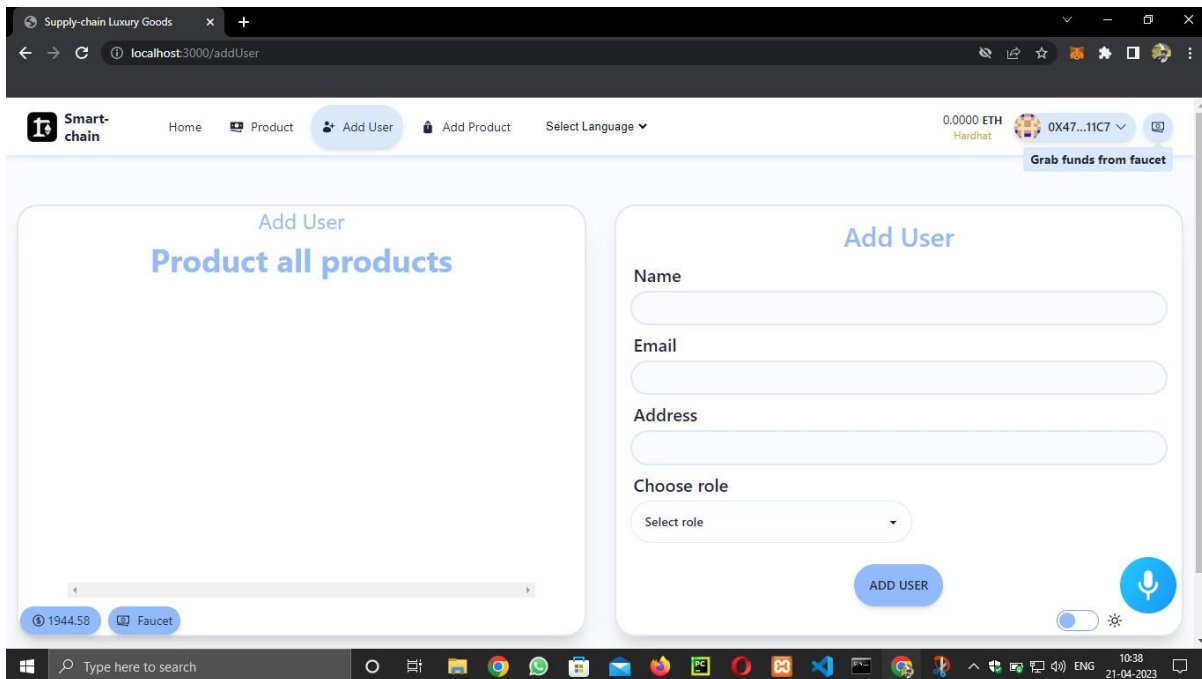


## Product Page

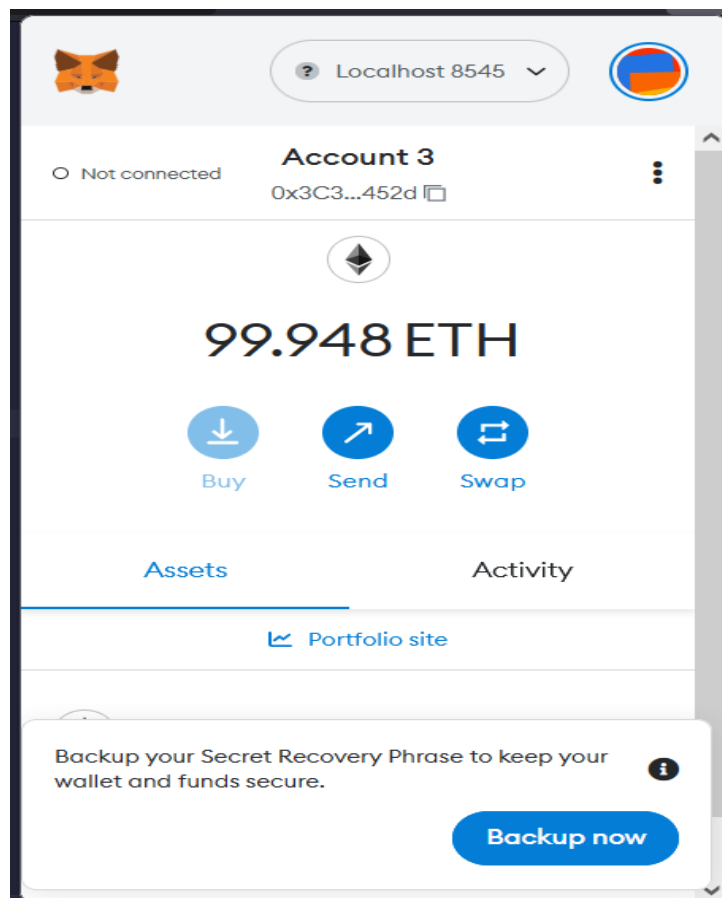


## QR code generation Page

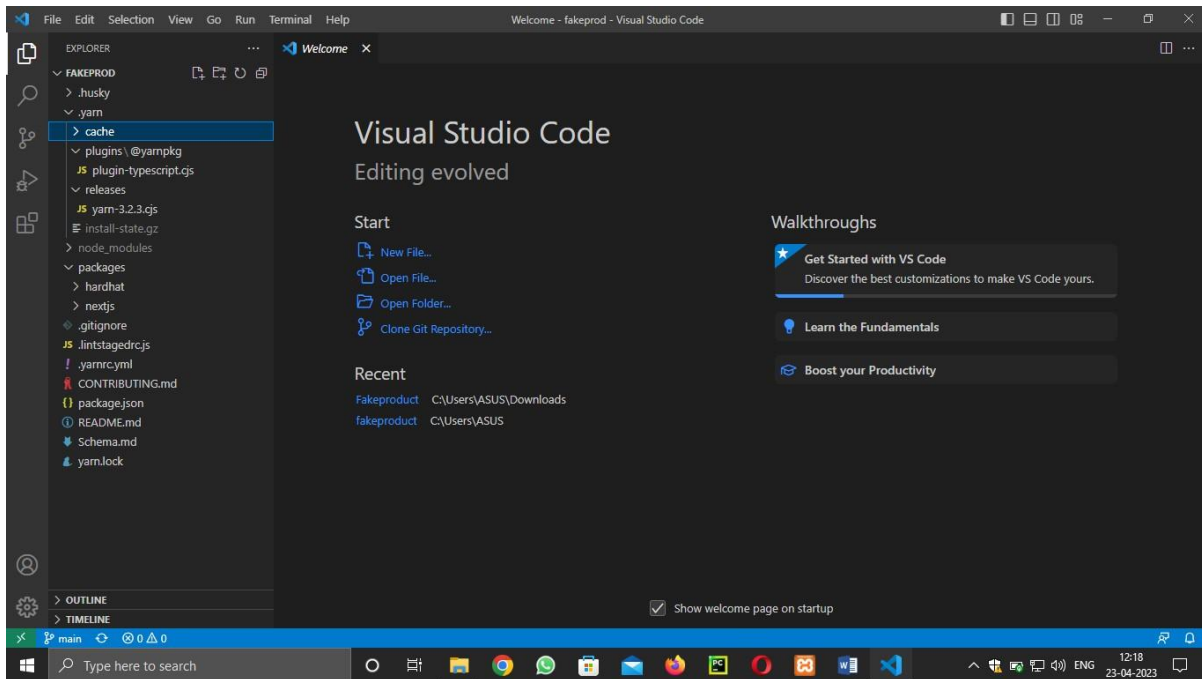




## Add User Page



## MetaMask UI



## Source Code

## C. RESEARCH

### *Product Identification System Using Blockchain*

*Rohith M<sup>1</sup>*

*Department of Computer Science  
and Engineering,  
Sathyabama Institute of Science and  
Technology,  
Chennai, India.*

[rohithsah262@gmail.com](mailto:rohithsah262@gmail.com)

*Srikanth B<sup>2</sup>*

*Department of Computer Science  
and Engineering,  
Sathyabama Institute of Science and  
Technology,  
Chennai, India.*

[srikanth.b9111@gmail.com](mailto:srikanth.b9111@gmail.com)

*Gracelin sheena.B<sup>3</sup>*

*Department of Computer Science and  
Engineering,  
Sathyabama Institute of Science and  
Technology,  
Chennai, India.*

[gracelinsheena@sathyabama.ac.in](mailto:gracelinsheena@sathyabama.ac.in)

**ABSTRACT**— Blockchain's various possible applications have garnered more attention in recent years. By removing the possibility of duplicate spending and independently verifying the accuracy of transaction records, Blockchain applications like Bitcoin have gained widespread attention. If a program's core architecture is built on Blockchain technology, then the data it stores will be completely secure at all times. This piece employs Blockchain, a distributed ledger technology, to show that buyers can verify an item's authenticity independently of the seller. We describe a decentralized Blockchain system that incorporates counterfeit-prevention measures for goods; this enables manufacturers to disperse genuine products without supervising separately-owned retailers, which may dilute the effectiveness of quality control methods.

**Keywords**—*Product identification, barcode, QR code, RFID, Image recognition, Machine learning, Inventory management, Supply chain management, Data collection.*

## I. INTRODUCTION

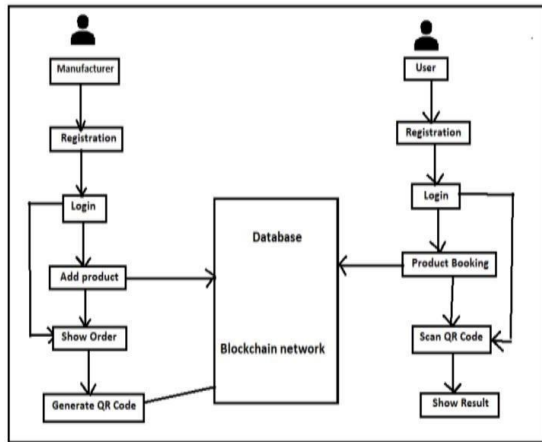
The advent of globalization and the fast development of technology have allowed for an unprecedented increase in the production and distribution of fake goods. The counterfeiting of items ranging from food and medicine to luxury consumer goods has companies searching for solutions to improve supply chain visibility. In the previous decade, blockchain technology has become more popular and has found several applications. Blockchain is an immutable public ledger that is spread over a network. With its help, an organization's assets may be tracked throughout a network and traded with little effort. Using Blockchain as the underlying technology for an app ensures that the information stored inside it is immutable. In today's technologically evolved world, counterfeiting and duplication are risk concerns linked with the global development of a product or technology that may have detrimental repercussions on a company's reputation, revenue, and the health of its consumers. Blockchain systems are in great demand now. It has received a lot of attention during the last several years. The financial industry has been an early adopter of blockchain. There is no trusted third party in a blockchain system. Blockchain technology is gaining popularity because it can be easily adapted to serve a variety of purposes. Its primary advantages are the speed and transparency of its transactions. It follows a common pattern for keeping track of monetary dealings. So, once its data has been recorded on the blockchain, it cannot be changed. Before making a purchase, the customer

pays significant thought to the product. The products he is going to purchase make him suspicious of their legitimacy. Making ensuring consumers have access to reliable data is crucial if we are to stop the fabrication of fake products. That's why blockchain technology has made great strides in ending the trade in fake and potentially harmful items. The supply chain of a market may be tracked using this approach. The purpose of this work is to guarantee that the products people buy are genuine.

A comparison is drawn between the blockchain and the traditional supply chain. As the company selling the service or product has ownership of the data, it may alter the information as it sees fit. The high value of the originals that are attempted to be copied motivates the production of counterfeits. Blockchain's decentralized database records every transaction influencing the product's data value, in contrast to the centralized network supplied by traditional supply chains. As blockchain is a distributed ledger, any participant in the network may verify the accuracy of the ledger at any time. By this method, companies may provide buyers with genuine products.

So, the product's market worth will increase as customer trust in it is maintained. Three essential components make up a blockchain block: data, a hash, and the hash of the previous block. The data contains the relevant information, whereas the hash contains the unique identifier. It is theoretically impossible to alter the information included in a single block without also altering the information contained in the majority of the network's nodes. A new hash will be generated whenever an attempt is made to modify a block's data.

The immutability of blockchain data ensures that the purchaser always receives up-to-date information about the item being purchased. This is a major improvement over the traditional top-down approach. In this effort, a decentralized Blockchain technology approach is used to guarantee that customers don't have to rely only on vendors to decide whether or not products are genuine. We describe a decentralized Blockchain system that incorporates counterfeit-prevention measures for goods; this enables manufacturers to disperse genuine products without supervising separately owned retailers, which may dilute the effectiveness of quality control methods. System is maintaining Status of product i.e., Manufacturer of product, current owner of product, and history of owners, time stamp i.e., at what time product was updated and a QR code.



#### Stage 1: Product Enrollment Process:

Initially manufacturer will be the first owner of product.

So, manufacturer will request administrator to add product on the network, at that time QR code will be generated. Administrator will enroll product and manufacturer on the network, and QR code is taken.

#### Stage 2: Ship Product to Distributor:

In the next step manufacturer will ship the product to distributor. When distributor receives product will scan the QR code and update his details on the network, about product ownership, time Stamp and date.

#### Stage 3: Ship Product to Retailer:

At this Stage, the retailer receives product from Distributor and scan QR code assigned to product using QR code scanner, and will update owner details of the product on the network.

#### Stage 4: End User Authentication Process:

At the end of the chain, customer will take the product, go to website and upload QR code over there, and customer will able to get all detail about product from manufacture to last retailer. And after getting details, it his question where to buy the product or not.

## II. METHODOLOGY

The suggested blockchain-based product identification system is predicated on a distributed and immutable ledger that allows for secure and transparent data sharing. The system's immutable record of a product's whole lifecycle, from manufacturing to distribution and sale, would increase supply chain transparency, accountability, and traceability. The suggested system would allow for easier product identification and tracking by using a combination of barcodes, fast response codes, and radio frequency identification tags. The obtained data would be securely stored on the

blockchain, making it difficult to tamper with or delete the data. Machine learning algorithms will evaluate data from sensors and cameras to improve supply chain efficiency by detecting anomalies and forecasting trends. The proposed blockchain-based product identification system would promote transparency and openness throughout the supply chain, allowing manufacturers, wholesalers, retailers, and consumers to track products with pinpoint precision from inception to disposal

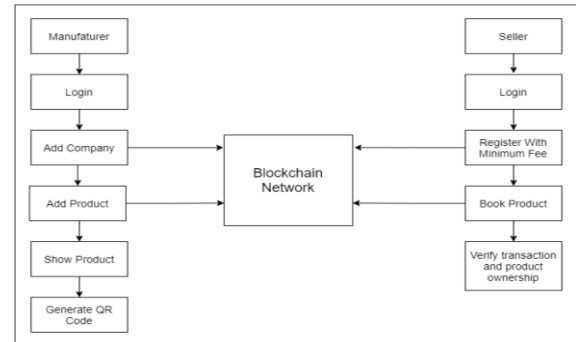


Figure 1: Architecture Diagram

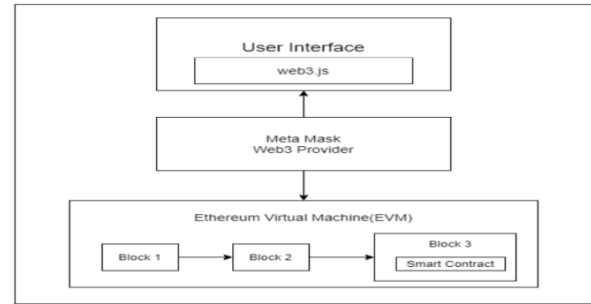
## MODULE 1: DATA COLLECTION AND PREPROCESSING

An important part of the blockchain-based product prediction system is the data gathering and preprocessing module. The data collection, cleaning, and transformation module prepares raw data for further analysis and forecasting. This module saves information gathered from numerous sensors, cameras, and RFID tags in one convenient area. Manufacturing details, component lists, shipping details, and even customer feedback might all be part of the data you gather. Blockchain technology is used in this component to safeguard data integrity and prevent unauthorized access, modification, or deletion. The immutability and transparency of the blockchain network's data storage makes it simple to conduct audits and checks on the information stored there. The acquired data is subsequently cleaned by the module to get rid of any mistakes, inconsistencies, or missing numbers that might throw off the predictions. Errors and blanks are fixed or filled in once the data has been verified for correctness and completeness. Integrating the data involves merging separate datasets from numerous sources to guarantee data uniformity and precision. When data is integrated, it is stored in one place and made readily accessible to all of the product prediction system's components. The data is subsequently preprocessed by the module to get it ready for analysis and

forecasting. Feature extraction, data scaling, and normalization are all possibilities. This part of the product prediction system generates consistent, clean data that the other parts of the system may then analyze and make predictions on. By supplying clean, consistent, and preprocessed data for analysis and prediction, the data collecting and preprocessing module plays a crucial role in assuring the accuracy and dependability of the product prediction system. With blockchain technology in place, data gathering and storage are both safe and transparent, increasing confidence in the information gathered.

## 2: MODEL TRAINING

An important part of the blockchain-based product prediction system is the model training component. The data gathered and stored in the blockchain network is used by this component to construct and train machine learning models. Machine learning methods including decision trees, random forests, support vector machines, neural networks, and deep learning are used in the model training module to build predictive models of goods' actions and outcomes. The module starts by picking an algorithm that works best with the given issue and data. This preprocessed data is then used to train the chosen algorithm on the blockchain. This section uses methods like cross-validation and hyperparameter tweaking to fine-tune the model's efficiency and precision. Performance parameters including accuracy, precision, recall, and F1-score are then used by the module to assess the trained model's efficacy. The overall performance of the model and its appropriateness for the product prediction system are determined based on the evaluation findings. When the model has been developed and tested, it is uploaded to the blockchain and made accessible to the rest of the product prediction system's components. Blockchain technology keeps the model safe from being altered or lost by making it visible and unchangeable. To guarantee that the machine learning models used in the product prediction system are constantly up-to-date and accurate, the module is regularly updated and refined. The use of blockchain technology guarantees the models' openness, letting users check and audit the models' efficiency and precision. By building trustworthy machine learning models that can forecast product behavior and performance, the model training module is an essential part of the product prediction system as a whole. Using blockchain technology improves the system's robustness, dependability, and security by making the models more secure, transparent, and trustworthy.



## MODULE 3: PREDICTING THE OUTPUT

An important part of the product prediction system built on the blockchain is the prediction module. Module 2 is in charge of making inferences about future product performance and behavior using the machine learning models learned in Module 1. The trained models are put to use in the prediction module, which then makes reliable predictions about the product's performance by pointing out potential problems and providing suggestions for how to enhance its efficacy. The module starts with a new data collection and preparation phase from the blockchain network. After the data has been cleaned and organized, it is fed into the machine learning models that were trained in the prior step so that conclusions can be drawn about the product's performance and behavior. Time-series analysis, anomaly detection, and clustering are only some of the methods used by this module to find patterns in the data and provide reliable predictions about the product's future behavior. The module's forecasts provide useful information for preventing problems and learning more about the product's performance. The system's predictions are always precise because the prediction module is regularly updated and fine-tuned. Users may check and audit the accuracy of the forecasts thanks to the usage of blockchain technology, which makes them safe, transparent, and immutable. Ultimately, the prediction module is crucial to the product prediction system because to its ability to effectively forecast product behavior and performance. The use of blockchain technology improves the system's robustness, reliability, and security by increasing the predictability's security, transparency, and trustworthiness. Businesses may get an advantage in the market because to the system's predictions, which can be utilized to improve product performance, save costs, and increase customer happiness.

## CONCLUSION

In conclusion, the blockchain-based product prediction system has the potential to significantly

impact how enterprises worldwide enhance product performance and behavior. Using blockchain technology, the system guarantees data confidentiality, transparency, and immutability, which increases confidence in the system's forecasts. Businesses may improve product quality, save operational expenses, and increase customer happiness thanks to the system's capacity to gather, process, and analyze data, construct reliable machine learning models, and make correct predictions. Although the product prediction system has come a long way, it still has a ways to go. The incorporation of real-time data streams is a promising area for development since it will enable organizations to make forecasts using the most recent data available. Reinforcement learning and deep reinforcement learning are two examples of advanced algorithms and methods that could be used to enhance the system's machine learning models. Businesses may benefit from the system's predictions and insights even more if the user interface is simplified and made more intuitive. Businesses may find it easier to make use of the system's predictions and insights if natural language processing tools are included into the mix. Blockchain-based product prediction systems give firms a leg up on the competition by helping them fine-tune their goods, save expenses, and boost consumer happiness. Exciting chances for organizations to acquire a competitive edge in the market exist thanks to the system's capacity for future expansion and refinement, providing them with important insights and information to make educated business choices

## ACKNOWLEDGEMENT

I would like to thank Dr. B.Gracelin Sheena for helping me and providing valuable feedback throughout the project.

## REFERENCES

- [1] T. J. Sayyad, "Fake Product Identification Using Blockchain Technology," in *International Journal of Future Generation Communication and Networking*, vol. 14, pp. 780-785, 2021, ISSN: 2233-7857 IJFGCN
- [2] Si Chen, Rui Shi, Ren, Jiaqi Yan, Yani Shi, "A Blockchain-based Supply Chain Quality Management Framework", 14th, IEEE International Conference on e-Business Engineering, 2017.
- [3] Blockchain-Based Fake Product Identification in Supply Chain [www.irjet.net](http://www.irjet.net): Ajay Funde, Pranjali Nahar, Ashwini Khilari.
- [4] Fake News Detection In Social Media using Blockchain: - Shovon Paul, Jubair Joy, Shaila Sarkar.
- [5] A Blockchain-Based Application System for Product Anti-Counterfeiting (IEEE Access): Jinhua Ma, Xin Chen, hung-Min Sun.
- [6] T. Tambe, S. Chitalkar, M. Khurud, M. Varpe, S. Y. Raut, "Fake Product Detection Using Blockchain Technology," in *International Journal of Advance Research, Ideas and INNOVATIONS in Technology*, vol. 7, pp. 314-319, 2021, IJARIE-ISSN(O)-2395-4396
- [7] J. Ma, S. Lin, X. Chen, H. Sun, Y. Chen and H. Wang, "A Blockchain-Based Application System for Product Anti-Counterfeiting," in *IEEE Access*, vol. 8, pp. 77642-77652, 2020, doi: 10.1109/ACCESS.2020.2972026.
- [8] K. Toyoda, P. T. Mathiopoulos, I. Sasase and T. Ohtsuki, "A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain," in *IEEE Access*, vol. 5, pp. 17465-17477, 2017, doi: 10.1109/ACCESS.2017.2720760.
- [9] Y. P. Tsang, K. L. Choy, C. H. Wu, G. T. S. Ho and H. Y. Lam, "Blockchain-Driven IoT for Food Traceability With an Integrated Consensus Mechanism," in *IEEE Access*, vol. 7, pp. 129000-129017, 2019, doi: 10.1109/ACCESS.2019.2940227.
- [10] S. Anandhi, R. Anitha and S. Venkatasamy, "RFID Based Verifiable Ownership Transfer Protocol Using Blockchain Technology," 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2018, pp. 1616-1621, doi: 10.1109/Cybermatics 2018.2018.00270.
- [11] S. Rahmadika, B. J. Kweka, C. N. Z. Latt and K. Rhee, "A Preliminary Approach of Blockchain Technology in Supply Chain System," 2018 IEEE International Conference on Data Mining Workshops (ICDMW), 2018, pp. 156-160, doi: 10.1109/ICDMW.2018.00029.

