A Minor Project Report On

Safeguarding Against Counterfeits With Blockchain Based QR-Code

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

Submitted By:

K.SUSMITHA	21UP1A0590
R.SUMANA	22UP5A0510
B.AKSHITHA	21UP1A0571

Under the guidance of

Mrs.AMULYA RACHANA

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VIGNAN'S INSTITUTE OF MANAGEMENT AND TECHNOLOGY FOR WOMEN

(An Autonomous Institution)

(Affiliated to JNTUH, Hyderabad, Accredited by NBA)

Kondapur(v), Ghatkesar(M), Medchal-Malkajgiri(D)-501301

[2021-2025]

VIGNAN'S INSTITUTE OF MANAGEMENT AND TECHNOLOGY FOR WOMEN
(An Autonomous Institution)
[Sponsored by Lavu Educational Society, Affiliated to JNTUH & Approved by AICTE, New Delhi]
Kondapur (V), Ghatkesar (M), Medchal - Malkajgiri (D) - 501 301. Phone: 96529 10002/3

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project work titled **"Safeguarding Against Counterfeits With Blockchain Bases QR Code"** submitted by **K.SUSMITHA (21UP1A0590),R.SUMANA(22UP5A0510),B.AKSHITHA(21UP1A0571)** of B-Tech in the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Computer Science and Engineering to the Vignan's Institute of Management and Technology for Women is a record of bona-fide work carried out by them under my guidance and supervision. The results embodied in this project report have not been submitted to any university for the award of any degree and the results are achieved satisfactorily.

INTERNAL GUIDE

Mrs.AMULYA RACHANA

(Assistant Professor)

HEAD OF THE DEPARTMENT

Mrs.M.PARIMALA

(Associate Professor)

(External Examiner)

DECLARATION

I, hereby declare that the results embodied in this dissertation entitled "**Safeguarding Against Counterfeits With Blockchain Bases QR Code**" is carried out by us during the year 2024-2025 in partial fulfillment of the award of Bachelor of Technology in Computer Science and Engineering from VIGNAN'S INSTITUTE OF MANAGEMENT AND TECHNOLOGY FOR WOMEN is an authentic record of our work carried under the guidance of Mrs.Amulya Rachana (Assistant Professor), Department of Computer Science and Engineering. I have not submitted the same to any other university or organization for the award of any other degree.

K.SUSMITHA(21UP1A0590)

R.SUMANA(22UP5A0510)

B.AKSHITHA(21UP1A0571)

ACKNOWLEDGEMENT

I would like to express sincere gratitude to **Dr. G. AppaRao Naidu, Principal**, **Vignan's Institute of Management and Technology for Women** for his timely suggestions which helped me to complete the project in time.

I would also like to thank our madam **Mrs. M Parimala, Head of the Department, Computer Science and Engineering,** for providing me with constant encouragement and resources which helped me to complete the project in time.

I would like to thank out project guide**, Mrs Amulya Rachana, Assistant Professor, Computer Science and Engineering,** for his timely cooperation and valuable suggestions throughout the project. I am indebted to his for the opportunity given to work under his guidance.

My sincere thanks to all the teaching and non-teaching staff of the Department of Computer Science and Engineering for their support throughout my project work.

K.SUSMITHA(21UP1A0590)

R.SUMANA(22UP5A0510)

B.AKSHITHA(21UP1A0571)

Abstract

Blockchain technology has garnered significant attention in recent years, primarily known for its impact on financial transactions. However, its potential extends beyond the realm of finance, offering a disruptive force in various industries. This project explores the utilization of blockchain technology as a means to combat counterfeiting. The study provides an overview of existing solutions in the anti-counterfeit domain, explores different blockchain technologies, and highlights the key characteristics that make blockchain particularly compelling for this use case. In pursuit of an effective anti-counterfeit system, three distinct concepts and the expansion of an existing system are examined in detail. The findings reveal that successfully curbing counterfeiting cannot be achieved solely through technological means. Rather, a holistic approach encompassing heightened awareness, legal measures against counterfeiters, a robust alert system, and tamper-proof packaging emerges as essential components.

By integrating blockchain technology into this comprehensive approach, the research demonstrates the potential for an efficient and comprehensive solution to combat counterfeiting. The transparency, immutability, and decentralized nature of blockchain contribute to enhancing product authentication, streamlining transactions, and fostering trust among stakeholders. This project emphasizes the significance of combining technological advancements with complementary strategies to effectively reduce counterfeiting and protect consumer interests in the digital age.

INDEX

LIST OF FIGURES

1. INTRODUCTION

Although it may seem like a far-off idea, we are surrounded by a lot of counterfeits. From fashion and retail products to software, digital media, electronics, piracy, and intellectual property, reports put the cost of counterfeiting somewhere around $600bn a year in the US alone. In fact, the International Chamber of Commerce predicts that the ―negative impacts of counterfeiting and piracy are projected to drain US$4.2 trillion from the global economy and put 5.4 million legitimate jobs at risk by 2022. In Pharmaceuticals, the counterfeit medicine market is now responsible for around 1 million deaths per year, in an industry estimated to be worth $75bn annually. In fact, the counterfeit medicine industry is estimated to be growing at twice the rate of legitimate pharmaceuticals, making it up to 25 times more lucrative than the global narcotics trade. Trust is a central element in all transactions. No matter if sending money or exchanging goods, it becomes difficult if there is no trust between the entities involved. It becomes even more difficult, as with many transactions, third parties are involved, such as banks. Often, not only one third-party is involved in a transaction, but multiple. An international money transfer does not only include the bank of the sender, the bank of the receiver, but also multiple intermediary entities such as clearing houses. The entities involved in the transaction do not only have to trust each other, but also the third parties. Removing these third parties can decrease transaction cost, facilitate faster transactions and add more transparency. Bitcoin has successfully shown that removing such third-parties is possible. The cryptocurrency permits direct sending coins to a transaction partner, without the need to use banks and clearing houses. The assets are directly transferred from one account to another. There are no intermediaries and thereby no need to trust third parties. In addition, the question if a transaction is valid is not answered by an institution, but by algorithms used.

Therefore, it completely removes the need to trust any third party. The technology behind Bitcoin, the blockchain, can however not only be used for financial transactions and crypto currencies in general. The technology has potential to ―redefine the digital economy‖ [10], because it allows immutable transactions, which can be checked at all times from everyone. This is because the information is publicly available and distributed globally. It is chronologically updated and cryptographically sealed‖[11]. The full range of applicable use cases for this technology has to be seen, but tracking ownership and history of a product is surely one of them. This paper explores the possibility to reduce counterfeit using blockchain technology.

Authentication, the act of establishing or conforming something as genuine. Authentication is of utmost importance because the use of counterfeit medicines can be harmful to the health and wellbeing of the patients. Their use may result in treatment failure or even death. Authentication is generally done through the overt or covert features upon the product-We now have more fakes than real drugs in the market. Christophe Zimmermann, the anti-counterfeiting and piracy coordinator of the World Customs Organization. Current anti-counterfeiting supply chains rely on a centralized authority to combat counterfeit products. This architecture results in issues such as single point processing, storage, and failure. Blockchain technology has emerged to provide a promising solution for such issues. In this paper, we propose the block-supply chain, a new decentralized supply chain that detects counterfeiting attacks using blockchain and Near Field Communication (NFC) technologies. Block-supply chain replaces the centralized supply chain design and utilizes a new proposed consensus protocol that is, unlike existing protocols, fully decentralized and balances between efficiency and security. Our simulations show that the proposed protocol offers remarkable performance with a satisfactory

level of security compared to the state of the art consensus protocol Tendermint.

EXISTING SYSTEM

In this paper author is using Blockchain technology to authenticate supply chain products as this product may be supplied from multiple third party distributors and this distributors can make clone/fake/counterfeits of this product BAR CODE and then manufacture fake products and add this counterfeit label to fake product and this fake products can cause huge loss of financial and lives if fake medicine manufacture.

Not only supply chain any other online transaction require third party to complete transaction and peoples has to trust on third parties to complete their transaction and sometime this third parties can make fraud transaction or misuse user data.

Disadvantages

- We now have more fakes than real drugs in the market
- Cloning of product
- Limited Traceability
- Manual Verification
- Paper Documentation
- Centralized Databases
- Lack of Transparency
- High Costs
- Inefficiency in Recall Processes

PROPOSED SYSTEM

To avoid this problem author using Blockchain technology which does not require any third party and verification will be done by software algorithm itself without involvement of any third party. In this to avoid forge counterfeit we are converting all products details/barcode into digital signatures and this digital signatures will be stored in Blockchain server as this Blockchain server support tamper proof data storage and nobody can hack or alter its data and if by an chance if its data alter then verification get failed at next block storage and user may get intimation about data alter.

Advantages

In supply chain also all products barcode digital Blockchain signatures will be stored and if any third party distributor make clone of barcode then its signature will be mismatch and counterfeit will be detected.

- Immutable Record
- Traceability
- Authentication
- Smart Contracts
- Decentralization
- Enhanced Security
- Consumer Empowerment

2. LITERATURE SURVEY

The purpose of the literature survey is to place each work in the context of its contribution to understanding the research problem being studied. It also identifies new ways to interpret prior research.

Anti-counterfeiting solutions should protect organizations from financial and reputation losses, and, especially in the case of pharmaceutical products, customer safety. [15] argues that good anti-counterfeiting techniques should generally be simple to apply, but difficult to imitate and have four main features:

They should be difficult to duplicate, it should be possible to identify them without special equipment, it should be difficult to re-use them, and it should be visible if they were tampered with. From a product perspective, there are three general technologies to reduce counterfeits.Overt (Visible) Features expected to assist the users to confirm the genuineness of a pack. Such features will be significantly visible, and complex or expensive to reproduce.

This includes holograms, color shifting inks, security threads, water marks etc. The advantage of overt technologies is that they can be checked by the end consumer.Covert (Hidden) Features the rationale of a covert feature is to aid the brand owner to recognize a counterfeit product.

The general public will not be aware of its presence nor will have the resources to confirm it. This includes UV, bi-fluorescent and pen-reactive ink, as well as digital watermarks and hidden printed messages. Covert technologies help to identify counterfeits in the supply-chain and are especially efficient combined with overt technologies.

Track and trace includes Radio Frequency Identification (RFID) tags, Electronic Product Codes (EPCs) and barcodes. Track and trace

technologies allow for simpler tracing of products, thereby enabling the reduction of counterfeits, as the history of a product is available. The tag or barcode is included by the manufacturer. Distributors scan the identification, enabling them to check the authenticity of the product and update the status. Finally, retailers can also scan the product, to check the history and authenticity of the product.

This approach does not only tackle the counterfeit problem, but also enables track and trace through the whole product lifecycle.

3. SYSTEM REQUIREMENTS

Hardware Requirements:

- ✞ System : Minimum i3.
- ✞ Hard Disk : 40 GB.
- ✞ Ram **:** 4 GB.

Software Requirements:

- ✞ Operating system **:** Windows 8.
- ✞ Coding Language : Python 3.7.
- ✞ Data Base **:** MySQL.
- ✞ **IDE** **:** IntelliJ IDEA Community Edition 2024.1/Visual Studio Code

4. SYSTEM DESIGN

SYSTEM ARCHITECTURE

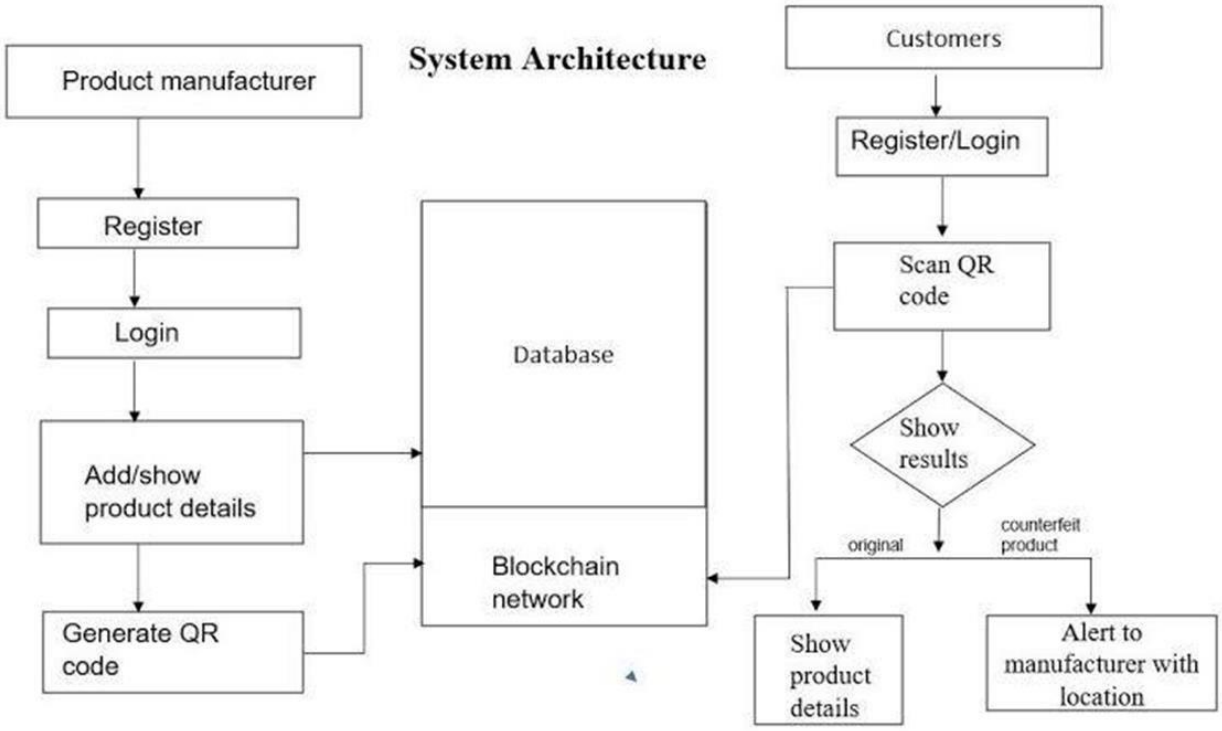

Fig 4.1 :-System Architecture

Methodologies:

The project is carried out based on the following methodologies listed:

1. Blockchain Technology
2. QR Code
3. Unique Identification
4. Blockchain Integration
5. Verification Process
6. Real-time Tracking
7. Secure Authentication
8. User-Friendly Interface

1. Blockchain Technology:

Blockchain is a decentralized, distributed ledger technology that records

transactions across multiple nodes in a secure and immutable manner. Each transaction is cryptographically linked to the previous one, creating a chain of blocks that cannot be altered without consensus from the network participants.

2. **QR Codes:**

Quick Response (QR) codes are two-dimensional barcodes that can store a significant amount of information. They are widely used for encoding URLs, text, or other data that can be quickly scanned and read using a smartphone or dedicated QR code scanner.

3. **Unique Identification:**

Each product is assigned a unique identifier, which is encoded into a QR code during the manufacturing process. This identifier can include information such as the product's serial number, manufacturing date, batch number, and other relevant details.

4. **Blockchain Integration:**

The QR code is linked to a corresponding record on the blockchain, which serves as a digital certificate of authenticity. This linkage ensures that the information stored in the QR code is tamper-proof and verifiable, as any attempt to counterfeit or alter the QR code will be detected by the blockchain.

5. **Verification Process:**

To verify the authenticity of a product, consumers or retailers can scan the QR code using a smartphone or QR code scanner app. The scanning process retrieves the information stored in the QR code and cross-references it with the corresponding record on the blockchain.

6. **Real-Time Tracking:**

Manufacturers can track the movement of their products in real-time by monitoring the blockchain ledger. This visibility allows them to detect any discrepancies or unauthorized activities along the supply chain, such as counterfeit products entering the market.

7. **Secure Authentication:**

Since blockchain technology ensures the integrity and immutability of the data stored in the QR code, consumers can trust the authenticity of the products they purchase. This helps in building brand loyalty and protecting consumers from counterfeit goods.

8. **User-Friendly Interfaces:**

Manufacturers can develop user-friendly interfaces, such as mobile apps or web portals, to facilitate the QR code scanning and verification process. These interfaces can provide additional information about the product, such as its origin, manufacturing process, and warranty details.

5. UML DIAGRAMS

UML is a way of visualizing a software program using a collection of diagrams. The notation has evolved from the work of Grady Booch, James Rumbaugh, Ivar Jacobson, and the Rational Software Corporation to be used for object-oriented design, but it has since been extended to cover a wider variety of software engineering projects. Today, UML is accepted by the Object Management Group (OMG) as the standard for modeling software development.

USE CASE DIAGRAM

It represents the functionality of a system by utilizing actors and use cases. It encapsulates the functional requirement of a system and its association with actors. It portrays the use case view of a system.Use case diagrams have use cases, actors, system, communication link, generalization.

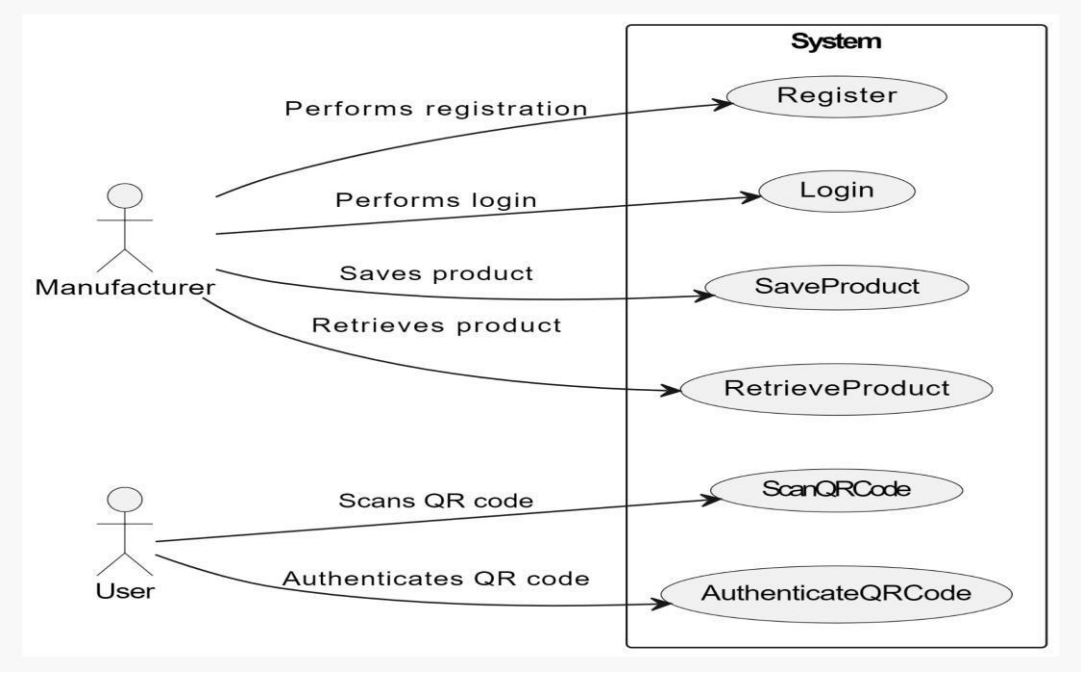

Fig 5.1: Use case diagram

ACTIVITY DIAGRAM

It models the flow of control from one activity to the other. With the help of an activity diagram, we can model sequential and concurrent activities. It visually depicts the workflow as well as what causes an event to occur.

Activity Diagrams describe how activities are coordinated to provide a service which can be at different levels of abstraction.

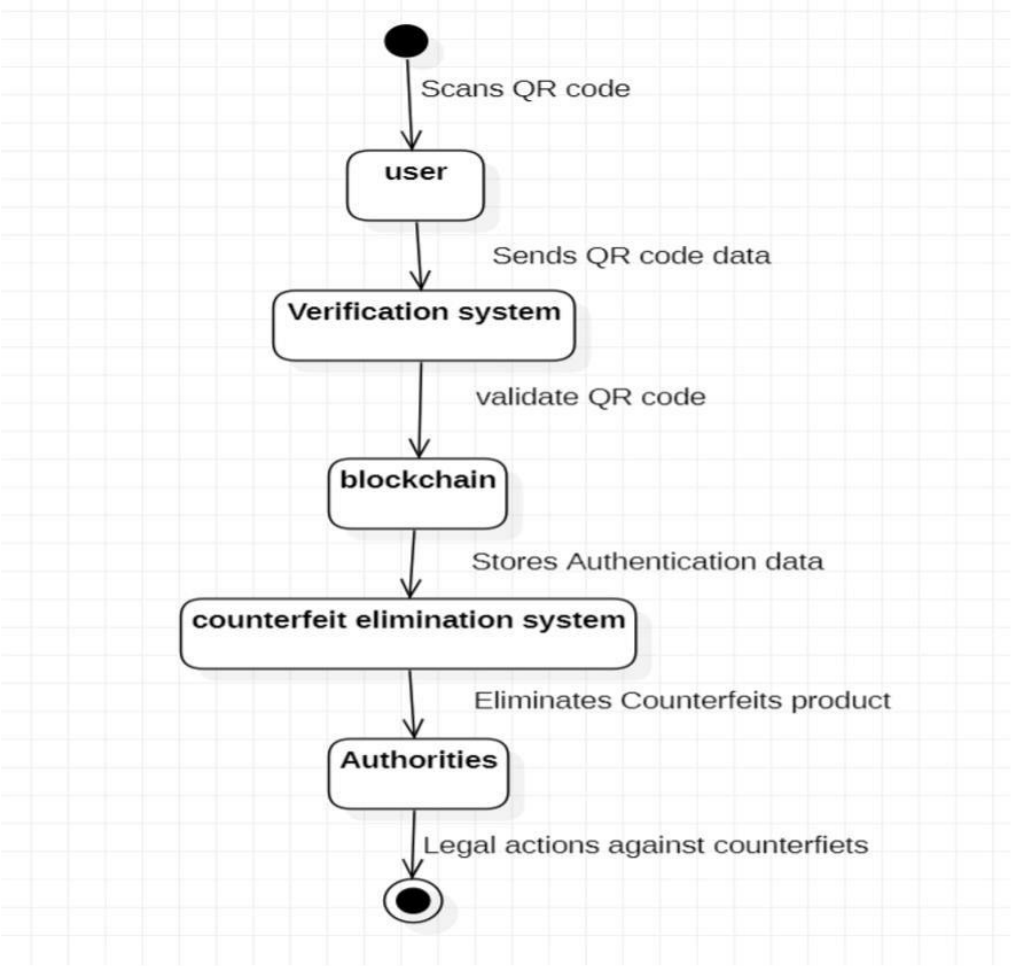

Fig 5.2 : Activity diagram

CLASS DIAGRAM

A class diagram in UML (Unified Modeling Language) is a visual representation of the structure and relationships within a system. It depicts classes, their attributes, methods, and the associations between them. Class diagrams are crucial for understanding the architecture of a system, facilitating communication among stakeholders, and guiding the implementation process in software development. They provide a blueprint for developers to design and implement software systems effectively

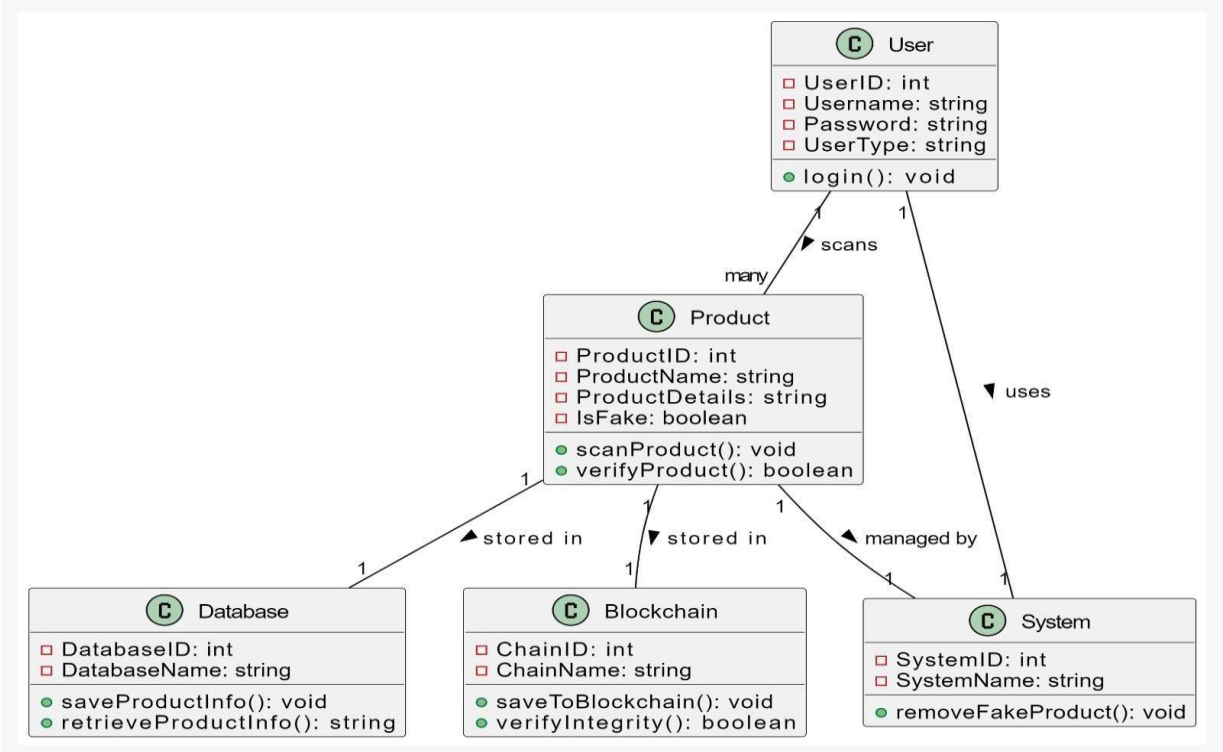

Fig 5.3 : Class diagram

6. SOFTWARE ENVIRONMENT

BLOCKCHAIN

Blockchain allows for the permanent, immutable, and transparent recording of data and transactions. This, in turn, makes it possible to exchange anything that has value, whether that is a physical item or something less tangible. A blockchain has three central attributes. First, a blockchain database must be cryptographically secure. That means in order to access or add data on the database, you need two cryptographic keys: a public key, which is basically the address in the database, and the private key, which is a personal key that must be authenticated by the network.

Next, a blockchain is a digital log or database of transactions, meaning it happens fully online. And finally, a blockchain is a database that is shared across a public or private network. One of the most well-known public blockchain networks is the Bitcoin blockchain. Anyone can open a Bitcoin wallet or become a node on the network. Other blockchains may be private networks. These are more applicable to banking and fintech, where people need to know exactly who is participating, who has access to data, and who has a private key to the database. Other types of blockchains include consortium blockchains and hybrid blockchains, both of which combine different aspects of public and private blockchains.

WORKING OF BLOCKCHAIN

When data on a blockchain is accessed or altered, the record is stored in a “block” alongside the records of other transactions. Stored

transactions are encrypted via unique, unchangeable hashes, such as those created with the SHA-256 algorithm.

New data blocks don't overwrite old ones; they are appended together so that any changes can be monitored. And since all transactions are encrypted, records are immutable—so any changes to the ledger can be recognized by the network and rejected. These blocks of encrypted data are permanently "chained" to one another, and transactions are recorded sequentially and indefinitely, creating a perfect audit history that allows visibility into past versions of the blockchain.

When new data is added to the network, the majority of nodes must verify and confirm the legitimacy of the new data based on permissions or economic incentives, also known as consensus mechanisms. When a consensus is reached, a new block is created and attached to the chain. All nodes are then updated to reflect the blockchain ledger.In a public blockchain network, the first node to credibly prove the legitimacy of a transaction receives an economic incentive. This process is called "mining."

Here's a theoretical example to help illustrate how blockchain works. Imagine that someone is looking to buy a concert ticket on the resale market. This person has been scammed before by someone selling a fake ticket, so she decides to try one of the blockchain-enabled decentralized ticket exchange websites that have been created in the past few years.

On these sites, every ticket is assigned a unique, immutable, and verifiable identity that is tied to a real person. Before the concertgoer purchases her ticket, the majority of the nodes on the network validate the seller's credentials, ensuring that the ticket is in fact real. She buys her ticket and enjoys the concert.

SECURITY IN BLOCKCHAIN

Blockchain has been called a “truth machine.” While it does eliminate many of the issues that arose in Web 2.0, such as piracy and scamming, it’s not the be-all and endall for digital security. The technology itself is essentially foolproof, but, ultimately, it is only as noble as the people using it and as good as the data they are adding to it.

A motivated group of hackers could leverage blockchain’s algorithm to their advantage by taking control of more than half of the nodes on the network. With this simple majority, the hackers have consensus and thus the power to verify fraudulent transactions.

In 2022, hackers did exactly that, stealing more than $600 million from the gamingcentered blockchain platform Ronin Network. This challenge, in addition to the obstacles regarding scalability and standardization, will need be addressed. But there is still significant potential for blockchain, both for business and society

USAGE OF BLOCKCHAIN

Cryptocurrency is only the tip of the iceberg. Use cases for blockchain are expanding rapidly beyond person-to-person exchanges, especially as blockchain is paired with other emerging technology.

Examples of other blockchain use cases include the following:

With blockchain, companies can create an indelible audit trail through a sequential and indefinite recording of transactions. This allows for systems that keep static records (of land titles, for example) or dynamic records (such as the exchange of assets).

Blockchain allows companies to track a transaction down to its current status. This enables companies to determine exactly where the data originated and where it was delivered, which helps to prevent data breaches.

Blockchain supports smart contracts, which are programs that trigger transactions automatically upon fulfillment of contract criteria.

Advantages Of Blockchain

Immutability. Blockchain supports immutability, meaning it is impossible to erase or replace recorded data. Therefore, the blockchain prevents data tampering within the network.

Traditional data do not exhibit immutability. The conventional database uses CRUD (create, read, update and delete) at the primary level to ensure proper application operation, and the CRUD model enables easy erasing and replacing of data. Such data can be prone to manipulation by rogue administrators or third-party hacks.

Transparency: Blockchain is decentralized, meaning any network member can verify data recorded into the blockchain. Therefore, the public can trust the network.

On the other hand, a traditional database is centralized and does not support transparency. Users cannot verify information whenever they want, and the administration makes a selected set of data public. Still, however, individuals cannot verify the data.

Censorship: Blockchain technology is free from censorship since it does not have control of any single party. Therefore, no single authority (including governments) can interrupt the operation of the network.

Meanwhile, traditional databases have central authorities regulating the operation of the network, and the authority can exercise censorship. For instance, banks can suspend users' accounts.

Traceability. Blockchain creates an irreversible audit trail, allowing easy tracing of changes on the network. The traditional database is neither transparent nor immutable; hence, no permanent trail is guaranteed.

Disadvantages Of Blockchain

Speed and performance. Blockchain is considerably slower than the traditional database because blockchain technology carries out more operations. First, it performs signature verification, which involves signing transactions cryptographically. Blockchain also relies on a consensus mechanism to validate transactions.

Some consensus mechanisms, such as proof of work, have a low transaction throughput. Finally, there is redundancy, where the network requires each node to play a crucial role in verifying and storing each transaction.

High implementation cost. Blockchain is costlier compared to a traditional database. Additionally, businesses need proper planning and execution to integrate blockchain into their process.

Data modification. Blockchain technology does not allow easy modification of data once recorded, and it requires rewriting the codes in all of the blocks, which is timeconsuming and expensive. The downside of this feature is that it is hard to correct a mistake or make any necessary adjustments.

INTRODUCTION TO PYTHON

Guido Van Rossum published the first version of Python code (version 0.9.0) at alt. sources in February 1991. This release included already exception handling, functions, and the core data types of list, dict, str and others. It was also object oriented and had a module system.

Python version 1.0 was released in January 1994. The major new features included in this release were the functional programming tools lambda, map, filter and reduce, which Guido Van Rossum never liked. Six and a half years later in October 2000, Python 2.0 was introduced. This release included list comprehensions, a full garbage collector and it was supporting unicode. Python flourished for another 8 years in the versions 2.x before the next major release as Python 3.0 (also known as "Python 3000" and "Py3K") was released. Python 3 is not backwards compatible with Python 2.x. The emphasis in Python 3 had been on the removal of duplicate programming constructs and modules, thus fulfilling or coming close to fulfilling the 13th law of the Zen of Python: "There should be one -- and preferably only one -- obvious way to do it."Some changes in Python 7.3:

Purpose

We demonstrated that our approach enables successful segmentation of intraretinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

MODULES USED IN PROJECT

Tkinter is the most commonly used method. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter is the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.
To create a tkinter :

Importing the module – tkinter

Create the main window (container)

Add any number of widgets to the main window

Apply the event Trigger on the widgets.

Importing tkinter is same as importing any other module in the Python code. Note that the name of the module in Python 2.x is 'Tkinter' and in Python 3.x it is 'tkinter'. import tkinter

There are two main methods used which the user needs to remember while creating the Python application with GUI.

Tk(screenName=None, baseName=None, className='Tk', useTk=1): To create a main window, tkinter offers a method 'Tk(screenName=None, baseName=None, className='Tk', useTk=1)'. To change the name of the window, you can change the className to the desired one. The basic code used to create the main window of the application is:

m=tkinter.Tk() where m is the name of the main window object

mainloop(): There is a method known by the name mainloop() is used when your application is ready to run. mainloop() is an infinite loop used to run the application, wait for an event to occur and process the event as long as the window is not closed. m.mainloop()

INSTALLATION OF PYTHON

Python a versatile programming language doesn't come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.

The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Windows.

How to Install Python on Windows and Mac

There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your System Requirements. Based on your system type i.e., operating system and based processor, you must download the python version. My system type is a Windows 64-bit operating system. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheat sheet here. The steps on how to install Python for Windows 10, 8 and 7 are divided into 4 parts to help understand better.

Download the Correct version into the system

Step 1: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: https://www.python.org

Now, check for the latest and the correct version for your operating system.

Step 2: Click on the Download Tab.

Step 3: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4 Step 4: Scroll down the page until you find the Files option.

Step 5: Here you see a different version of python along with the operating system.

To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.

Note: To know the changes or updates that are made in the version you can click on the Release Note Option.

Installation of Python

Step 1: Go to Download and Open the downloaded python version to carry out the installation process.

Step 2: Before you click on Install Now, Make sure to put a tick on Add Python 3.7 to PATH.

Step 3: Click on Install NOW After the installation is successful. Click on Close.

With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

7. IMPLEMENTATION

ALGORITHM USED

Blockchain Hash Function

A hash function is a mathematical function that takes an input string of any length and converts it to a fixed-length output string. The fixed-length output is known as the hash value.

Working of Hash function

The hash function takes the input of variable lengths and returns outputs of fixed lengths. In cryptographic hash functions, the transactions are taken as inputs and the hash algorithm gives an output of a fixed size. The below diagram shows how hashes work

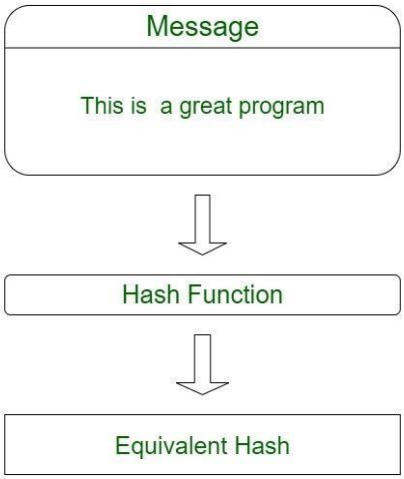

Fig. 7.1 working of hash

Secure Hashing Algorithm

- The family of SHA comprises four SHA algorithms: SHA-0, SHA-1, SHA-2, and SHA-3.

- SHA-0 is a 160-bit hash function that was published by the National Institute of
  Standards and Technology in 1993.
- SHA-2 has the following SHA variants, SHA-224, SHA-256, SHA-384, SHA-512, SHA-512/224, and SHA-512/256. It is a stronger hash function and it still follows the design of SHA-1.
- In 2012, the Keccak algorithm was chosen as the new SHA-3 standard.
- SHA-256 is the most famous of all cryptographic hash functions because it's used extensively in blockchain technology. The SHA-256 Hashing algorithm was developed by the National Security Agency (NSA) in 2001.

MODULES

- **Save Product with Blockchain Entry:** In this module user will enter product details and then upload product bar code image and then digital signature will be generated on uploaded barcode and then this transaction details will be store in Blockchain. Before storing transaction Blockchain will verify all old transaction and upon successful verification new transaction block will be store.
- **Retrieve Product Data:** Using this module user can search existing product details by entering product id
- **Authenticate Scan:**Here in this module we don't have any scanner so we are uploading original or fake bar code images and then Blockchain will verify digital signature of uploaded bar code with already store bar codes and if match found then Blockchain will extract all details and display to user else authentication will be failed.
-

8. CODE

```
 import tkinter as tk from
tkinter import messagebox
from tkinter import *
import threading import
datetime import random
import os import pickle
from hashlib import
sha256 import imageio
import PIL.Image,
PIL.ImageTk from PIL
import Image, ImageTk
import pyqrcode import
Blockchain from Block
import * from Blockchain
import * from hashlib
import sha256 import
```

```
base64 from datetime
import datetime


main = tk.Tk() main.title("Safeguarding against counterfeits
using blockchain-based qr code") main.attributes('-fullscreen',
True)
#main.geometry('1300x1200')

global filename

blockchain = Blockchain() if
os.path.exists('blockchain_contract.txt'):
with open('blockchain_contract.txt', 'rb') as
fileinput:          blockchain =
pickle.load(fileinput)


def addProduct():      global filename
text.delete('1.0', tk.END)      pid = tf1.get()
name = tf2.get()      user = tf3.get()
address = tf4.get()      shreya =
hex(random.getrandbits(128))      bytes_data
= shreya.encode('utf-8')      digital_signature
= sha256(bytes_data).hexdigest()

    logo = Image.open('bg/logo.jpg')      basewidth
= 100      wpercent = (basewidth /
float(logo.size[0]))      hsize = int((float(logo.size[1])
* float(wpercent)))      logo =
logo.resize((basewidth, hsize), Image.LANCZOS)

    QRcode = pyqrcode.create(digital_signature)
```

```
    QRcode.png(f'original_barcodes/{pid}productQR.png', scale=6)

    if len(pid) > 0 and len(name) > 0 and len(user) > 0 and len(address)
> 0:
        current_time                    =
datetime.now()          data =
f"{pid}#{name}#{user}#{address}#{current_time}#{digital_signa
ture}"          blockchain.add_new_transaction(data)
hash_val = blockchain.mine()          block =
blockchain.chain[-1]          text.insert(tk.END, f"Blockchain
Previous Hash         :
{block.previous_hash}\n")
        text.insert(tk.END, f"Block No                    : {block.index}\n")
text.insert(tk.END, f"Product QR-code no          : {digital_signature}\n")
blockchain.save_object(blockchain, 'blockchain_contract.txt')          img2 =
Image.open(f'original_barcodes/{pid}productQR.png')          load =
img2.resize((200, 200))          render = ImageTk.PhotoImage(load)          img
= tk.Label(main, image=render)          img.image = render  # to keep
reference          img.place(x=140, y=500)

        tf1.delete(0, 'end')
tf2.delete(0, 'end')
tf3.delete(0, 'end')
tf4.delete(0, 'end')
        messagebox.showinfo("QR  Code  Generator",  "QR
Code is saved successfully!")      else:
        text.insert(tk.END, "Please enter all details")


def searchProduct():
text.delete('1.0', END)
pid = tf1.get()
```

```
    flag = True
if len(pid) > 0:
        for i in
range(len(blockchain.chain)):
if i > 0:
            b = blockchain.chain[i]
data = b.transactions[0]
arr = data.split("#") if
arr[0] == pid: logo =
Image.open('bg/logo.jpg')
basewidth = 100
            wpercent = (basewidth/float(logo.size[0]))
hsize = int((float(logo.size[1])*float(wpercent)))
logo = logo.resize((basewidth, hsize), Image.LANCZOS)
            QRcode = pyqrcode.create(arr[5])
            QRcode.png('original_barcodes' + '\\' + str(pid) +
'productQR.png', Scale=6)
            output = ''
            text.insert(END, "Product Details extracted from
Blockchain using Product ID : " + pid + "\n\n")
            text.insert(END, "Product ID :
" + arr[0] +
"\n") text.insert(END, "Product Name
: " + arr[1] + "\n") text.insert(END, "Company/User
Details : " + arr[2] +
"\n") text.insert(END, "Address Details
: " + arr[3] + "\n") text.insert(END, "Product
Registered Date & Time : " + arr[4]
+ "\n") text.insert(END, "Product Qr Code :
" + arr[5] + "\n") output = '<html><body><table border=1>'
            output += '<tr><th>Block No</th><th>Product
ID</th><th>Product
```

```
Name</th><th>Company/User Details</th><th>Address Details</th><th>Scan Date & Time</th><th>Product Qr code</th></tr>'
                output += '<tr><td>' + str(i) + '</td><td>' + arr[0] + '</td><td>' + arr[1] + '</td><td>' + arr[2] + '</td><td>' + arr[3] + '</td><td>' + arr[4] +
'</td><td>' + arr[5] + '</td></tr>'
f = open("output.html", "w")
f.write(output)
                f.close()
flag = False
                img2 =
Image.open(f'original_barcodes/{pid}productQR.png')
load = img2.resize((200, 200))                render =
ImageTk.PhotoImage(load)                img = tk.Label(main,
image=render)
                img.image = render  # to keep
reference                img.place(x=140,
y=500)                break     if flag:
        text.insert(END, "Given product id does not exist")

main.wm_attributes('-transparentcolor', '#ab23ff') font = ('times',
30, 'bold') title = Label(main, text='Safeguarding against
counterfeits using blockchain-based qr code') title.config(bg='white',
fg='black') title.config(font=font) title.config(height=3, width=50)
title.place(x=170,y=5)

font1 = ('times', 13, 'bold')

l1 = Label(main, text='Product ID :')
l1.config(font=font1,bg="black",fg="white")
l1.place(x=280,y=200)
```

```
tf1 = Entry(main,width=80)
tf1.config(font=font1,highlightbackground="black",highlightthickness=1)
tf1.place(x=470,y=200)

l2 = Label(main, text='Product Name :')
l2.config(font=font1,bg="black",fg="white")
l2.place(x=280,y=250)

tf2 = Entry(main,width=80)
tf2.config(font=font1,highlightbackground="black",highlightthickness=1)
tf2.place(x=470,y=250)

l3 = Label(main, text='Company/User Details :')
l3.config(font=font1,bg="black",fg="white")
l3.place(x=280,y=300)

tf3 = Entry(main,width=80)
tf3.config(font=font1,highlightbackground="black",highlightthickness=1)
tf3.place(x=470,y=300)

l4 = Label(main, text='Address Details :')
l4.config(font=font1 ,bg="black",fg="white")
l4.place(x=280,y=350)

tf4 = Entry(main,width=80)
tf4.config(font=font1,highlightbackground="black",highlightthickness=1)
tf4.place(x=470,y=350)

def run13():
main.destroy()
import Main
```

```
    #os.system('AdmMain.py',)


scanButton = Button(main, text="Home
Page",bg="dark orange", command=run13)
scanButton.place(x=1400,y=200)
scanButton.config(font=font1)

saveButton = Button(main,bg="green", text="Save Product
with Blockchain Entry", command=addProduct)
saveButton.place(x=420,y=400) saveButton.config(font=font1)

searchButton = Button(main, text="Retrieve Product
Data", command=searchProduct,bg="green")
searchButton.place(x=850,y=400)
searchButton.config(font=font1)


font1 = ('times', 13, 'bold') text=Text(main,height=15,width=100)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set,highlightbackground="bla
ck",highlightt hickness=1) text.place(x=400,y=450)
text.config(font=font1)

main.config(bg='white')
main.mainloop()
```

9. RESULTS:

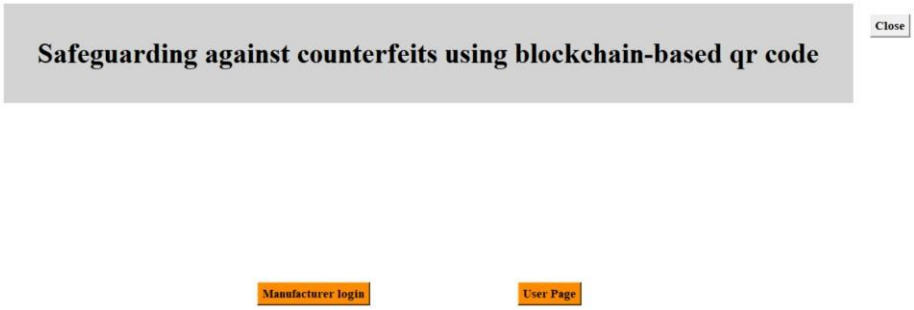

Figure 9.1: Home page

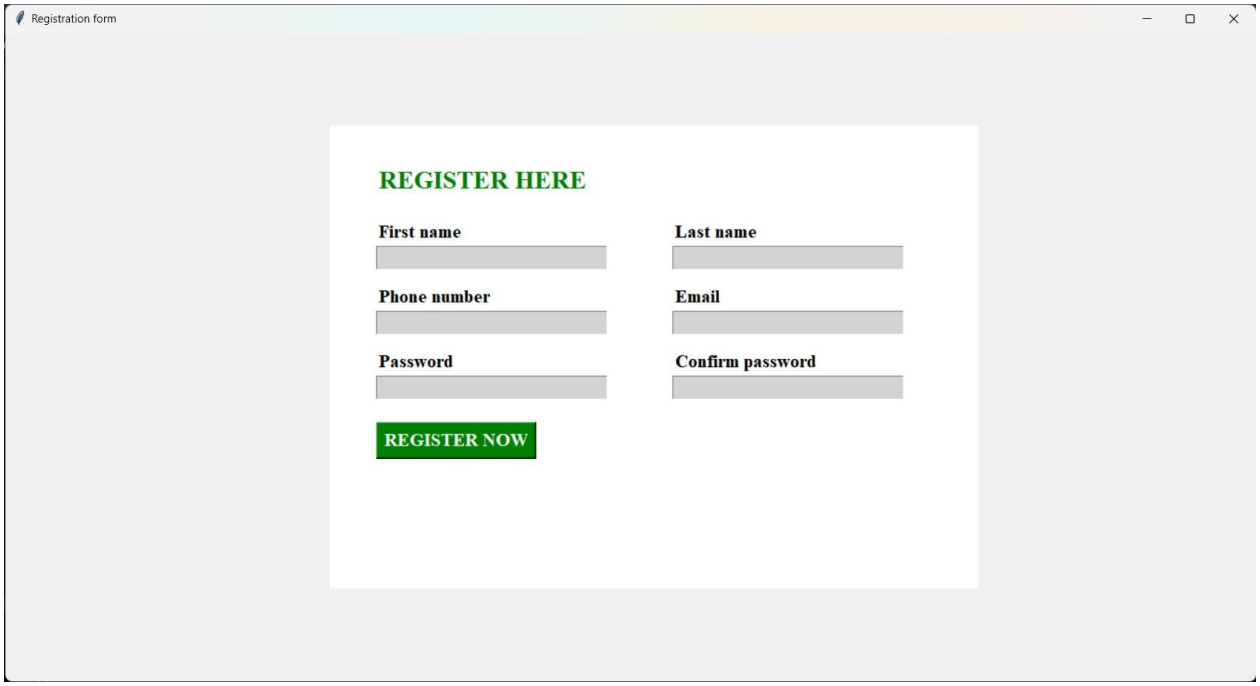

Figure 9.2: Manufacturer Registration page

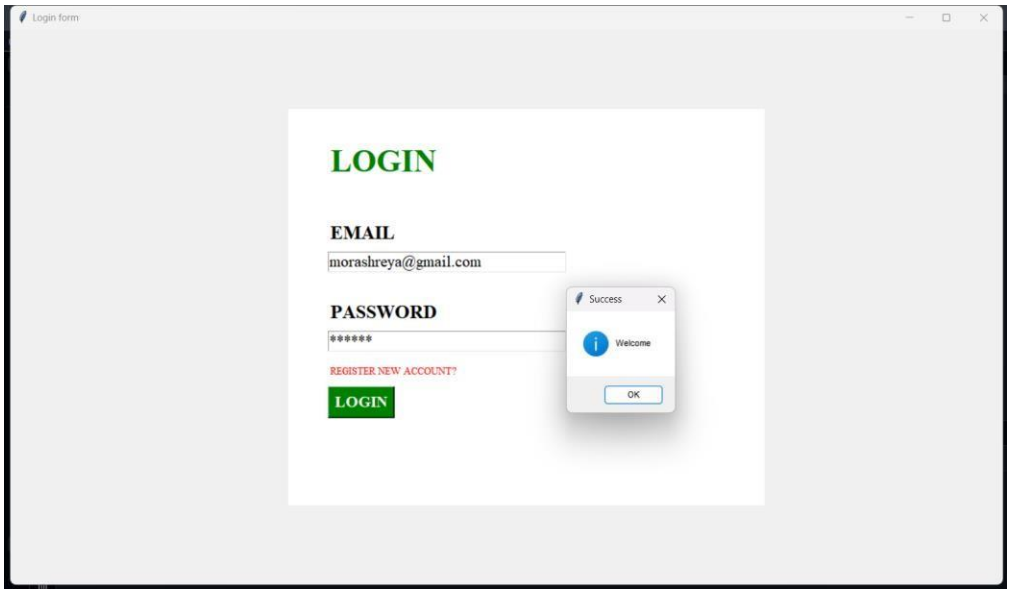

Figure 9.3: Manufacturer Login Page

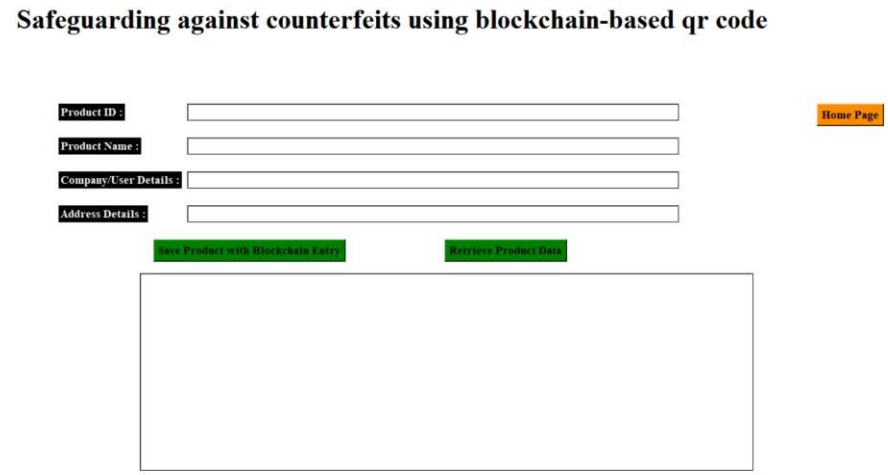

Figure 9.4: Manufacturer Page

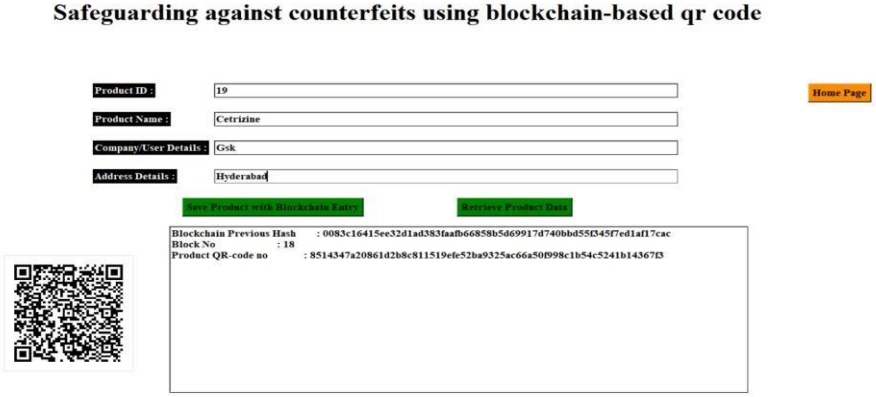

Figure 9.5: Saving Products to Blockchain

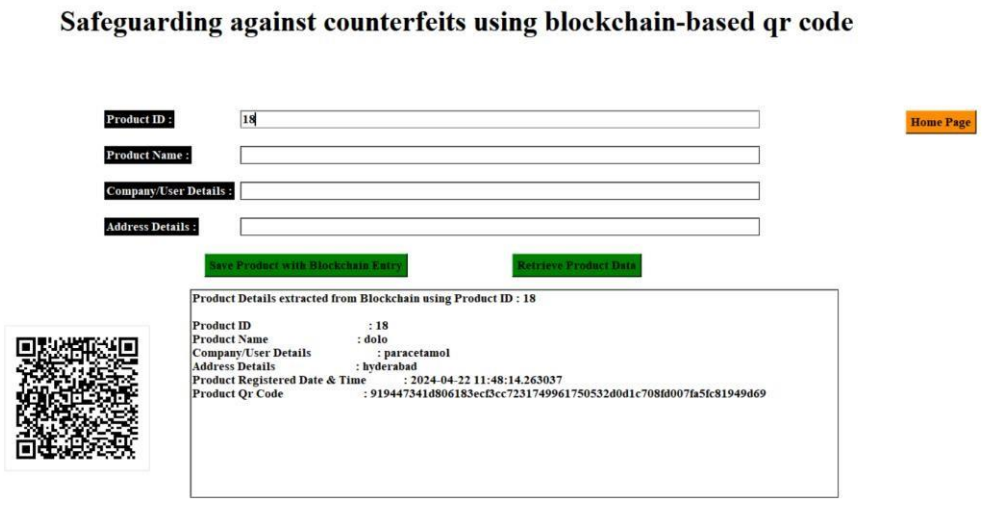

Figure 9.6: Retrieving Products

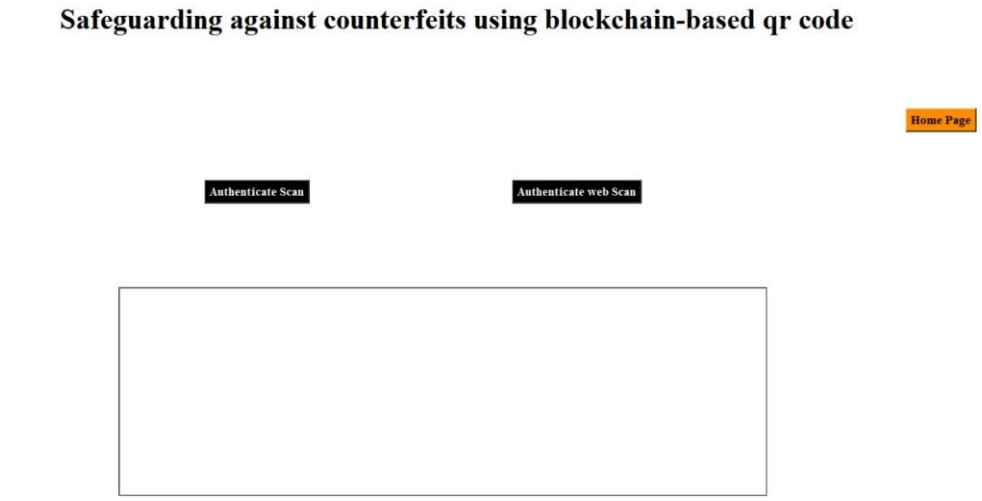

Figure 9.7: User Page

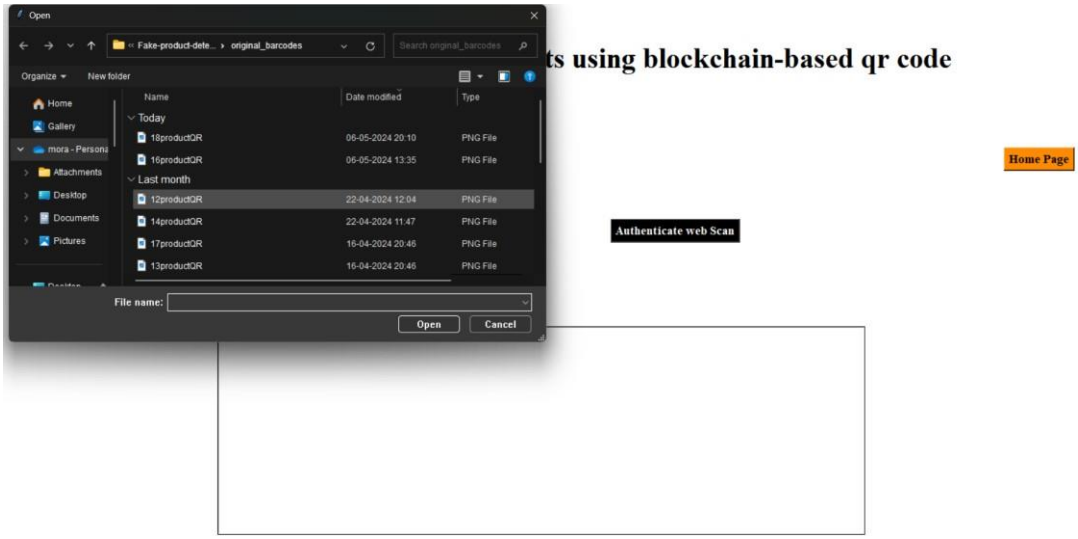

Figure 9.8: Authenticate Product QR Code

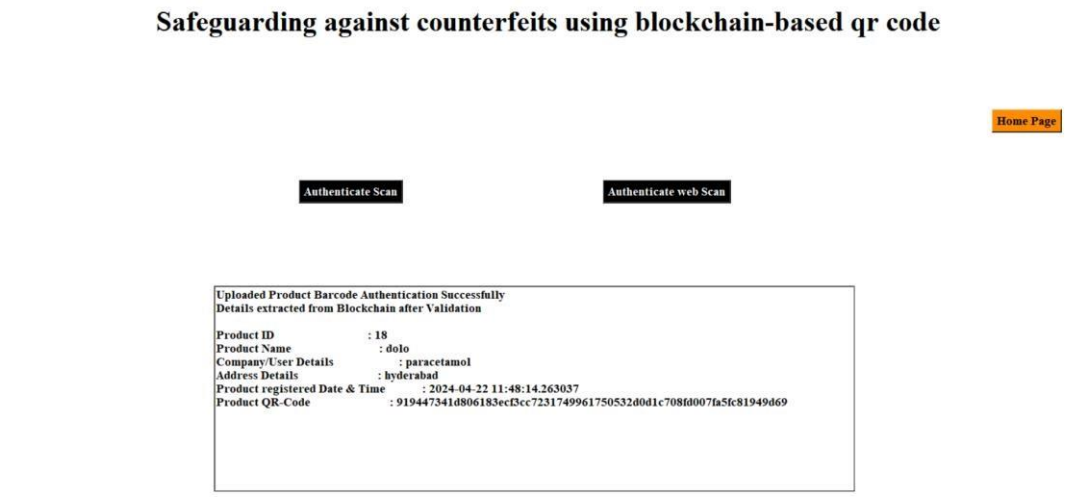

Figure 9.9: Product Details

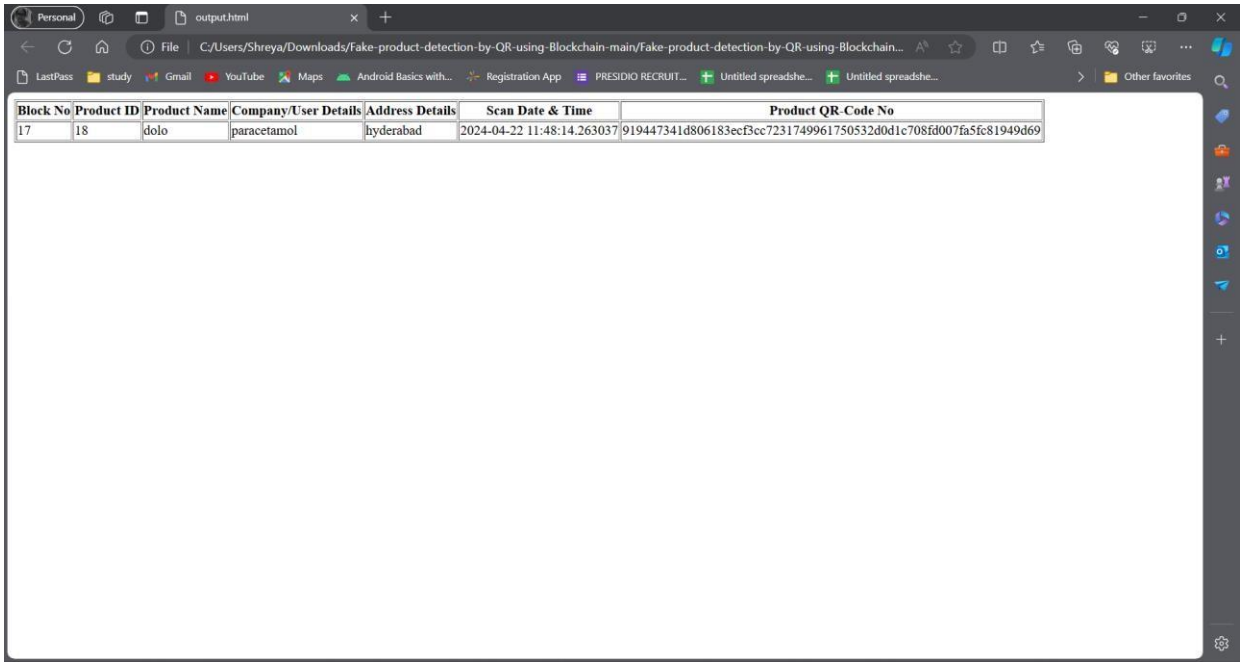

Figure 9.10: Output Page

10. CONCLUSION

With this system, the products journey from manufacturing to customer can be recorded, and the customer is assured that the scans weren‘t faked. Manufacture is able to prove their product is authentic and is also able to track their product ‘s pathway. The setup is easy to implement and requires less operation cost. Manufacturer can also adopt RFID (Radio Frequency Identification) or NFC (Near Field Communication) tokens instead of QR codes to further strengthen their system.

11. FUTURE SCOPE/ENHANCEMENT

Multiple techniques to reducing counterfeits were examined in this thesis. These improvements were considered, and their impact on minimising counterfeits was assesed, in order to be less reliant on external variables. Due to time constraints and the fact that several other system changes were also required, it was not possible to implement all of the suggested changes. The finalisation of these implementations for the proposed system, as well as the potential of running pilots, are among the next steps. The concept for reducing counterfeits in the humanitarians supply chain is currently being developed, as is the execution.

12. BIBLIOGRAPHY

[1]Manoj malik-Authentication of products and counterfeit elimination using blockchain based QR code

[2] Satoshi Nakamoto, ―Bitcoin: A Peer-to-Peer Electronic Cash System‖, 2008

[3] Hyperledger, ―Hyperledger Blockchain Performance Metrics‖, V1.01, October 2018

[4] R.C. Merkle, "Protocols for public key cryptosystems," In Proc. 1980 Symposium on Security and Privacy, IEEE Computer Society, pages 122-133, April 1980.

[5] G. Wood, __Ethereum: A secure decentralised generalized transaction ledger,“ Tech. Rep., 2014.

[6] OECD (2016), Illicit Trade: Converging Criminal Networks, OECD Reviews of Risk Management Policies, OECD Publishing, Paris, https://doi.org/10.1787/9789264251847-en.

[7] M. Castro and B. Liskov, __Practical byzantine fault tolerance and proactive recovery,“ ACM Trans. Comput. Syst., vol. 20, no. 4, pp. 398–461, Nov. 2002.

[8] Clement, E. Wong, L. Alvisi, M. Dahlin, and M. Marchetti, __Making byzantine fault tolerant systems tolerate byzantine faults,“ in Proc. 6th USENIX Symp. Netw. Syst. Design Implement., 2009, pp. 153–168.

[9] Cachin, __Architecture of the hyperledger blockchain fabric,“ Tech. Rep., Jul. 2016..

[10] S. Underwood, ―Blockchain Beyond Bitcoin‖, in Communications of the ACM, vol. 59, no. 11, p. 15-17, 2016.

[11] Deloitte, Israel: A Hotspot for Blockchain Innovation, 2016. [Online]. Available: https://www2.deloitte.com/content/dam/ Deloitte/il/Documents/financialservices/israel_a_hotspot_for_blockchain _innovation_ feb2016_1.1.pdf. [Accessed: 2.11.2016].

[12] G. Greenspan and M. Zehavi, Will Provenance Be the Blockchain‘s Break Out Use Case in 2016?, 7.1.2016. [Online]. Available:

http://www.coindesk.com/ provenance-blockchaintech-app/. [Accessed: 12.12.2016].

[13] Counterfeit medicines. QA counterfeit. World Health Organization (WHO) 2009. Available from: http://www.who.int/medicines/ services/counterfeit/faqs/QACounterfeitoctober2009.pdf [last cited on 2010 Jun 12].