Capstone Project

Product Tracking Application on Blockchain

Presented by

Warisa	Thaweekul	6222771758
Sivakorn	Seinglek	6222780460
Pattharadanai	Sanitjairak	6322771724

Presented to

Lecturer: Dr. Watthanasak Jeamwatthanachai

Blockchain Development – CSS486 Section 1

Sirindhorn International Institute of Technology

Thammasat University

22 May 2023

Table of Contents

Table of Figures

Chapter 1

Introduction

Problem Statement

Product traceability is considered as one of the critical features of today’s commerce, especially in the second-hand product category. In general, a customer should be at least informed of a detail of trace of a product or a batch of products such as the history, previously held locations, original manufacturer, previous owners of the product, or original product labels before deciding to purchase it at any cost. Its functionality does not only increase the creditability of a product to a customer but also enables the manufacturers to identify issues with the product.

However, the technological capability of people including customers and non-customers has risen at this moment and tampering with product traceability is inevitable leading to decreasing in product trust. It does not matter whether the traceability interface is on paper-based or digital platforms that the risk of unauthorized modification of product traceability still is very high degree.

Objectives

1. Create a digital-based product tracking platform
2. Create a secured, transparent, and non-tampered product tracking platform
3. Provide customer access to significant product details

Stakeholder Benefits

Transparency and Visibility

Customers can now view the status of their orders through our product tracking system. Customers can track the status of their orders with this level of transparency, which increases customer satisfaction, trust, and operational transparency. We improve the overall customer experience by giving real-time updates on the details.

Accurate and Current Inventory Data

Our product tracking system enables accurate and up-to-date inventory management. Businesses can track products in detail and send the current owner details. Moreover, our project can be used to check stock products which will give accurate information about the stock.

Product Protection against Fraud and Counterfeiting

Our product tracking system helps to strengthen product security against fraud and counterfeiting. Customers can check product information within the system, ensuring authenticity and removing the possibility of buying fake goods. Customers become more trusting of our system as a result of this feature.

Chapter 2

Related Documents and Literatures

Blockchain

Blockchain is a continuously expanding list of records that are kept in a container called a "block” and linked and secured using cryptography as a "chain". Due to its decentralized and immutable nature, which makes it very difficult for unauthorized individuals to alter or manipulate data, it offers enhanced security. Additionally, it encourages transparency because everyone using the blockchain network has access to the same data, eliminating the need for middlemen. Several advantages of blockchain make it valuable across multiple industries. To detail each component:

Block

A block can be simply defined as a container of records or information. It usually includes metadata such as a block number, timestamp, digital signature, hash value, and nonce, depending on the purpose of the blockchain system designer. Each block is linked by including additional metadata called the previous hash value that points to the previous block.

Cryptographic Hashing

Cryptographic Hashing, or hash in short, is a computation technique to produce a distinct code or fingerprint of any input information. The most significant property of a hash is that it will always produce a completely unique code, even if the information has been changed by only a single word or bit. As we have stated before, every block should store its hash value and the previous hash value of the previous block. In cases where a particular block in a chain has been tampered, the block which is linked to that block will automatically be noticed as an invalid block according to the mismatch of the previous hash value and will make the rest of the chain completely unstable. Therefore, cryptographic hashing will provide a challenge to changing any one block’s data without also changing the entire chain, leading to the significant property of a blockchain called "immutability".

Distributed Network

A network of computers (peer to peer) known as nodes is responsible for maintaining the blockchain. A complete copy of the blockchain is kept on file by each node, which consistently updates itself as new blocks are added. This enhances the immutability of a blockchain in the case that some people may sophisticatedly modify an entire chain instead of a block; however, they will be exactly rejected by the other nodes because the blockchain data is mismatched with their copy. Therefore, this architecture ensures that a blockchain is resilient to errors and attacks. Every node in the network will also act as a validator, or "miner" in a cryptocurrency term, that includes a new block in a blockchain that will be performed logically by their consensus. This leads to another significant property of a blockchain called "decentralization".

Consensus Mechanism

Participating nodes must concur on the legitimacy of the information contained in a block before it will be included in the blockchain as a new block. Through a consensus mechanism, which varies depending on the implementation, this ensures that the newly added block will be the one and only version of the truth that's agreed upon by all the nodes in the network.

Proof of Work (PoW)

Participating nodes, commonly known as miners, will compete to use their sacred computational resources to solve a challenging mathematical puzzle. A miner who first solves the puzzle will have the right to validate the new block before adding it to the blockchain and then receive a reward. This method consumes a lot of energy and may result in centralization. Examples of this kind of blockchain are Bitcoin, Dash, Litecoin, and Monero.

Proof of Stake (PoS)

Participating nodes must lock up or "stake" a certain amount of their asset (a commonly used cryptocurrency) to take part in the validation process. The amount staked determines the likelihood of being selected to validate a block and receive a reward. After being elected as a validator, the node will make a contract with the network (a commonly used smart contract) that if the block is later found to be fraudulent or malicious, all their stakes will be forfeited. This approach promotes decentralization and is more energy-efficient than proof of work. Examples of this kind of blockchain are Ethereum, Cardano, Polkadot, and Solana.

Proof of Authority (PoA)

Unlike the proof of work, there is no technical competition between the nodes or miners here. The nodes will be elected to be the validators based on their trust or reputation. It can be considered as a variant of proof of stake in that the nodes stake their "reputation" instead of the asset (cryptocurrency). The process is automated and does not require validators to constantly monitor their computers like PoW and PoS. The proof of authority model relies on a limited number of block validators, which makes it a highly scalable system. Examples of this kind of blockchain are VeChain, Bitgert, and Xodex.

Smart Contract

Smart contracts enable us to implement many programs on a blockchain besides storing data. A contract will be embedded in a block and will be automatically executed when the conditions are met. Smart contracts can improve efficiency, security, and transparency so that parties can carry out agreements directly between themselves. It is an effective tool for reshaping numerous industries and conventional contract management procedures. Smart contracts can be used in multiple industries such that it automates and streamlines processes by executing actions based on predefined conditions automatically.

Decentralized Application

Some blockchain platforms, like VeChain and Ethereum, can support applications built on top of blockchain platforms and smart contracts running across a distributed system called decentralized applications (DApps), which make blockchain able to offer cutting-edge features and services. DApps have also been developed to facilitate secure, blockchain-based voting and governance. DApps can also be integrated into web browsers to function as plugins that can help serve ads, track user behavior, and solicit crypto donations. There are many examples of commercial decentralized applications deployed in various real world use cases.

Food Safety on Walmart China

Blockchain technology is enabling Walmart China to implement a product traceability strategy and pioneer the large-scale application of food product traceability. By scanning the desired products, customers can acquire detailed information, including the source of the scanned products and the geographic location received by Walmart, the logistics process, a product inspection report, and many more data points. Data collection and data availability are to be continually added to the scale of the platform and its use of blockchain technology.

As a retailer trusted by consumers, Walmart has been devoted to improving food safety and quality management, and blockchain technology will enable it to take the lead in traceability management among retail peers. Walmart China has pioneered the large-scale application of a traceability platform based on the Blockchain, giving priority to products of high concern and high risk. The first batch of 23 product lines has been tested and implemented using the public blockchain platform, with more than 100 product lines following in the second half of this year, covering more than 10 categories.

Domestic customers are proposing higher demands on food safety, and the digitalization of food traceability will help suppliers increase their brand value and win their consumers' trust. Walmart China expects the use of the platform to continue to grow in scope, accounted scales, and geographic implementation. As the platform continues to expand, more categories of food will be traceable, and thus the value transfer will be further expanded in the industry as reflected in the Blockchain technology.

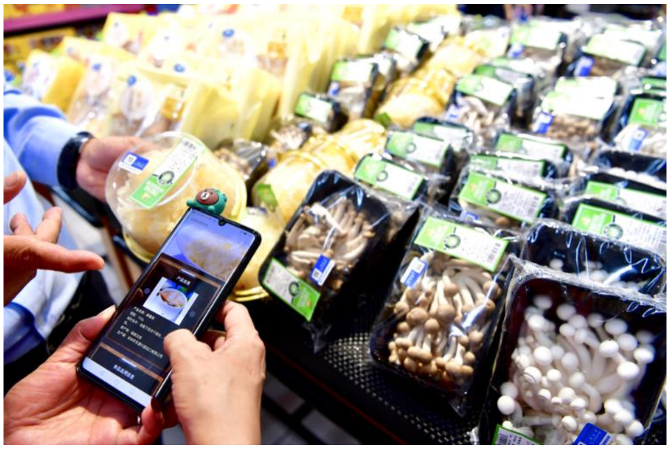
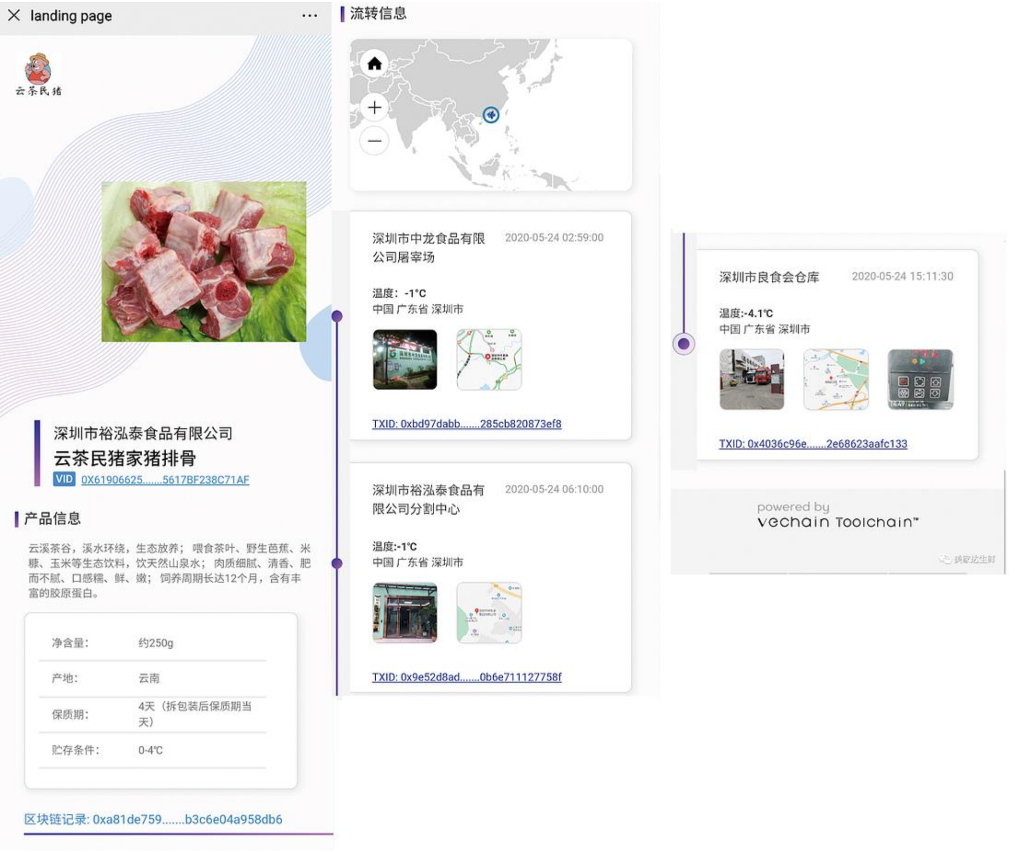


Figure 2.1: Use case of blockchain in Walmart China for food safety

Yuhongtai Foods Traceability Platform for Premium Pork Products

On June 2nd, 2020, Shenzhen Yuhongtai Foods Limited Company announced the adoption of blockchain technology to power their food traceability platform, starting from the premium pork products by their wholly owned subsidiary, Meijiada Fresh Foods. While Meijiada prides itself on providing only the best quality products for its customers, proving that has been difficult. This is especially a problem as Meijiada's business is mainly conducted in Shenzhen. In this advanced technology hub, consumers are increasingly tech-savvy and thus skeptical of information presented on readily packaged meat products, such as the date of packaging. This lack of visibility between Meijiada and its customers negatively affects its positioning as a premium brand.

Meijiada has chosen to adopt blockchain technology so that they are now able to upload key supply chain data of their products, such as origin, processing flow, logistics information, date of packing, as well as marketing information. Shenzhen Yuhongtai Foods Co., Ltd. and Meijiada have revealed that the food traceability technology will be initially used for its Yuncha Minzhu brand of premium pork products. Through a scan of the QR code on the packaged product, customers will be able to learn more about the product's information and, most importantly, can be assured of the product's safety before consumption. More product lines can be expected soon when the current implementation is gaining success.



深圳市裕泓泰食品有限公司
云茶民猪家猪排骨
VID 0X61906625.......5617BF238C71AF

产品信息

云溪茶谷，溪水环绕，生态放养； 喂食茶叶、野生芭蕉、米糠、玉米等生态饮料，饮天然山泉水； 肉质细腻、清香、肥而不腻、口感糯、鲜、嫩； 饲养周期长达12个月，含有丰富的胶原蛋白。

净含量：	约250g
产地：	云南
保质期：	4天（拆包装后保质期当天）
贮存条件：	0-4℃

区块链记录: 0xa81de759.......b3c6e04a958db6

深圳市中龙食品有限公司屠宰场 2020-05-24 02:59:00
温度：-1°C
中国 广东省 深圳市
TXID: 0xbd97dabb.......285cb820873ef8

深圳市裕泓泰食品有限公司分割中心 2020-05-24 06:10:00
温度:-1℃
中国 广东省 深圳市
TXID: 0x9e52d8ad.......0b6e711127758f

深圳市良食会仓库 2020-05-24 15:11:30
温度:-4.1℃
中国 广东省 深圳市
TXID: 0x4036c96e.......2e68623aafc133

powered by
vechain Toolchain™

Figure 2.2: Premium pork product traceability blockchain application

Python

Python is a high-level, object-oriented programming language with dynamic semantics. It is interpreted, which means there is no compilation step, resulting in a fast edit-test-debug cycle. Python's readability and simplicity make it popular for Rapid Application Development and scripting tasks. It supports modular programming through modules and packages, promoting code reuse. Python's interpreter and extensive standard library are freely available for major platforms.

Programmers often appreciate Python for its productivity. Debugging is straightforward as errors raise exceptions instead of causing segmentation faults. The interpreter provides a stack trace when exceptions are not caught. Python's introspective power is evident in its source level debugger, which allows inspection of variables, expression evaluation, setting breakpoints, and step-by-step code execution. Additionally, adding print statements for debugging purposes is a common and effective approach due to the language's quick development cycle. The example of packages that benefit our capstone projects:

Web3

web3.py is a Python library specifically designed for seamless interaction with Ethereum. It is frequently utilized in decentralized applications (DApps) to facilitate tasks like transaction handling, smart contract interaction, block data retrieval, and more. While initially inspired by the Web3.js JavaScript API, web3.py has since evolved to cater to the preferences and convenience of Python developers, offering an array of features and functionalities to support Ethereum-related operations within Python-based projects.

Crypto

Crypto or PyCrypto (Commonly known as PyCryptoDome, the upgraded version of PyCrypto) is a library of collection comprises secure hash functions like SHA256, as well as diverse cryptographic algorithms such as AES, and RSA, and more. The module finds application in various scenarios, including the development of secure administration tools, the creation of daemons and servers, or simple username password authentication module. Additionally, Python's capabilities, such as support for arbitrary-length integers, offer an advantageous framework for prototyping and experimenting with cryptographic algorithms, particularly in the realm of public key cryptography.

Flask

Flask is a lightweight and user-friendly web framework for Python, designed to simplify web application development. It offers a minimalist approach, allowing developers to quickly create routes, define templates, and leverage extensions for additional functionality. With its built-in development server and adherence to the model-view-controller pattern, Flask provides a flexible and customizable framework for building web applications. Supported by an active community, Flask offers an extensive ecosystem of extensions and libraries that can be integrated to enhance its capabilities, enabling developers to create robust and feature-rich web applications efficiently.

HTML and CSS

HTML or Hypertext Markup Language and CSS or Cascading Style Sheets are two of the core technologies for building web page interfaces. While HTML provides the structure of the page, CSS provides visual and aural layout along with graphics and interactions. HTML gives authors the means to:

- Publish online documents with proper structure design including headings, text, tables, lists, photos, etc.
- Retrieve online information such as tracking details, product details via hypertext links, at the click of a button or component in the web page
- Design forms for conducting transactions with remote services, for use in searching information, making reservations, registering to the system, ordering products, sending products, etc.
- Include spreadsheets, video clips, sound clips, and other applications directly in their documents

CSS is a programming language used to define how web pages should look and be presented, encompassing elements like colors, fonts, and layout. It enables customization for various devices, such as screens of different sizes or printers. CSS operates independently from HTML. By separating HTML and CSS, websites become easier to manage, as style sheets can be shared across multiple pages and pages can be adapted to different environments more efficiently.

Ganache

Ganache is a blockchain application designed for Ethereum development and testing purposes. It functions as a local Ethereum network running on a developer's machine, allowing simulation of Ethereum behavior without real network connections or spending actual Ether. Ganache provides predefined test accounts with test Ether, enabling developers to iterate and debug smart contracts and decentralized applications (DApps) efficiently. With features like on-demand block mining, instant transaction confirmation, and gas price control, Ganache offers control over the blockchain environment during development and testing. Its user-friendly interface and developer-friendly APIs make it a popular choice for Ethereum developers to locally develop, unit test, and debug smart contracts and DApps before deployment to the live Ethereum network.

Public Key Infrastructure

Public Key Infrastructure (PKI) is a comprehensive framework utilized to create and administer digital certificates and public-private key pairs. Its purpose is to establish a secure environment for cryptographic operations, guaranteeing confidentiality, integrity, authentication, and non-repudiation in digital communications and transactions. PKI encompasses key generation, distribution, and management, playing a vital role in enhancing the security of online communications and verifying the identities of users and entities involved.

Public Key

In asymmetric cryptography, a public key is an element linked to an individual or entity and is openly distributed. It serves the purpose of encrypting data or validating digital signatures. The public key operates in conjunction with a private key, forming a key pair. When someone intends to send an encrypted message to the owner of a public key, they utilize the public key to encrypt the message, ensuring that only the possessor of the corresponding private key can decipher and retrieve the information.

Private Key

In asymmetric cryptography, the private key serves as the complementary element to the public key. It remains confidential and securely stored by the rightful owner, whether an individual or organization. The private key is employed for decrypting messages that were encrypted using the corresponding public key and for generating digital signatures. Safeguarding the private key is of utmost importance to prevent unauthorized access or disclosure, as it bestows control over cryptographic operations linked to the associated public key.

PKI Certificate

A certificate functions as an electronically signed document that links the identity of an individual, organization, or device to a public key. It is issued by a trusted third party referred to as a Certificate Authority (CA) and contains vital details such as the owner's name, public key, expiration date, and the CA's digital signature. Certificates are instrumental in establishing trust during online communications by verifying the legitimacy of the public key holder. When engaging with a party, the recipient can ensure the sender's identity and integrity by validating the digital certificate provided by the sender, guaranteeing the validity and integrity of the associated public key without any tampering. Certificates play a pivotal role in securing diverse online services, including secure websites (HTTPS), email encryption, and digital signatures.

Selenium

Selenium is a widely used open-source framework that automates web browsers. It offers a suite of tools and libraries for web automation, browser testing, and web scraping tasks. Supporting multiple programming languages such as Python, Selenium enables developers and testers to write scripts for interacting with web elements, simulating user actions, and navigating web pages across various browsers. It integrates with popular testing frameworks and tools, facilitating the creation of comprehensive test suites. Additionally, Selenium's capabilities extend to web scraping, allowing users to extract data from websites. With its versatility and extensive features, Selenium serves as a powerful and flexible solution for efficient web automation, browser testing, and data extraction.

Chapter 3

Design and Implementation

Key Requirements

User Role Separation

We designed to separate the users of the application into 2 groups with different use case which are a plain customer or consumer who is deciding to buy the product according to the product tracking information from the application and an inventory manager who will manage their product in stock such as adding new product to the system, transfer to ownership of product to others when trading contract is satisfied, and etc.

Plain Customer/Consumer

- Enter product ID to see the product tracking information
- May scan QR Code embedded in product package to see the product tracking information
- Make decision to buy the product based on product tracking information

Inventory Manager

- Authenticate themselves before access to the inventory system
- Add new product into the system
- Change ownership of the product in the system
- Secure the inventory information from unauthorized user

Product Tracking Functionality

The product tracking platform contains multiple components related to activities for managing products on the platform. The key requirements are defined to set the scope of this platform when reaching the development process and make the explanations of each component concise. In total, there are four groups of requirements.

Adding Product

This group of requirements shows the user status to run the platform and the instances created. To add the product to the platform, there are two requirements that the prototype must satisfy:

1) The platform should allow any user to create an instance of a product and add it to the platform. It should be done when the user logs in to the platform.

2) The instance of the product created should have a unique ID generated by the platform.

Transferring Product

This group of requirements sets the details of an action that the user can take to transfer the product to another user on the platform. Two requirements are set for the proper transfer action:

1) The platform should allow the user to transfer a product to the other user one piece at a time.

2) To transfer the product, the product owner must log in to the platform, and the receiver must have an account on the platform.

Viewing Product

The platform allows multiple types of users, such as the product's owner, a user who logs in to the platform, or an unregistered user, to view the product's track. However, the information displayed should be different for each type of user. Four requirements are stated to set the content display for each user:

1) The platform should allow the user to view the list of products that the user currently owns while the user must log in to the platform.

2) The product each user currently owns can't be seen by the other users.

3) The platform should allow any user to see the tracking of any product even if the user doesn't log in to the platform.

4) The tracking of data displayed in the platform should include origin, manufacturer, and list of previous owners.

Tracking Output

A requirement on tracking output is given to let the object representing the tracking stick with the physical product. Therefore, the user who owns the product can print out the QR code that includes the information about the product's track.

Immutability and Transparency

Storing the product tracking information into immutable ledger technology such as blockchain will make it feasible to be tampered. With the decentralization property of the blockchain network, the transparency and trust in the product tracking information will significantly increase in the customer before deciding to purchase the product.

Security and Authentication

When the inventory manager manages the stock, they should be authenticated before. Moreover, user's sensitive information such as password, name, surname, phone number, address, or email should be protected by encryption before being stored in blockchain.

User Requirements

Requirement Table

User ID	Role	Requirement
UID001	Plain Customer	How to ensure that the product tracking is trustable?
		I want to see the list of product history before purchasing.
UID002	Inventory Manager	I don’t want unauthorized people to access my inventory.
		I want to transport the ownership of the product easily and efficiently.
		How to ensure that my inventory system has not been tampered by malicious attacker?
UID003	Online Shopping Customer	I would love to scan some QR code and then see all the product tracking.
		I always use my mobile phone to manage everything instead of standalone desktop.
UID004	Inventory Manager	I want to manage my inventory anywhere and anytime even if I do not use the same computers. (Web-based service)
UID005	Plain Customer	How to ensure that the product tracking is real?
		Sometimes, I use my computer to purchase the product, I want to see its track without using external camera to scan something like QR code or barcode.
		How can I get the product tracking If I have only it’s package or container?

Table 3.1: A collection of user requirements

User Journey Map

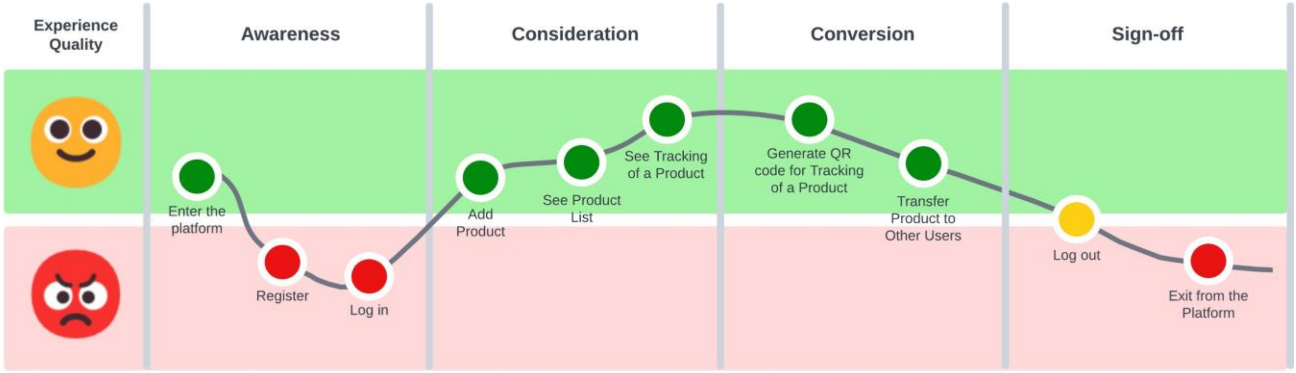


Figure 3.1: User Journey Map

User Value Propositional Canvas

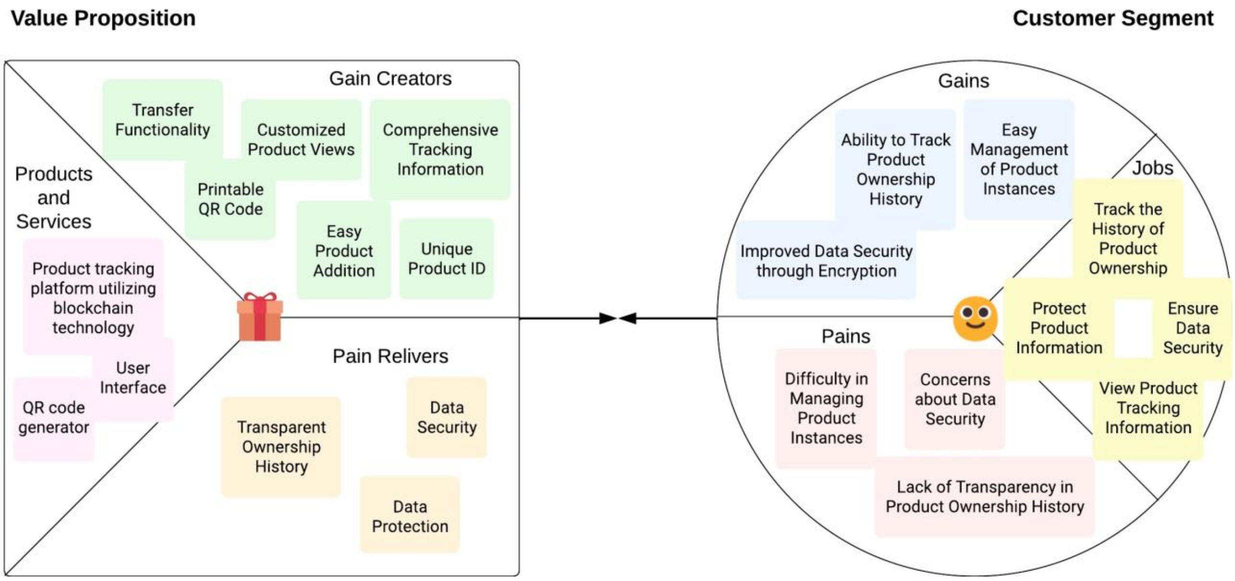

Value Proposition
Customer Segment
Gain Creators
Transfer Functionality
Customized Product Views
Comprehensive Tracking Information
Printable QR Code
Easy Product Addition
Unique Product ID
Products and Services
Product tracking platform utilizing blockchain technology
User Interface
QR code generator
Pain Relivers
Transparent Ownership History
Data Security
Data Protection
Gains
Ability to Track Product Ownership History
Easy Management of Product Instances
Improved Data Security through Encryption
Jobs
Track the History of Product Ownership
Protect Product Information
Ensure Data Security
View Product Tracking Information
Pains
Difficulty in Managing Product Instances
Concerns about Data Security
Lack of Transparency in Product Ownership History

Figure 3.2: User Value Propositional Canvas

System Requirements

Non-functional Requirements

Usability Requirements

1) The system should be responsive to both desktop and mobile user.
2) The system should be a web-based application instead of standalone.

Security Requirements

1) Any information related to the system should not be subject to tamper-based attack so that it should be stored in blockchain.
2) There should be a flexible encryption that can encrypt any size of user personal information before store to the blockchain. First, symmetric cryptography such as AES will be used to encrypt the information, and then asymmetric cryptography such as RSA will be responsible for encryption of symmetric. Moreover, password should be hashed before store in blockchain.
3) A user should be responsible for securing the private key. While a public key will be stored in blockchain in A PKI certificate manner.
4) Any sensitive operation should be confirmed by the user by submitting the password before initiation.

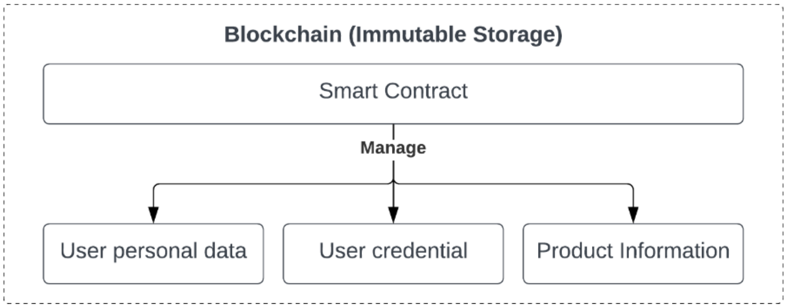


Figure 3.3: Security Mechanism that Enhances Non-tamperable data storage

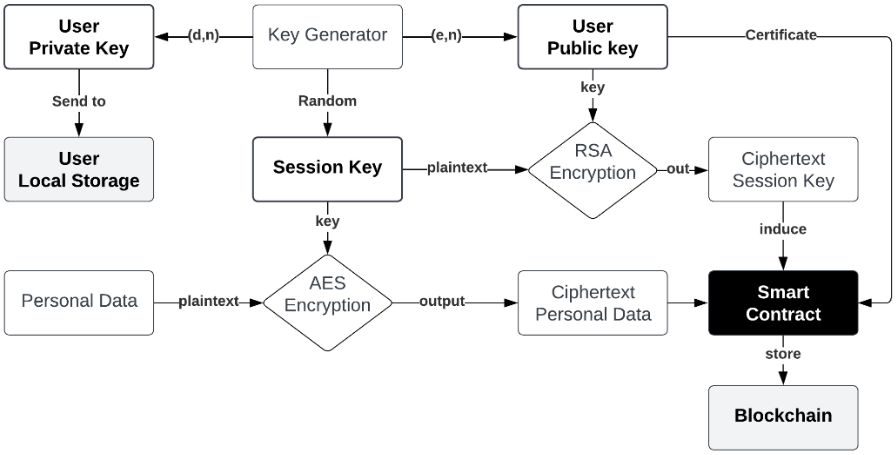


Figure 3.4: Security Mechanism that Enhances confidentiality of the data

Functional Requirements

There should be separated 2 functions for a plain customer/consumer and for an inventory manager. The details of related functionality are explained below:

Retrieval Information Requirements

Plain Customer/Consumer

1) A user will have two choices to get the product tracking information which are scanning QR code embedded on the product package and inputting the product ID to our system. However, both will trigger the same request to Flask.
2) After Flask has received the request, it will induce the smart contract to retrieve the product tracking information matching with the request.
3) Then, Flask will receive the information from blockchain and display to the user.
4) The user will see the tracking of the product and can plan whether they will purchase that product.

Inventory Manager

1) A user will submit the username and password to Flask for authentication process.
2) After Flask has received the request, it will induce the smart contract to access the user's credential which is originally stored in the blockchain after the registration process.
3) Flask will compare if the user's submitted credential and the credential from blockchain is matched or not. If they do not match, it will trigger notification to the user that "username or password is incorrect" so that the user will be able to try again.
4) After the authentication process is finished, Flask will induce the smart contract to retrieve all inventory information along with that user's personal information such as name, surname, telephone number, etc.
5) Then, Flask will render the user's information (ciphertext) and all of inventory information.
6) With private key on the user's local storage, the user client or browser will decrypt the user's information so that the user will see their own personal information as plaintext.

Register and Authentication Requirements

1) A user submits personal information including password, name ,surname, username, etc. to Flask request.
2) Flask will generate key-pairs that the public key will be stored as a certificate via smart contract, private key will be send to the user local storage.
3) Flask will generate symmetric key and use to encrypt the user personal information and will use the public key to encrypt that symmetric key again, and then store all of them in a blockchain.
4) Flask will notify the user that the registration process is complete.

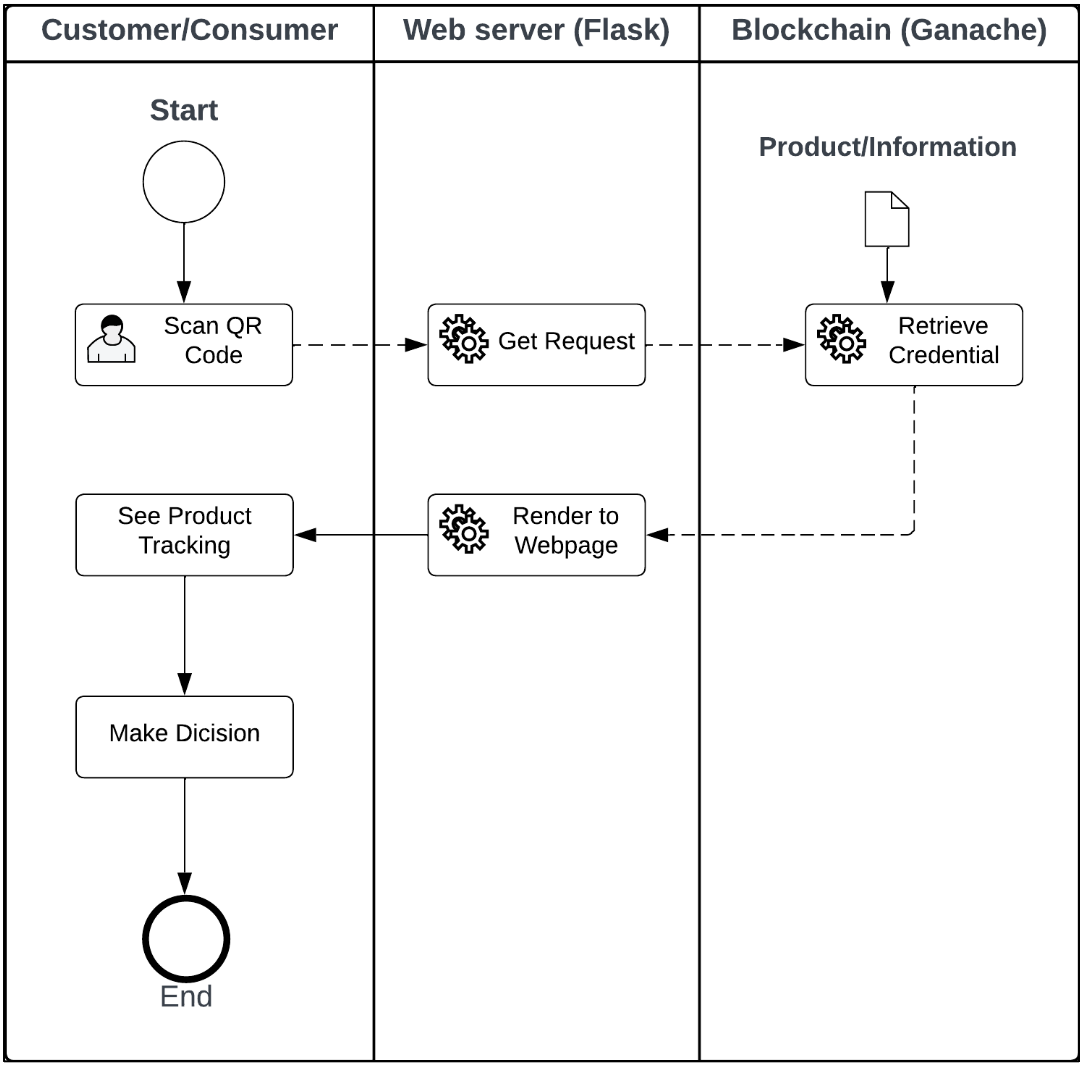


Figure 3.5: BPMN for a plain customer/consumer who want to see the tracking for making decision whether to purchase the product

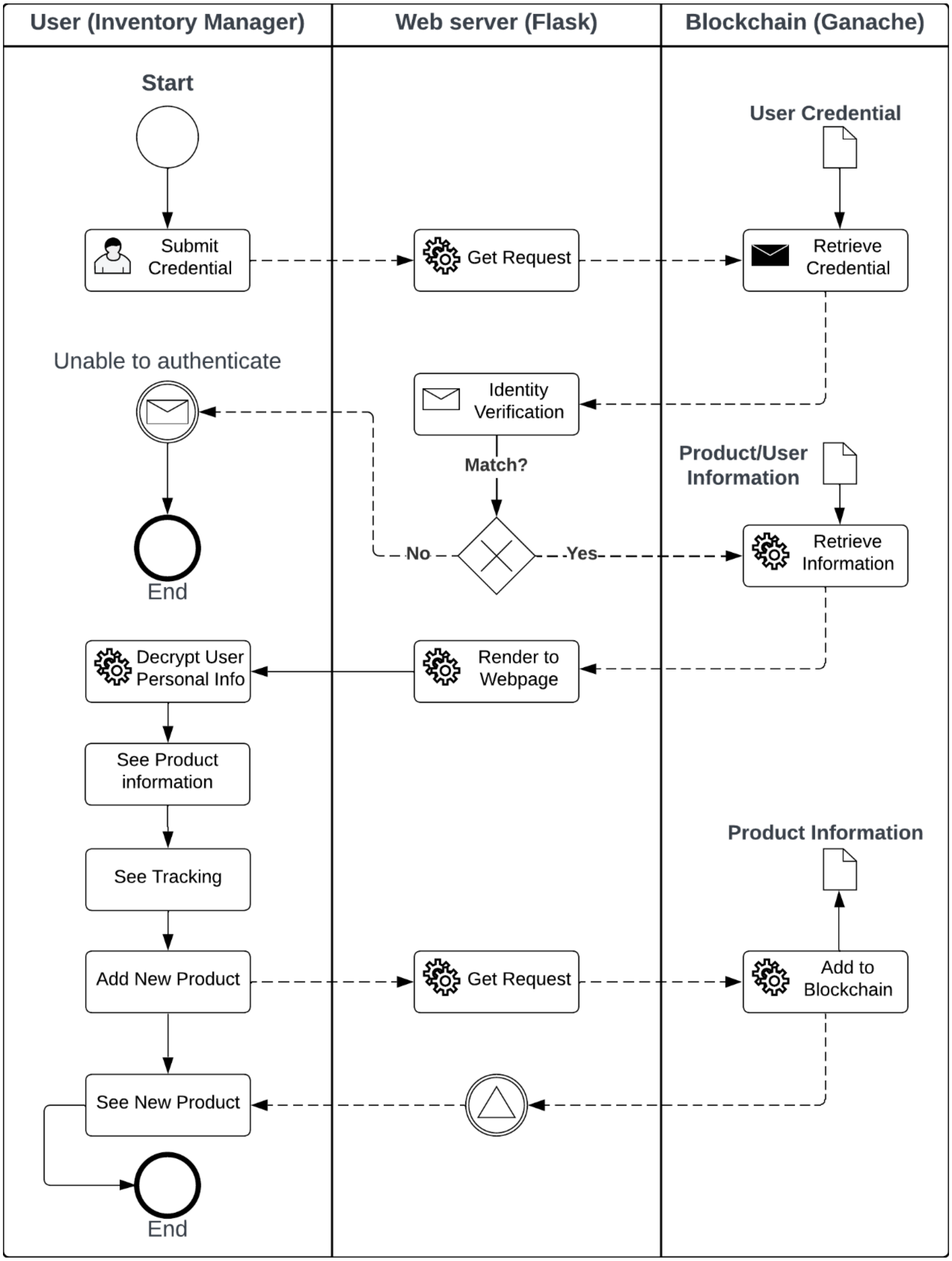


Figure 3.6: BPMN for an inventory manager when they want to manage the inventory

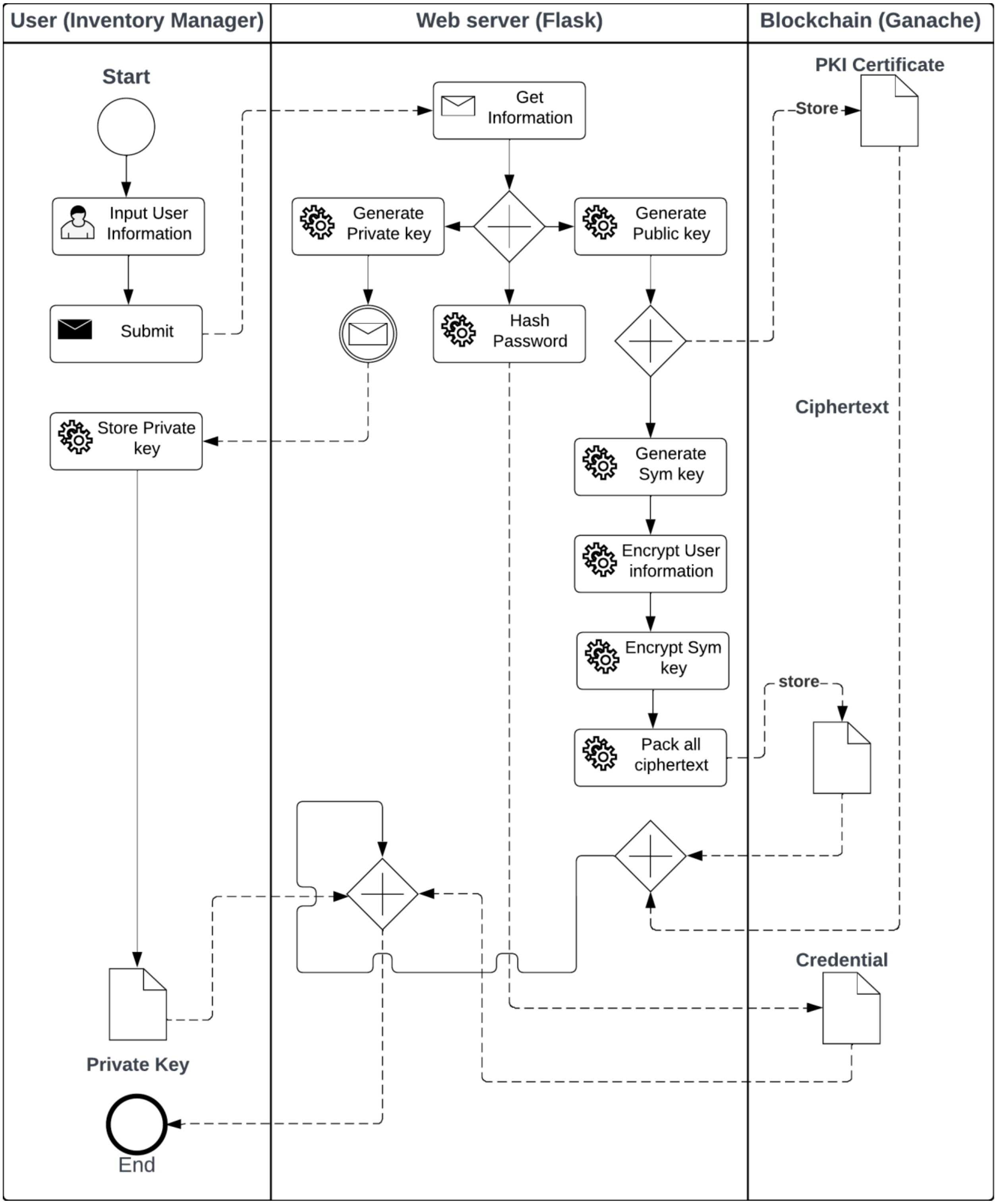


Figure 3.7: BPMN for an inventory manager when they want to register to the system

Use Case Diagram

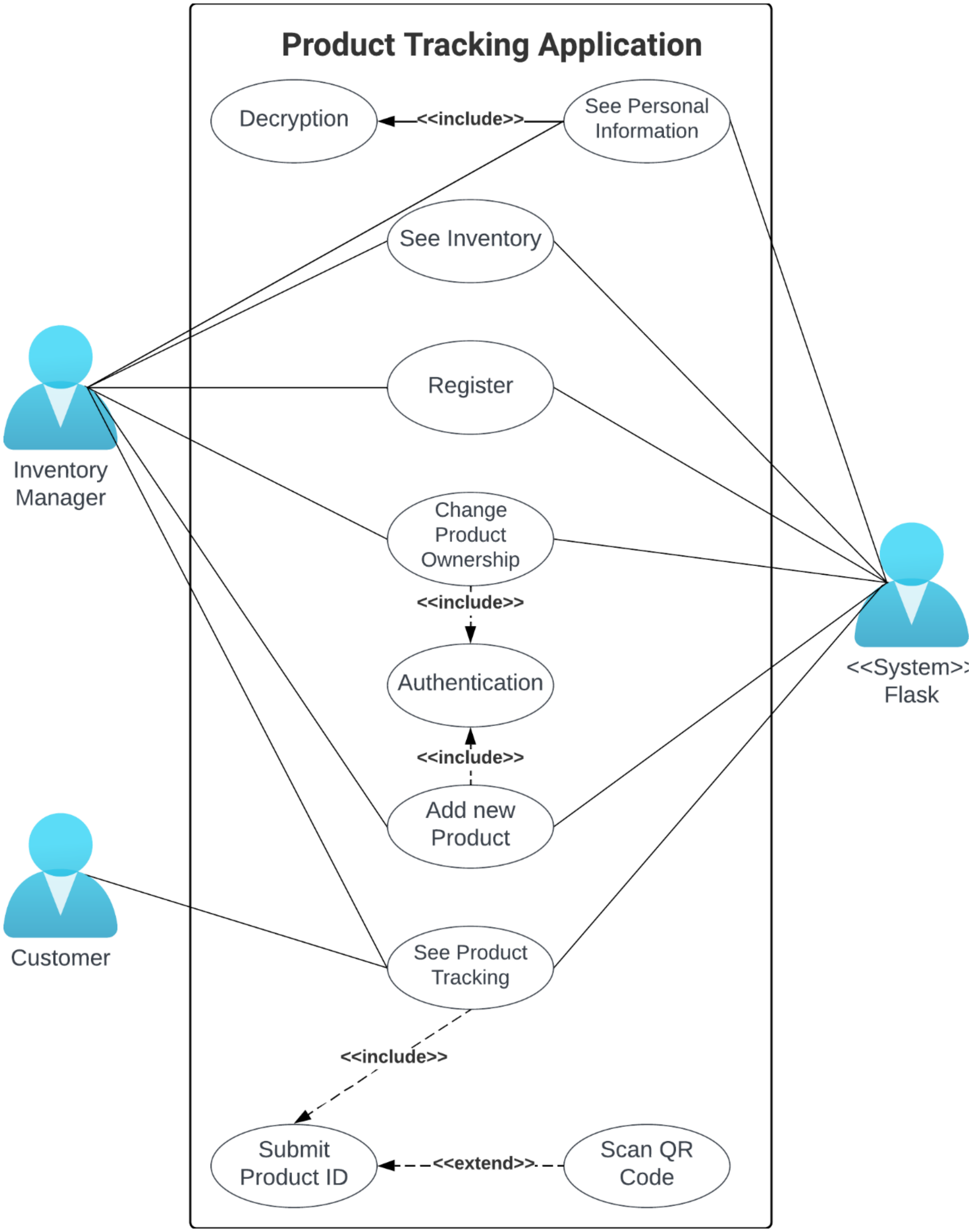


Figure 3.8: Use case diagram of the product tracking application

Blockchain Network Requirements

The blockchain network should be permissionless so that any nodes can connect and disconnect to the system because the product tracking information is visible to any customers. Moreover, they should not charge a gas fee for retrieving the tracking information. In cases of consensus mechanisms, Proof of Authority (PoA) should be implemented because of its speed execution, performance, transparency, and security. In fact, the process of block validation is fast compared to Proof of Work in that it wastes the computation time and resource to solve the mathematical puzzle to validate and add new blocks to the chain. However, the nodes in PoA blockchain should be trustable so that the nodes should come from the partner or cooperative organizations to the developer of product tracking system to increase creditability of each node. In this project, we will simulate the PoA blockchain by configuring Ganache hard fork value to be “Merge”.

System Architecture

Figure 3.5 shows the overview architecture of our product tracking platform. There are three main components that make up the platform. The first component is the local storage, which contains variables for verifying each individual user. The variables stored in local storage include the user’s private key, PKI serial number, and attribute serial number. All variables are kept separately for each individual user.

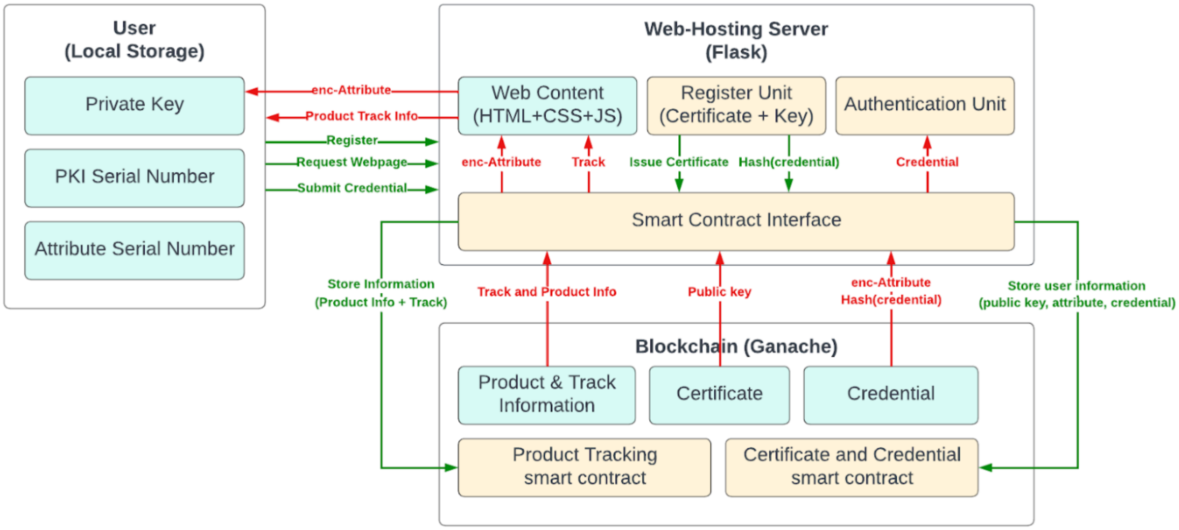


Figure 3.9: The overview architecture of the product tracking application

The second component is Ganache, which is the tool used for running a personal blockchain network as an infrastructure in developed decentralized apps. In this work, Ganache is used for blockchain generation according to smart contract execution. There are two smart contracts created. The detailed contents of both smart contracts will be discussed in the software design section. In short, product and track information, user certificate information, and user credentials are created from the smart contracts. All data is stored as data in a block, and these blocks concatenate as a blockchain in Ganache. The smart contract is written in solidity language.

The last component is Flask, which is the web-hosting server that doesn’t need any external databases or functions from third-party storage libraries. Four main elements are stored in Flask: web content, the register unit, the authentication unit, and the smart contract interface. The smart contract interface is the medium of communication between the web server and the data inside the blockchain, such that it interacts directly with the smart contracts for manipulating information. It is written in Python. Also, the web page processing, including registration and authentication, is implemented in Python, and the data is displayed in the web interface, which is written in HTML (Hypertext Markup Language), CSS (Cascading Style Sheets), and JavaScript. The user’s private key, PKI serial number, and attribute serial number generated from modules in Flask are saved in the user’s local storage.

System Features and Benefits

As the aim of this capstone project is to derive the properties and the various unique features of the blockchain technology to the product tracking application. Thus, product tracking applications are improved with a new level of transparency, trust, immutability, security, and efficiency through the blockchain characteristics, which is renowned for its decentralized and unchangeable nature.

Immutable user credential and information

The user credentials and personal information that the user inserts into the platform, either in the new user registration or the edit of the user’s profile, will be stored in the blockchain. Due to the nature of blockchain, where the data can’t be changed or the blockchain will cause an error, the information stored in the blockchain, including the hash of the user's information, is immutable. On this platform, a smart contract regarding the manipulation of users' information with the storage of the information on the blockchain is implemented. As a result, this platform's user credentials and data are inherently immutable, adding another level of security and trust.

Public Key Infrastructure Implementation

According to the concepts of the Public Key Infrastructure (PKI) illustrated in Chapter 2, the PKI builds up the security, trust, and authentication mechanisms in the platform. It sets up secure communication and data exchange between different components of the platform. With the encryption and decryption of data using different keys that are stored in different places, the information transferred in the system is confidential and unaltered. Furthermore, the PKI certificate originated from trusted certificate identities reduces the risk of fraud and unauthorized access. So, this platform could boost its security, trustworthiness, and protection of data due to the implementation of PKI.

Immutable product tracking information and product metadata information

Similar to the storage of user credentials and information, the product tracking information and product metadata information are immutable since the data related to the product that the user has to enter into the system or update the product's track is stored inside a blockchain with a function in the smart contract. As mentioned previously, blockchain has the feature of an immutable ledger, so the information stored is unchangeable.

User information protection

Apart from the use of features regarding the immutability of data on the blockchain, our platform protects user information by using both symmetric and asymmetric algorithms. In the system design, the user's information is encrypted using the symmetric key, which our team used the AES algorithm in performing this task. The symmetric key is then also encrypted with the recipient's public key with the RSA algorithm, which is an asymmetric algorithm. As the platform uses the PKI, where the public key and private key are generated, the keys are stored in different components. The private key is stored in a user local storage, which contains user's other personal data. The public key and the encrypted attribute session key are stored in the blockchain network. This enhances the protection of user's data.

User Interface and User Experience Design

This project makes use of HTML, CSS, and responsive web design, which enables the website to dynamically adjust its layout and size to suit different devices, including mobile devices and computers. The web pages are designed using HTML and CSS, resulting in an interface that is user-friendly for the website, which is composed of many pages, as the information below shows.

Landing Page

The landing page is when the user clicks on the link, it will show this page in the first order. Users can log in to enter the product page as the homepage.

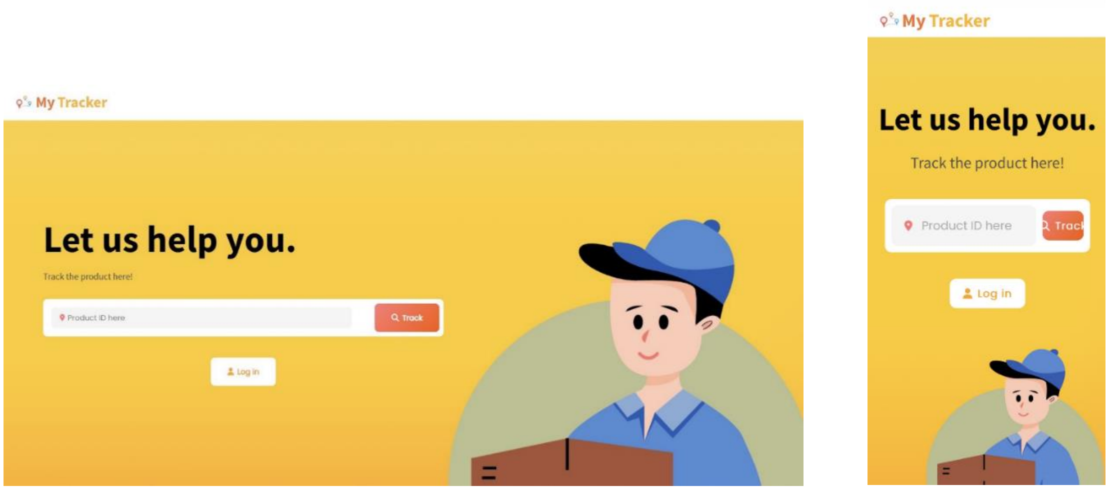


Figure 3.10: Landing or Home Page

Tracking Component Bar

Users can put their ID and click on the track button. It will show details of the product.

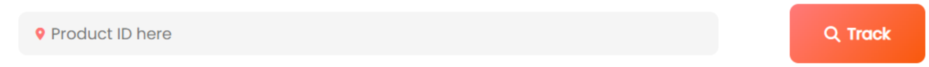


Figure 3.11: Input form that receive product ID

Log-in Pop-up and Form

Users who want to enter the site must log in first.

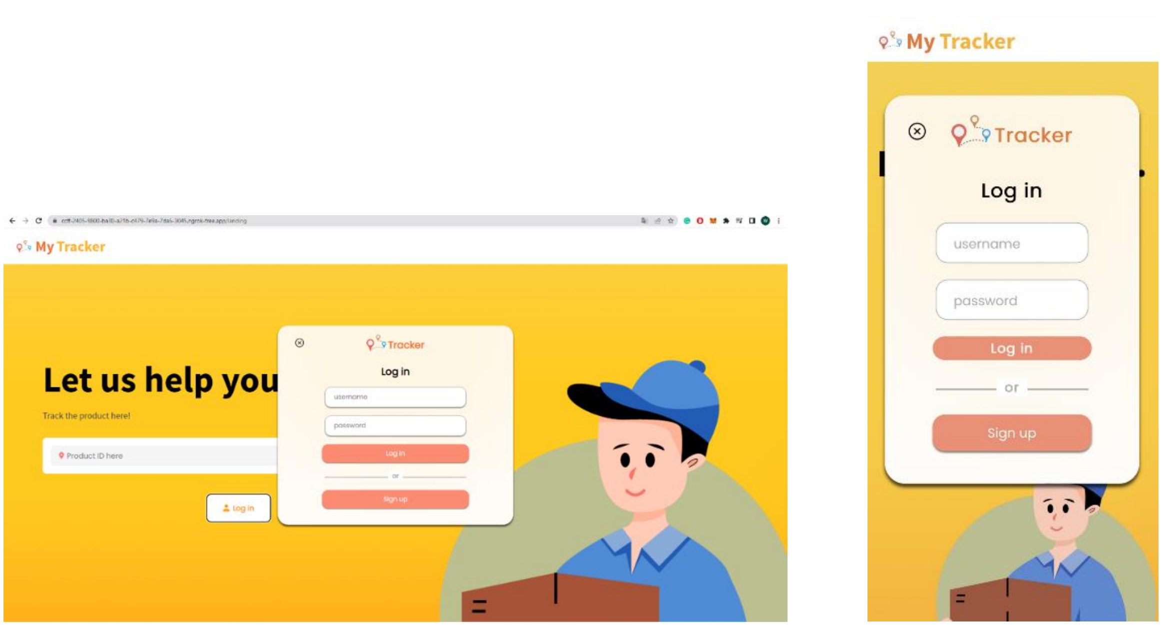


Figure 3.12: Log-in Pop-up and Form

Register Pop-up and Form

If users don’t have an account, they must register one.

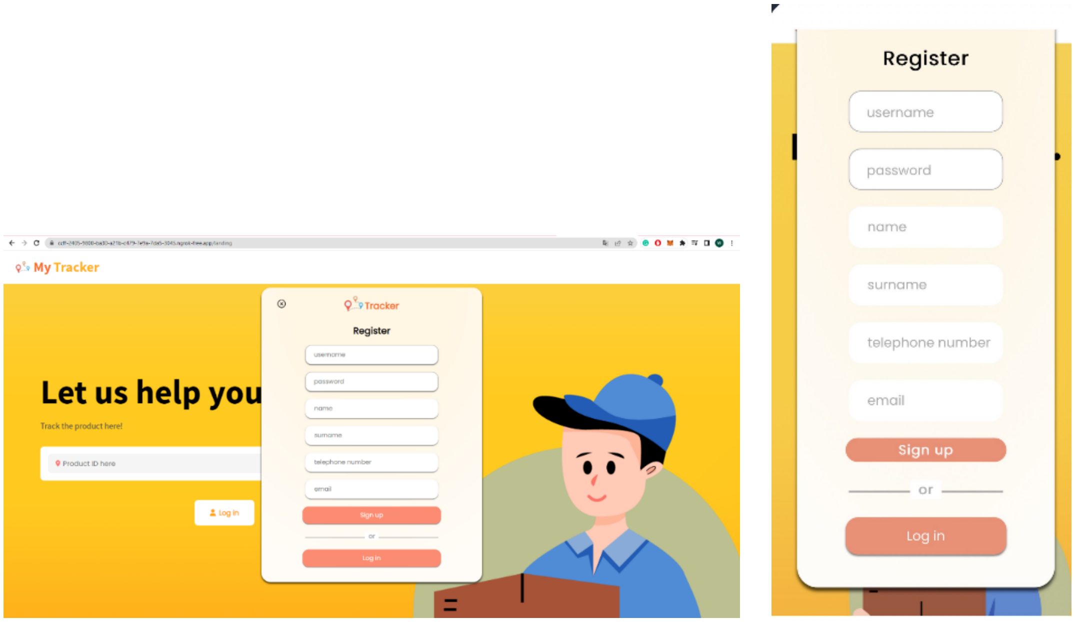


Figure 3.13: Register Pop-up and Form

Product Listing Page

After the user logs in to the website, it will display the user’s own product.

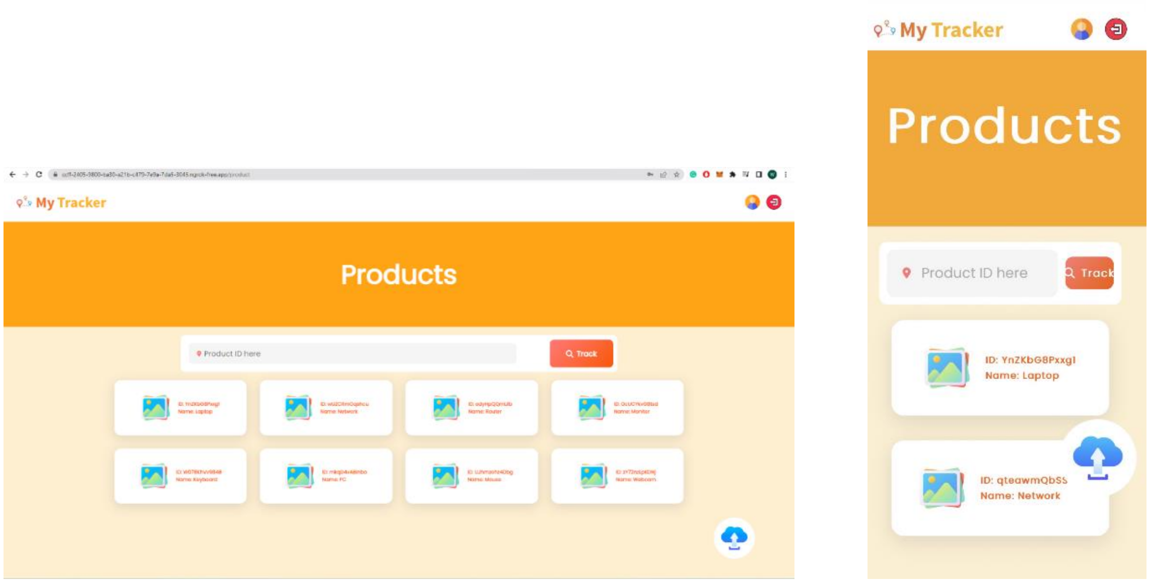


Figure 3.14: Product Listing Page

Add Product Popup and Form

Users can click the icon cloud upload to upload their own products.

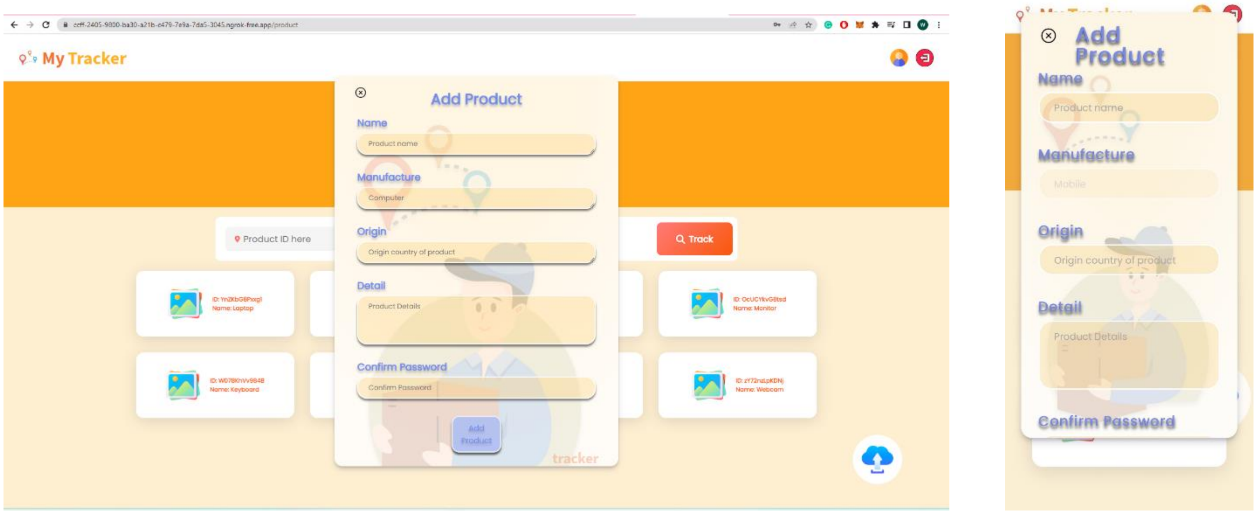
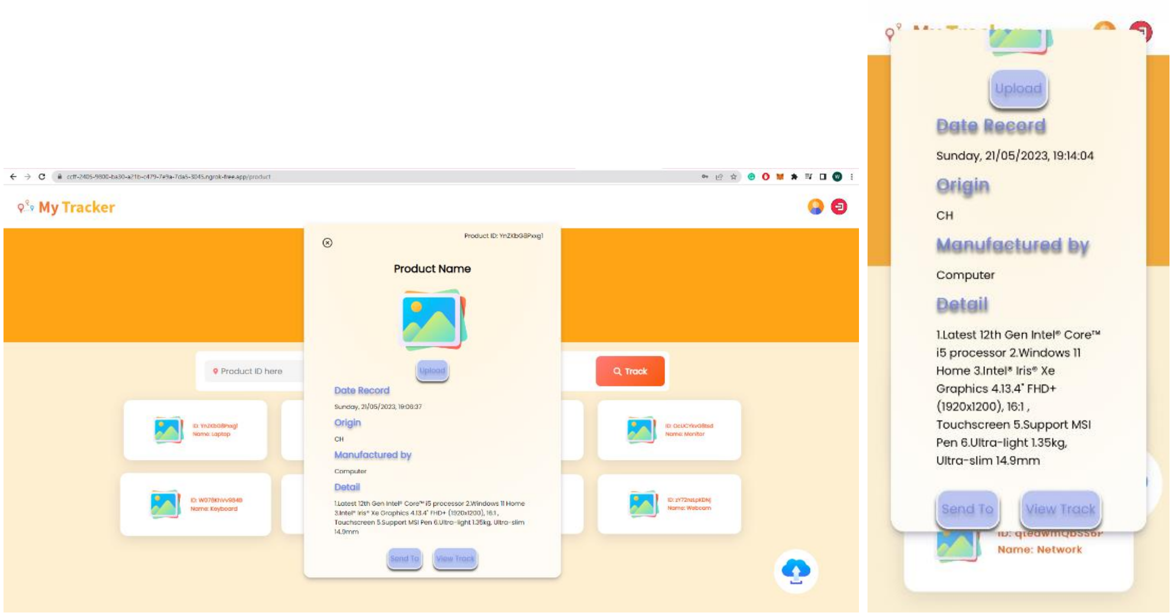


Figure 3.15: Add Product Popup and Form

Product details Pop-up

This popup will show the details of the product including a button to trigger get product tracking record and transfer ownership of the product.



Date Record

Sunday, 21/05/2023, 19:14:04

Origin

CH

Manufactured by

Computer

Detail

1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Figure 3.16: Product details Pop-up

Product Ownership Transfer

After clicking send to, users must fill in the information for the username of a receiver, and product information and confirm the password to send the product

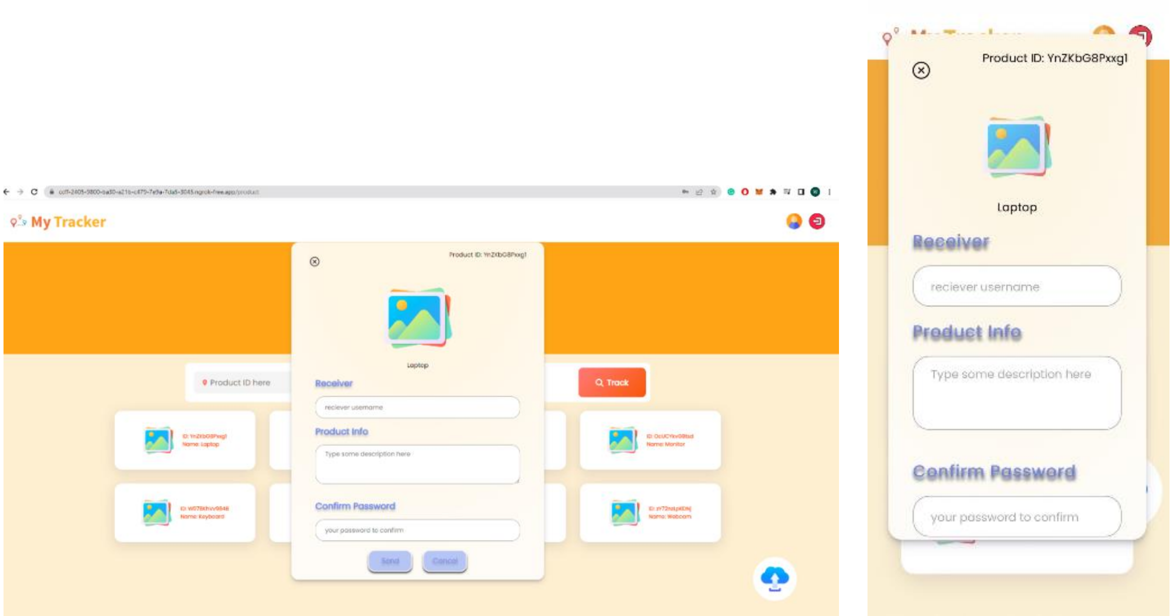
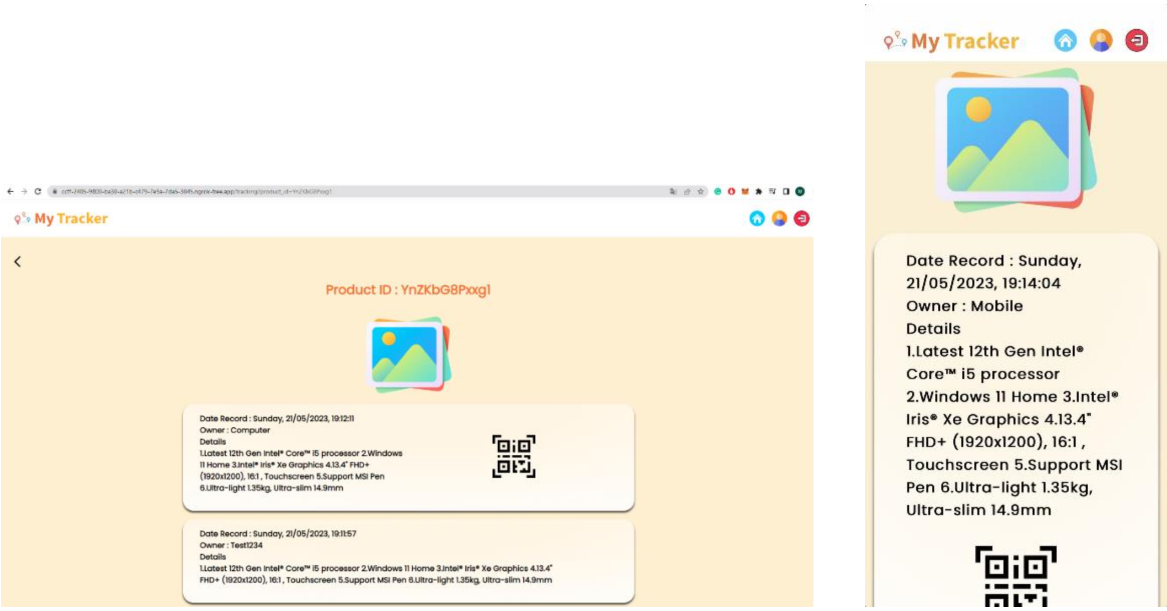


Figure 3.17: Product Ownership Transfer

View Track

After clicking the view track button, it will show the product details and the current owner will be shown in the first list.



Date Record : Sunday, 21/05/2023, 19:12:11
Owner : Computer
Details
1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

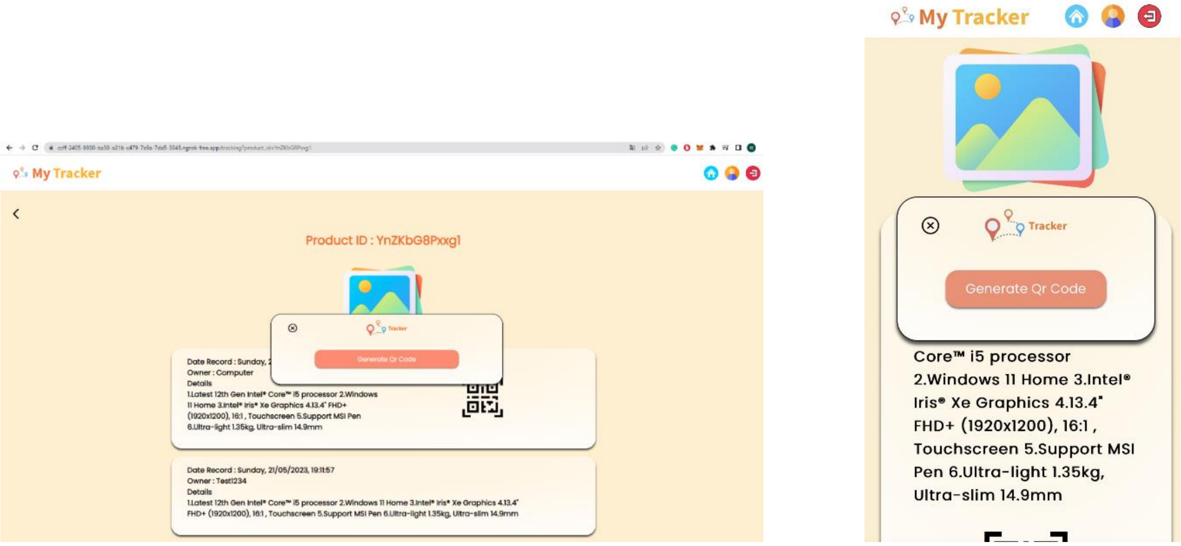
Date Record : Sunday, 21/05/2023, 19:11:57
Owner : Test1234
Details
1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Date Record : Sunday, 21/05/2023, 19:14:04
Owner : Mobile
Details
1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Figure 3.18: View Track

Generate a QR code

When the user clicks on the QR code in the details of the product, it will show the generated QR code button, and after clicking on it, it will generate a QR code for this product that the user can scan to see details of the product.



Product ID : YnZKbG8Pxxg1

Date Record : Sunday, 2
Owner : Computer
Details
1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

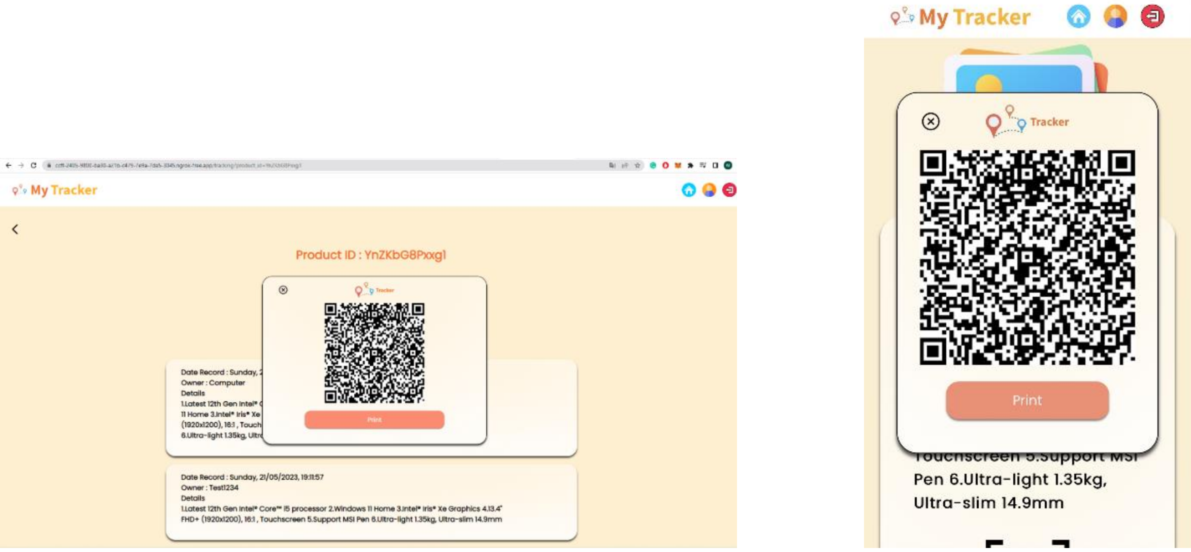
Date Record : Sunday, 21/05/2023, 19:11:57
Owner : Test1234
Details
1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Figure 3.19: Generate QR code

QR code

After clicking on generate QR code, the system will display the QR code which is bound to the product.



Product ID : YnZKbG8Pxxg1

Date Record : Sunday, 2
Owner : Computer
Details
1.Latest 12th Gen Intel® C
11 Home 3.Intel® Iris® Xe
(1920x1200), 16:1 , Touch
6.Ultra-light 1.35kg, Ultr

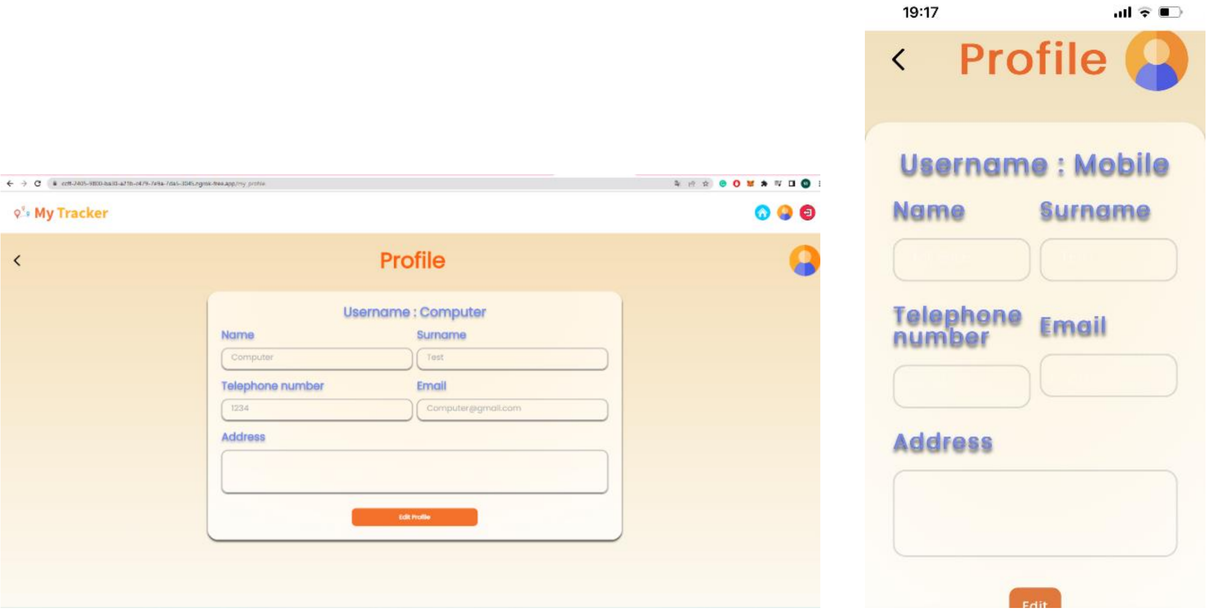
Date Record : Sunday, 21/05/2023, 19:11:57
Owner : Test1234
Details
1.Latest 12th Gen Intel® Core™ i5 processor 2.Windows 11 Home 3.Intel® Iris® Xe Graphics 4.13.4" FHD+ (1920x1200), 16:1 , Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Touchscreen 5.Support MSI Pen 6.Ultra-light 1.35kg, Ultra-slim 14.9mm

Figure 3.20: Display the QR code

User Profile Page

The user can see their own information on this page and edit it.



Username : Computer

Name: Computer

Surname: Test

Telephone number: 1234

Email: Computer@gmail.com

Address:

Username : Mobile

Name

Surname

Telephone number

Email

Address

Figure 3.21: User Profile Page

Software Implementation and Deployment

Smart Contract Design

According to the system requirements, we have separated into 2 contracts covering public key infrastructure (PKI) management smart contract, and product tracking and user information management smart contract.

PKI Smart Contract

Global Variables

1) Structure data for storing user’s credentials
   - Credential’s value : string
   - Credential’s algorithm : string
2) Structure data for public key infrastructure certificate (PKI)
   - Certificate Version : string
   - Issuer Name : string
   - Validity Period (issue date + expiration) : string
   - Public key’s Algorithm : string
   - Public key’s value : string
   - Issuer Signature : string
   - Certificate Fingerprint : string

3) Structure data for user’s information or attribute certificate
    - Certificate Version : string
    - Issuer Name : string
    - Attribute’s holder PKI serial number: string
    - Validity Period (issue date + expiration) : string
    - Attributes or Information’s value : string
    - Issuer Signature : string
    - Certificate Fingerprint : string
4) Hash table map username to user’s credential
5) Hash table map username to encrypted session key
6) Hash table map attribute certificate’s serial number to user’s attribute certificate data
7) Hash table map PKI certificate’s serial number to user’s PKI certificate body data

Methods

1) Store session key of attribute’s information
2) Get session key of attribute’s information
3) Store PKI certificate
4) Get PKI certificate
5) Store user’s attribute certificate
6) Get user’s attribute certificate
7) Store user’s credential
8) Get user’s credential

Product and User Information Smart Contract

Global Variables

1) Structure data for store a track record of product
    - Product Name : string
    - Product Origin : string
    - Product Original Manufacturer : string
    - Product Original details : string
    - Current Owner of Each Record : string
    - Timestamp of Each Record : string
2) Array of list product ID
3) Hash table map product ID to array of it’s track record

Methods

1) Get array of product’s track record
2) Get last track history of product’s record
3) Get list of all product ID
4) Store new product to the system
5) Add new record or track of the product

Front-end Design

Webpage Design

Landing/Home Page

This is the first page of the website which should include an approach to register and log in to the system. Moreover, it should include an input that can be submitted with the product ID to see the track table. Therefore, this page should not be privileged that any people can access and interact with it.

User Profile Page

This page should be available after the user is authenticated to Flask. This is used to display user’s personal information such as name, surname, email address, phone number, and address. Moreover, it should enable users to configure or change their personal information while using the same input that is used to display the information.

Product Listing Page

This page should also be available after the user is authenticated to Flask. It should be embedded with 4 functions which are searching for product by ID, showing list of the product that own by the authenticated user, transferring product ownership portal, and adding new product into system.

Product Tracking Record Page

This page is used to display the record of a specific accordance with its product ID or QR code. It should list all the product history starting with the current record at the top, and so on. Specifically, this page is publicly visible to any people so that they do not need to authenticate themselves before accessing this page.

Flask Routing Design

Home Page Route

This route is used to render the home page webpage when requested

User Profile Route

This route is used to render the user profile page after being authenticated to the server. However, the user will be unable to access without authorization even passing with a static URL into the web browser.

Authentication Route

This route will accept the result of log in or registration in terms of post methods (HTML form that embedded the result in payload body instead of URL). It will check whether the request is registration or authentication. It should relay the information of registration to the blockchain via smart contract interface. While authentication will include the smart contract to obtain the credential of that request user to compare with the one submitted by the user. Then, it should establish the session for that user and redirect the user to the product listing page. It should also induce the user client to decrypt the user's information from blockchain with private key after finishing the retrieval process. Moreover, it should notify the user whether the process is successful or failed.

Product Listing Route

This route will first communicate with the smart contract to retrieve all of product information related to that user, then will be listed in a proper manner. Then, it can render the product page to the user so that user can see and manage their inventory system such as get the track, adding new product, obtaining product tracking QR code, and transfer the ownership to other users. Moreover, it should notify the user whether operations occurred on this page are successful or failed.

Product Adding Route

This route is an intermediate route after the user has submitted the form for adding a new product to the system. This route includes user authentication so that it should prevent unauthorized users maliciously adding wrong product to the system. Then, it will automatically redirect to the product listing page.

Product Ownership Transferring Route

This route is an intermediate route after the user has submitted the form of transferring the ownership of the product to other users. This route includes user authentication so that it should prevent unauthorized users from maliciously transferring ownership of that products. Then, it will automatically redirect to the product listing page.

Product Tracking Record Route

This route will first retrieve the product tracking information regarding its product ID specified in the homepage or the QR code link. This route accepts the parameter as GET method which product ID can also be visible in the URL. It should also notify the user if there is no product matched with that specified product ID.

Log Out Route

It is an intermediate route manage for logging out requests from the user. It will completely end the session of the user to the system and redirect the user to the landing or home page.

Python and Smart Contract Interface Design

Smart Contract Deployment

The approach of deployment is not restricted. However, the system should response its application binary interface (ABI), and its address in the blockchain back so that we can interface the smart contracts with other frameworks or programming language suites.

Smart Contract Interfacing

We have designed to interface with smart contracts using Python Web3 library that requires both contract ABI and address before communicating with the smart contracts. Moreover, the RPC address of blockchain is also required before establishing the communication. We have separated it into 2 interface files which are PKI smart contract interface and product tracking and user information smart contract for convenience programming and debugging purpose. The Python functions as interfaces to interact with smart contracts in blockchain can be categorized into 2 groups:

- Function that induces new block adding that will used Web3 transact method
    - Store session key of attribute’s information
    - Store PKI certificate
    - Store user’s attribute certificate
    - Store user’s credential
    - Store new product to the system
    - Add new record of track of the product to the system
- Function that only view the block data that will used Web3 call method
    - Get session key of attribute’s information
    - Get PKI certificate
    - Get user’s attribute certificate
    - Get array of product’s track record
    - Get last track history of product’s record
    - Get list of all product ID

Testing Design

We will use Selenium library in Python to test the system along with manually config testing environment mainly used from the smart contract interface. First, we need to make Selenium to mimic the user process specified in use case diagram with pre-configured artificial data to measure the system functionality and performance. We also include a timer library to set the initial point of timer and end point of that timer to measure running time at any specific program section.

Functional Test

Data Confidentiality Protection

We will compare the test result between directly retrieving the personal data in the blockchain via smart contract interface that does not require any authentication, and properly retrieving data from the system that requires authentication (properly decrypt with private key).

Block Generation

We will test all of the functions specified in the system requirements to compare the amount of block generated of each operation between theory implementation (Code of each function in smart contract interfaces) and exact block generated in the blockchain simulation network (Ganache).

Performance Test

Time Performance of Adding New Product to the system

- Navigate Selenium to the product listing page while also obtained the authorization from the Flask server
- Selenium catches the add product button and press it
- Selenium catches the form to add the product
- Selenium puts the artificial data into the form and submit
- After submitting the form, record the first timestamp
- Selenium waits for the redirection
- After redirect to the product listing page, record the second timestamp
- Store the running time by the subtracting the second timestamp with the first recorded timestamp and store in the list
- Displaying the graph with x-axis indicate the number of iterations, and y-axis indicate the running time in millisecond
- Repeat with 100 iterations, and then calculate the average running time

Time Performance of Retrieval of Product tracking

- Navigate Selenium to the landing page
- Selenium catches the form that used to submit the product ID
- Selenium puts the product ID that is in the system and submit the form
- After submitting the form, record the first timestamp
- Selenium waits for the redirection
- After redirect to the product tracking record page, record the second timestamp
- Selenium navigates back to the landing page
- Store the running time by the subtracting the second timestamp with the first recorded timestamp and store in the list
- Repeat with other product ID and the above process 100 iterations
- Calculate the average running time for each product ID

System Deployment Design

This project has been simulated simulating system environment visualized in figure 3.22 using only one device so that the performance will be different with real environment such as blockchain test net or real blockchain network.

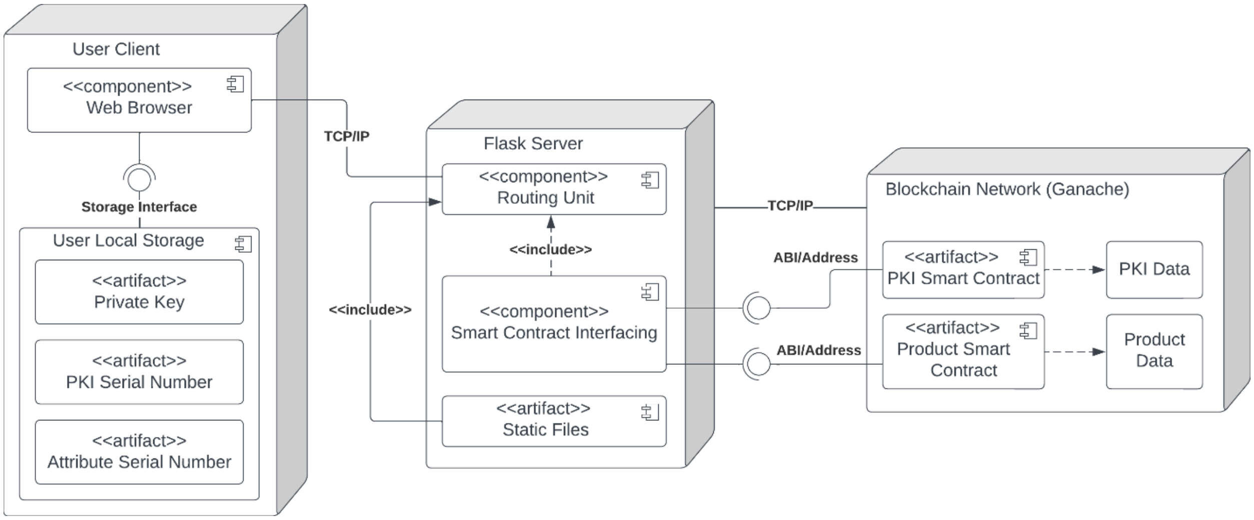


Figure 3.22: Deployment Model of Product Tracking Application

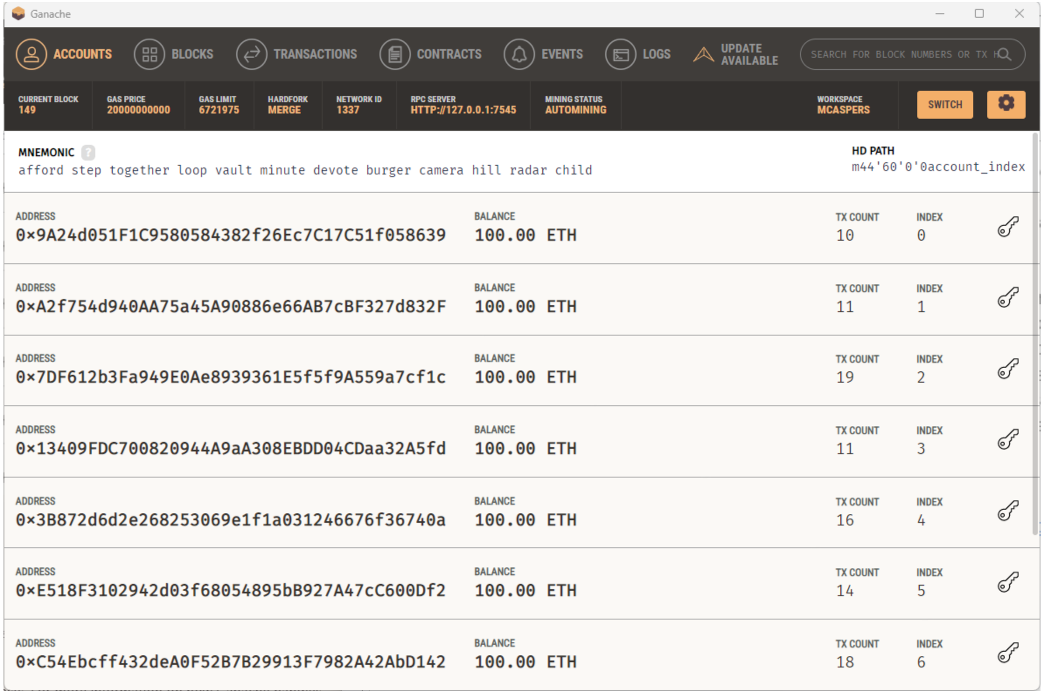
As you can see, Ganache will be simulated as a blockchain network using local IP address and port to establish RPC connection or TCP/IP connection to the Flask server. As the same way, Flask server that also be deployed in local IP address and port will also establish the TCP/IP connection to the user web browser which is also simulated to store private key and others. The specifications of the device that use to simulate this system can be listed below:

Host Device

- 24 GB Memory
- CPU 3.5 GHz
- SSD 512GB
- 4 Cores CPU
- Windows 11 OS

Blockchain Network Service on Host (Ganache)

- 10 Nodes / Accounts
- Ethereum Merge Fork which used PoA simulated as its consensus mechanism
- Deploy on local host with Port 7545, Network ID 5777
- Each account maintains 10 ETH
- The gas limit is 6721975
- The gas price is 0.02 ETH



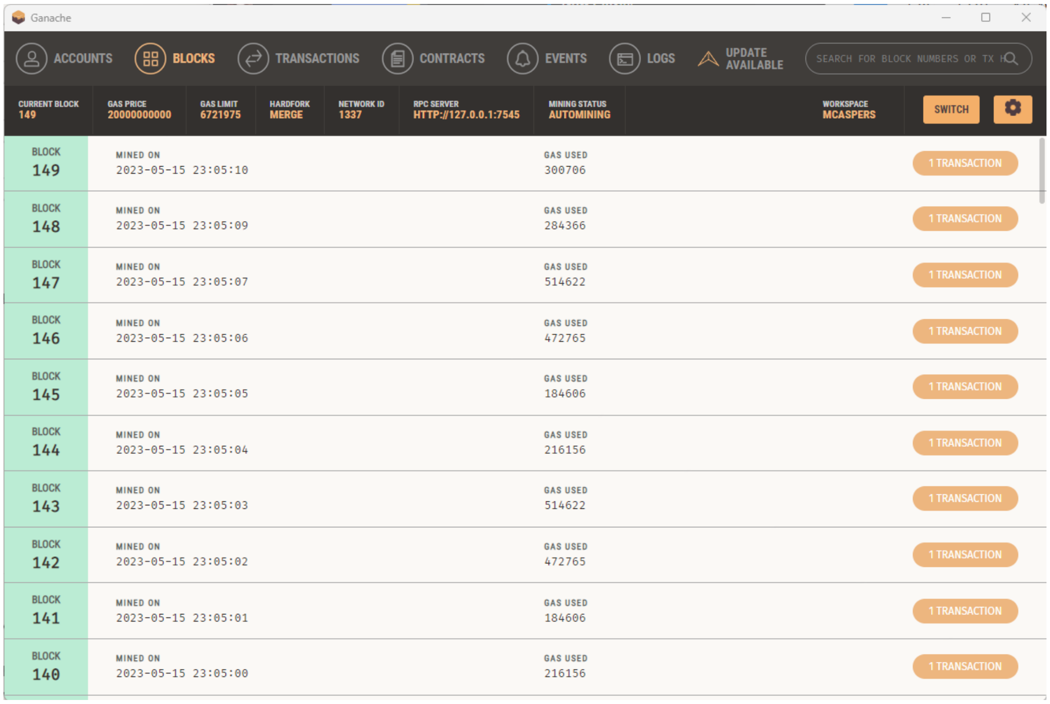
CURRENT BLOCK 149 | GAS PRICE 20000000000 | GAS LIMIT 6721975 | HARDFORK MERGE | NETWORK ID 1337 | RPC SERVER HTTP://127.0.0.1:7545 | MINING STATUS AUTOMINING | WORKSPACE MCASPERS

MNEMONIC
afford step together loop vault minute devote burger camera hill radar child

HD PATH
m44'60'0'0account_index

ADDRESS	BALANCE	TX COUNT	INDEX
0×9A24d051F1C9580584382f26Ec7C17C51f058639	100.00 ETH	10	0
0×A2f754d940AA75a45A90886e66AB7cBF327d832F	100.00 ETH	11	1
0×7DF612b3Fa949E0Ae8939361E5f5f9A559a7cf1c	100.00 ETH	19	2
0×13409FDC700820944A9aA308EBDD04CDaa32A5fd	100.00 ETH	11	3
0×3B872d6d2e268253069e1f1a031246676f36740a	100.00 ETH	16	4
0×E518F3102942d03f68054895bB927A47cC600Df2	100.00 ETH	14	5
0×C54Ebcff432deA0F52B7B29913F7982A42AbD142	100.00 ETH	18	6

Figure 3.23: Simulated Ethereum account in Ganache Blockchain



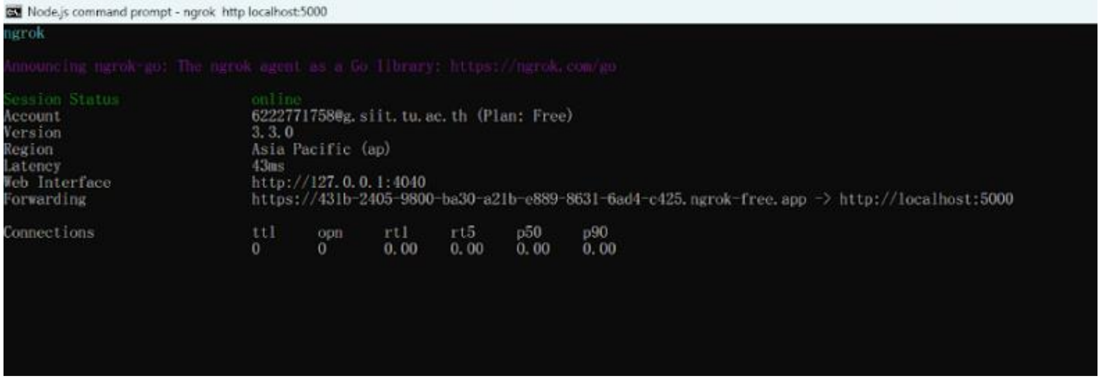
CURRENT BLOCK 149 | GAS PRICE 20000000000 | GAS LIMIT 6721975 | HARDFORK MERGE | NETWORK ID 1337 | RPC SERVER HTTP://127.0.0.1:7545 | MINING STATUS AUTOMINING | WORKSPACE MCASPERS

BLOCK	MINED ON	GAS USED	
149	2023-05-15 23:05:10	300706	1 TRANSACTION
148	2023-05-15 23:05:09	284366	1 TRANSACTION
147	2023-05-15 23:05:07	514622	1 TRANSACTION
146	2023-05-15 23:05:06	472765	1 TRANSACTION
145	2023-05-15 23:05:05	184606	1 TRANSACTION
144	2023-05-15 23:05:04	216156	1 TRANSACTION
143	2023-05-15 23:05:03	514622	1 TRANSACTION
142	2023-05-15 23:05:02	472765	1 TRANSACTION
141	2023-05-15 23:05:01	184606	1 TRANSACTION
140	2023-05-15 23:05:00	216156	1 TRANSACTION

Figure 3.24: List of block transactions in Ganache Blockchain

Web-Hosting Service on Host

- Ngrok free account with port 80
- Flask web server will be deployed in local address with port 5000
- Ngrok will interface localhost with port 5000 into public IP with port 80



```
Node.js command prompt - ngrok  http localhost:5000
ngrok

Announcing ngrok-go: The ngrok agent as a Go library: https://ngrok.com/go

Session Status                online
Account                       6222771758@g.siit.tu.ac.th (Plan: Free)
Version                       3.3.0
Region                        Asia Pacific (ap)
Latency                       43ms
Web Interface                 http://127.0.0.1:4040
Forwarding                    https://431b-2405-9800-ba30-a21b-e889-8631-6ad4-c425.ngrok-free.app -> http://localhost:5000

Connections                   ttl     opn     rt1     rt5     p50     p90
                              0       0       0.00    0.00    0.00    0.00
```

Figure 3.25: The output of cmd after publicizing the website address

Chapter 4

Evaluation

User Experience Evaluation

Positive Comments

- Accessing product tracking information is very easy thanks to QR code
- Authentication process provide feeling of information safety
- Function is easy to use and understand
- Responsive in both desktop and mobile users

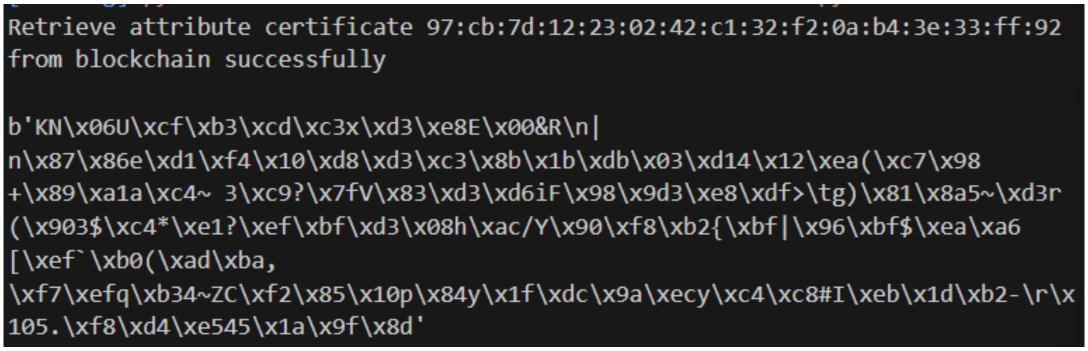
Negative Comments

- The web interface design still not so attractive
- Some of function expose high response time
- Should include the image of user's profile and the products
- Should have more function such as user search, recovery password loss

Functionality Evaluation

Data Confidentiality Protection

As you can see, figure X.X presents the result when a user tries to directly access the blockchain to unauthorizedly obtain other users' information. However, the result is unreadable.

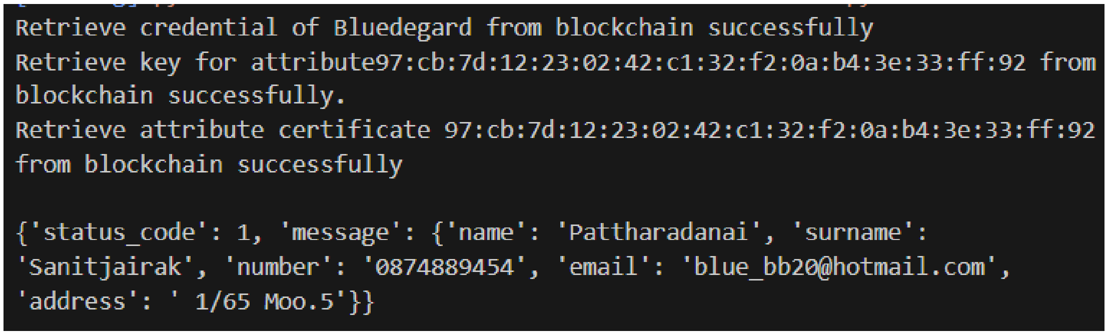


```
Retrieve attribute certificate 97:cb:7d:12:23:02:42:c1:32:f2:0a:b4:3e:33:ff:92
from blockchain successfully

b'KN\x06U\xcf\xb3\xcd\xc3x\xd3\xe8E\x00&R\n|
n\x87\x86e\xd1\xf4\x10\xd8\xd3\xc3\x8b\x1b\xdb\x03\xd14\x12\xea(\xc7\x98
+\x89\xa1a\xc4~ 3\xc9?\x7fV\x83\xd3\xd6iF\x98\x9d3\xe8\xdf>\tg)\x81\x8a5~\xd3r
(\x903$\xc4*\xe1?\xef\xbf\xd3\x08h\xac/Y\x90\xf8\xb2{\xbf|\x96\xbf$\xea\xa6
[\xef`\xb0(\xad\xba,
\xf7\xefq\xb34~ZC\xf2\x85\x10p\x84y\x1f\xdc\x9a\xecy\xc4\xc8#I\xeb\x1d\xb2-\r\x
105.\xf8\xd4\xe545\x1a\x9f\x8d'
```

Figure 4.1: Ciphertext generated from directly access to data in blocks

On the other hand, when a user accesses the system with an existing and qualified private key, the result of information retrieval is readable. Thus, accessing personal information requires a qualified user's private key.



```
Retrieve credential of Bluedegard from blockchain successfully
Retrieve key for attribute97:cb:7d:12:23:02:42:c1:32:f2:0a:b4:3e:33:ff:92 from
blockchain successfully.
Retrieve attribute certificate 97:cb:7d:12:23:02:42:c1:32:f2:0a:b4:3e:33:ff:92
from blockchain successfully

{'status_code': 1, 'message': {'name': 'Pattharadanai', 'surname':
'Sanitjairak', 'number': '0874889454', 'email': 'blue_bb20@hotmail.com',
'address': ' 1/65 Moo.5'}}
```

Figure 4.2: Plaintext generated from properly access to data in blocks

Block Generation

- User registration consumes 3 block(s)
    1. Block to store a PKI certificate
    2. Block to store an encrypted attribute certificate (user's information)
    3. Block to store an encrypted attribute session key
- Adding new product to the system consumes 1 block(s)
    1. Block to store a new record of product with one tracking record
- Transferring the ownership of the product consume 1 block(s)
    1. Block to store a new modification of the product tracking record
- Changing or Updating user's information consumes 2 block(s)
    1. Block to store a new encrypted attribute certificate
    2. Block to store a new encrypted attribute session key

Performance Evaluation

Time Performance of Adding New Product to the system

The average time of in this testing is 3.914 second

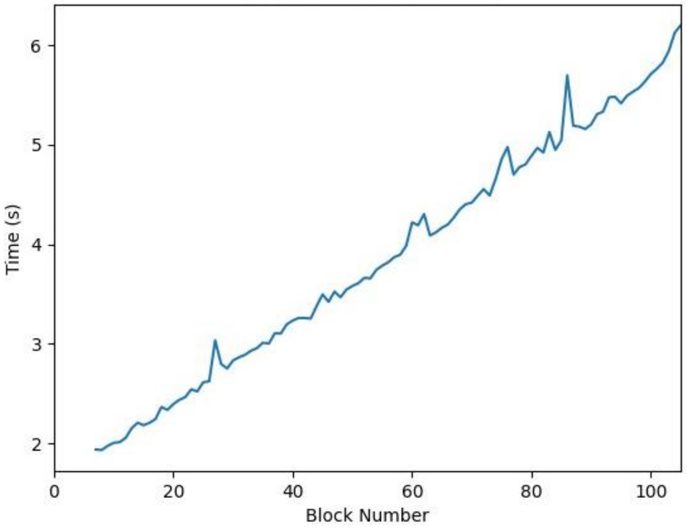


Figure 4.3: Relation between number of block and time consuming in add product function

Time Performance of Retrieval of Product tracking

The result of this testing is shown in figure X.X indicating that the product information retrieval time does not depend on the number of history tracking record of the product in the blockchain. The average time of the testing is:

- 1 Record(s) : 778.208 ms
- 2 Record(s) : 785.919 ms
- 3 Record(s) : 795.873 ms
- 4 Record(s) : 774.222 ms

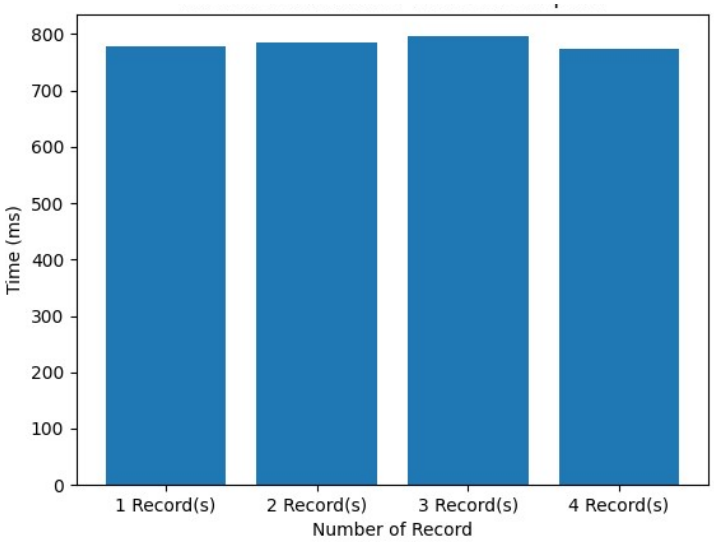


Figure 4.4: Bar graph of track record retrieval time consumption comparing with products with different number of tracking records

Chapter 5

Conclusion and Discussion

Conclusion

To summarize our project, this capstone product is designing and developing decentralized application in field of product tracking system that enhance the credibility and security to the product data by the unique characteristics of the blockchain. Moreover, we also implemented web-based service with proper authentication system that can offer various range of users such as Desktop users, mobile users, or any of devices or platforms. However, our project is deployed on simulation environments that some problem in real case might not be addressed properly. Overall, evaluation is in a good criterion including functional evaluation and performance evaluation.

Obtained Benefits

1) Understand how to utilize the characteristics of the blockchain
2) Understand how to design and develop full system of application including both UI/UX design, front-end design, back-end design, and requirement engineering.
3) Understand how to develop decentralized applications

Performance and Limitations

When the system deploys on the real blockchain network .There will be some factors affecting to the performance of the system as below.

Gas Fee and ETH

The blockchain ecosystem depends on gas fees and the use of ether (ETH). We need to take gas fees into account when implementing the product tracking system on a blockchain network to determine the financial impact. The cost of gas can change based on network demand and congestion. The potential costs of gas fees and the availability of ETH, it is crucial to evaluate the economic viability of using the blockchain for product tracking.

Performance Degradation due to Traffic Congestion

Blockchain networks, particularly open ones, may perform worse when there is a high volume of transactions and network congestion. We must assess the product tracking system's performance and scalability under various load conditions before deploying it. In-depth testing and optimization are required to make sure that our system can handle more traffic without suffering performance or response time degradation. In order to do this, it may be necessary to investigate scalability options like layer-2 protocols, sidechains, or shrading.

Deployment and Maintenance Costs

Costs that need to be carefully considered will be incurred when deploying and maintaining a product tracking system on a blockchain network. These expenses cover the initial deployment fee, maintenance and support fees, and any potential upgrades or adjustments needed as the system develops. The long-term financial effects of operating the product tracking system on the blockchain network must be carefully considered to make sure the advantages outweigh the costs.

Security Issues

There are many security issues with which systems must be concerned, such as

Man-in-the-Middle Attack

When information is being exchanged between the web server and user clients, there is a chance that unauthorized parties will intercept it and change it. It is essential to mandate the use of the Transport Layer Security (TLS) or Secure Sockets Layer (SSL) protocols in order to reduce this risk. These encryption protocols offer secure communication channels, eavesdropping on private information.

Weak Input Form Validation

To ensure data integrity, the current input form validation, which is limited to max length, is insufficient. It is advised to implement regular expressions and pattern matching to strengthen input form validation. This will improve data validation and avoid malicious or invalid input from being submitted.

Insecure Password Submission

Users' passwords submitted to the hosting server are not currently sufficiently secure. It is advised to use asymmetric encryption, such as RSA, to secure the password submission process in order to address this. Key-exchange algorithms will also improve the security of password transmission and protect sensitive user data.

Inadequate Authentication Mechanisms

To increase security, the existing authentication mechanisms must be improved. The use of access control only is insufficient. Consider using higher-level encryption methods, like Ciphertext-Policy Attribute-Based Encryption (CP-ABE), to improve authentication and access control and make sure that only people with the proper authorization can access sensitive product data.

Side Channel Attacks

Attackers may trick users into disclosing sensitive information through side channel attacks like phishing. Stronger validation and accountability measures should be implemented in order to reduce this risk. Implement thorough validation procedures and warn users of potential phishing attacks to encourage users to be cautious when sharing product information.

Attacks on Web-Hosting Server

Various attacks, such as Distributed Denial-of-Service (DDoS) attacks and unauthorized access, can be launched against web hosting servers. Ensure that the hosting provider you choose is reputable, reliable, and trustworthy. In order to guard against unauthorized access attempts, mutual authentication and challenge-response protocols should also be used.

Blockchain Overhead:

The use of blockchain technology may result in extra overhead, such as high block consumption and storage needs. Reduce block consumption by optimizing the system's operations and features. To reduce storage costs and boost scalability, think about moving some data, such as product and user images, to a cloud storage solution.

References

Abdelhadi Dyouri. (August 2021). *How To Create Your First Web Application Using Flask and Python 3.* https://www.digitalocean.com/community/tutorials/how-to-create-your-first-web-application-using-flask-and-python-3

Antolin, M. (2022). *What Is Proof-of-Authority?* https://www.coindesk.com/learn /what-is-proof-of-authority/

Ashish, Vanessa, Mohammed, Erkan. (October 2022). *Smart Contracts Could Improve Efficiency and Transparency In Financial Transactions.* https://www.spglobal.com/en/research-insights/featured/special-editorial/smart-contracts-could-improve-efficiency-and-transparency-in-financial-transactions

Ben Lutkevich. (2020). *HTML (Hypertext Markup Language).* https://www.theserverside.com/definition/HTML-Hypertext-Markup-Language

Buterin, V. (2014). *A Next-Generation Smart Contract and Decentralized Application Platform.* https://ethereum.org/en/whitepaper/

CertiPro Solutions, LCC. (2021). *The Top Five Benefits of Traceability for Food & Beverage Manufacturers.* https://certiprosolutions.com/blog/the-top-five-benefits-of-traceability-for-food-beverage-manufacturers/

Chauhan, H., Gupta, D., Gupta, S., Singh, A., Aljahdali, H. M., Goyal, N., Noya, I. D., & Kadry, S. (2021). *Blockchain Enabled Transparent and Anti-Counterfeiting Supply of COVID-19 Vaccine Vials. Vaccines, 9(11), 1239.* https://doi.org/10.3390/vaccines9111239

Cointelegraph. (March 2023). *Proof-of-authority vs. proof-of-stake: Key differences explained*. https://cointelegraph.com/blockchain-for-beginners/proof-of-authority-vs-proof-of-stake-key-differences-explained

ConsenSys Software Inc. (2022). *Ganache Ethereum workspace overview*. https://trufflesuite.com/docs/ganache/concepts/ethereum-workspace/overview/

GeeksforGeeks. (2022). *Consensus Algorithms in Blockchain*. https://www.geeksforgeeks.org/consensus-algorithms-in-blockchain/

Georgia Weston. (August 2022). *What is Ganache Blockchain* https://101blockchains.com/ganache-blockchain/

Gregory McCubbin. (March 2023). *Intro to Web3.py, Ethereum for Python Developers* https://www.dappuniversity.com/articles/web3-py-intro

James Humphreys. (April 2023). *What is traceability and how to implement in manufacturing?.* https://katanamrp.com/blog/product-traceability/

Keeping TABS. (2017). *VeChain is using tech to combat counterfeiting*. https://medium.com/@canvas8/vechain-is-using-tech-to-combat-counterfeiting-d5db7172bb3c

King, S., & Nadal, S. (2012). *PPCoin: Peer-to-Peer Crypto-Currency with Proof-of-Stake.* https://peercoin.net/assets/paper/peercoin-paper.pdf

Ngrok, Inc. (2023). *Your app online, with one command.* https://ngrok.com/

Phemex. (2022). *What Is Proof-of-Authority: Staking Credibility Instead of Coins*. https://phemex.com/academy/what-is-proof-of-authority

Scantrust, Inc. (2022). *Product Traceability in Supply Chains: The Definitive Guide.* https://www.scantrust.com/product-traceability-definitive-guide/

Szabo, N. (1997). *Formalizing and Securing Relationships on Public Networks*. https://firstmonday.org/ojs/index.php/fm/article/view/548/469

Venafi, Inc. (2022). *What is PKI and How Does it Work?*. https://venafi.com/machine-identity-basics/what-is-pki-and-how-does-it-work/

Wikipedia. (May 2023). ***Blockchain***. https://en.wikipedia.org/wiki/Blockchain

William Fisher. (January 2015). *Benefits of Food Traceability.* https://www.food-safety.com/articles/4192-benefits-of-food-traceability