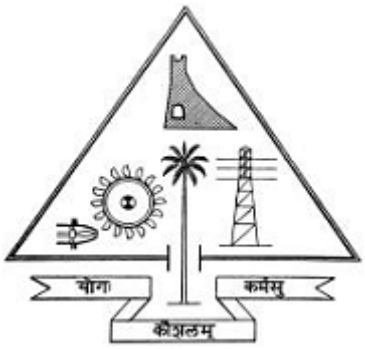
BLOCK CHAIN BASED COUNTERFEIT PRODUCT DETECTION

Project submitted in partial fulfillment of the requirements for the award of the degree of ***Master of Computer Applications*** *of the* ***APJ Abdul Kalam Technological University***

submitted by

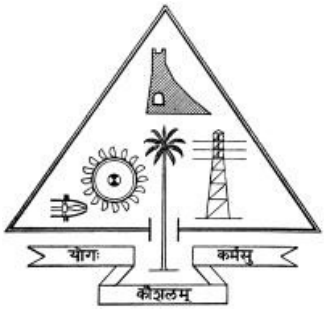
AMINA FIDHA
(TCR19MCA002)



DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE
THRISSUR - 680009

MAY 2022

DEPARTMENT OF COMPUTER APPLICATIONS
GOVERNMENT ENGINEERING COLLEGE, THRISSUR
THRISSUR, KERALA STATE, PIN 680009



CERTIFICATE

This is to certify that the Project titled ***"BLOCK CHAIN BASED COUNTERFEIT PRODUCT DETECTION"*** *is a bonafide work done by* ***AMINA FIDHA (TCR19MCA002)*** *under my supervision and guidance, and is submitted in May 2022 in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications from APJ Abdul Kalam Technological University(KTU).*

Jibin Thomas	Soumia Chandran M	Dr. Sminesh C N
Project Guide	**Project Coordinator**	**Head of Department**

Place : THRISSUR
Date : 13-05-2022

External Supervisor

DECLARATION

I hereby declare that the main project named, **BLOCK CHAIN BASED COUNTERFEIT PRODUCT DETECTION**, is my own work and that, to the best of my knowledge and belief, it contains no material previously published another person nor material which has been accepted for the award of any other degree or course of the university or any other institute of higher learning, except where due acknowledgement and reference has been made in the text.

Signature

AMINA FIDHA
(TCR19MCA002)

Place : THRISSUR
Date : 13-05-2022

ACKNOWLEDGEMENT

I would like to thank Department of Computer Applications of GEC Thrissur, for giving us this opportunity to pursue this project and successfully complete it.

I are highly indebted to our head of department,**Dr. Sminesh C N** for his guidance and constant supervision as well as for providing necessary information regarding the project and also for his support in completing the project.

I express our heart-felt gratitude to project guide,**Asst Prof. Jibin Thomas** , for his committed guidance, valuable suggestions, constructive criticisms and precious time that he invested throughout the work.

I also express special thanks to project coordinator,**Prof. Soumia Chandran M**, for his keen support and consistent encouragement in our academic activities.

I sincerely thank all other faculties of Department of Computer Applications for guiding through the processes involved in the project .

ABSTRACT

In recent years, Counterfeit goods play a vital role in product manufacturing industries. This Phenomenon affects the sales and profit of the companies. To ensure the identification of real products throughout the supply chain, a functional block chain technology used for preventing product counterfeiting.

By using a block chain technology, consumers do not need to rely on the trusted third parties to know the source of the purchased product safely. Any application that uses block chain technology as a basic framework ensures that the data content is 'tamper resistant'. In view of the fact that a block chain is the decentralized, distributed and digital ledger that stores transactional records known as blocks of the public in several databases known as chain across many networks. Therefore, any involved block cannot be changed in advance, without changing all subsequent block.

In this paper, counterfeit products are detected using QR code reader, where a QR code of the product linked to a Block Chain Based Management (BCBM) system. So the proposed system may be used to store product details and unique code of that product as blocks in database. It collects the unique code from the customer and compares the code against entries in block chain database. If the code matches, it will give notification to the customer, otherwise it gets information from the customer about where they bought the product to detect counterfeit product manufacturer.

CONTENTS

LIST OF FIGURES

NOMENCLATURE

BCBM	Block Chain Based Management
DFD	Data Flow Diagram
POM	Product Ownership Management
RDBMS	Relational Database Management System
RFID	Radio Frequency Identification

CHAPTER 1
INTRODUCTION

A block chain technology is a digital ledger of record of transactions, which based on decentralized network in a peer-to-peer network around the world. Block chain, is focused primarily online transactions and disseminated digital ledger system. In this proposed framework, we have proposed a BCBM system that will find counterfeit products in supply chain.

Block chains consist of blocks or data records and each block is linked to the next block in an irreversible chain and the transactions are gathered as a block –hence the term "block chain". While comparing with traditional centralized database, the information cannot be handled due to block chain's feature of distributed nature and confirmed guarantees by the peers. If someone wants to perform a transaction, it goes to the network directly and algorithms find the authenticity of the transaction. After the transaction is verified, the new one is linked with the previous transaction forming a chain of transactions. This chain is called the block chain.

The agreement between two people in the form of a computer code is called as Smart Contract. They run on the block chain, so they are stored on a public database and cannot be changed. The combination of block chain technology and smart contract gives more flexibility to design, develop and implement in real-time with minimum cost. Block chain-based smart contracts provide number of advantages: they are quick and real-time refurbish, minimum cost and lower risk in execution, no intermediaries and high accuracy.

Nowadays, small-sized enterprises often have financial burdens, which never compare with large companies with strong financial resources. In the

brand management sector, small-sized enterprises will unavoidably need to reduce costs and will be most likely unable to prevent counterfeited goods. To ensure the identification and traceability of real products throughout the supply chain, the paper is to propose a block chain management system to prevent product counterfeiting and Product Authentication Using QR (Quick Response) Code by manufacturer. QR codes provide a robust mechanism for combating counterfeit practice with products. Encrypted QR codes are mainly used in security and proprietary applications. Benefits of block chain in the supply chain are increasing transparency, scalability, providing security and increasing innovative technologies. Block Chain Technology provides secure business operations in logistics and reduces added costs, human errors and time delays.

CHAPTER 2
LITERATURE SURVEY

2.1 BACKGROUND STUDY

The supply chain management system consists of several stages and various sectors of applications in supply chain. Block chain system makes supply chain sector more reliable.

Authors proposed POM system, with each stage of supply chain in need of transferring and proving ownership of RFID tag. If the seller does not possess their ownership, consumers can reject the purchase of counterfeit products even with a genuine product code. This technique can discover several fake products such an approach is vulnerable against identical tags. Once the attacker copies the RFID tags attached to the authentic product, and then this counterfeit duplicate label is inserted in the supply chain.

Author proposed a decentralized Block chain technology approach to make sure that users do not trust on the sellers to find if products are real. Therefore, manufacturer can use this system to provide real products without managing the stores, which operated directly. It can considerably reduce the cost and product quality assurance. The system can effectively lower the threshold of the genuine products and provide with limited financial resources to the companies. It is also an easier approach to provide consumers with the confidence that they will not purchase fake products. However, there is no code simplicity and redundancy.

In The base paper ' A Block Chain based Management System for Detecting Counterfeit Product in Supply Chain ' , we propose a block chain management system to prevent product counterfeiting and product authen-

tication using QR code by the manufacturer. Encrypted QR codes ensure identification and traceability of real products throughout the supply chain.

CHAPTER 3
ENVIRONMENTAL STUDY

3.1 System Configuration

System configuration describe the hardware and software requirement of the system for development

3.1.1 Hardware Requirements

- Memory : 8 GB of RAM
- Processor : Intel Core i3 or equivalent CPU and GPU enabled system
- Speed : 2.4 GHz
- Proper Internet Connection

3.1.2 Software Requirements

- Operating system : Windows
- Front End : Html, css, javascript, Java
- IDE Used : Pycharm, Android Studio
- Back End : Mysql

3.2 Technology Specification

3.2.1 PYTHON

Python is an interpreted, high-level, general-purpose programming language. Python's design philosophy emphasizes code readability with its

notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed AND supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

3.2.2 JAVASCRIPT,CSS

JavaScript is used to create interaction between webpages and the user. CSS: CSS stands for Cascading Style Sheet, it is a style sheet language used to shape the HTML elements that will be displayed in the browsers as a web-page. Without using CSS, the website which has been created by using HTML will look dull.

3.2.3 Solidity

Solidity is an object-oriented, high-level language for implementing smart contracts. Smart contracts are programs which govern the behaviour of accounts within the Ethereum state. Solidity is a curly-bracket language designed to target the Ethereum Virtual Machine (EVM). It is influenced by C++, Python and JavaScript.
Solidity is statically typed, supports inheritance, libraries and complex user-defined types among other features. With Solidity you can create contracts for uses such as voting, crowdfunding, blind auctions, and multi-signature wallets. When deploying contracts, you should use the latest released version of Solidity. Apart from exceptional cases, only the latest version receives security fixes. Furthermore, breaking changes as well as new features are introduced regularly.

3.2.4 PyCharm

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python programming language. It is developed by the Czech company JetBrains (formerly known as IntelliJ).

It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda.
PyCharm is cross-platform, with Windows, macOS and Linux versions. The Community Edition is released under the Apache License, and there is also an educational version, as well as a Professional Edition with extra features (released under a subscription-funded proprietary license)

3.2.5 Node.js

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser. Consequently, Node.js represents a "JavaScript everywhere" paradigm, unifying web-application development around a single programming language, rather than different languages for server-side and client-side scripts.

Node.js has an event-driven architecture capable of asynchronous I/O. These design choices aim to optimize throughput and scalability in web applications with many input/output operations, as well as for real-time Web applications (e.g., real-time communication programs and browser games).

3.2.6 NPM

npm (originally short for Node Package Manager) is a package manager for the JavaScript programming language maintained by npm, Inc. npm is the default package manager for the JavaScript runtime environment Node.js. It consists of a command line client, also called npm, and an online database of public and paid-for private packages, called the npm registry. The registry is accessed via the client, and the available packages can be browsed and searched via the npm website. The package manager and the registry are managed by npm, Inc.

3.2.7 TRUFFLE

Sweet Tools for Smart Contracts The Truffle Suite gets developers from idea to dapp as comfortably as possible.

TRUFFLE SMART CONTRACTS MADE SWEETER Truffle is the most popular development framework for Ethereum with a mission to make your life a whole lot easier. INSTALLATION STATEMENT OF TRUFFLE USING NPM npm install truffle -g

h. GANACHE - Quickly fire up a personal Ethereum blockchain which you can use to run tests, execute commands, and inspect state while controlling how the chain operates.

* BLOCKCHAIN LOG OUTPUT - See the log output of Ganache's internal blockchain, including responses and other vital debugging information.

* ADVANCED MINING CONTROLS - Configure advanced mining with a single click, setting block times to best suit your development needs.

* BUILT-IN BLOCK EXPLORER - Examine all blocks and transactions to gain insight about what's happening under the hood.

* THE ETHEREUM BLOCKCHAIN - Byzantium comes standard, giving you the latest Ethereum features needed for modern dapp development.

i.Visual Studio 2022 Version 17.1-The best comprehensive IDE for .NET and C++ developers on Windows. Fully packed with a sweet array of tools and features to elevate and enhance every stage of software development.The desktop c++ build tools are essential to install web3 python package.

3.2.8 WEB3

Web3 (also known as Web 3.0 and sometimes stylized as web3) is an idea for a new iteration of the World Wide Web based on blockchain technology, which incorporates concepts such as decentralization and token-based economics. Some technologists and journalists have contrasted it with Web 2.0, wherein they say data and content are centralized in a small group of companies sometimes referred to as ”Big Tech”. The term ”Web3” was coined

in 2014 by Ethereum co-founder Gavin Wood, and the idea gained interest in 2021 from cryptocurrency enthusiasts, large technology companies, and venture capital firms.
Some experts argue that web3 will provide increased data security, scalability, and privacy for users and combat the influence of large technology companies. Others have raised concerns about a decentralized web, citing the potential for low moderation and the proliferation of harmful content, the centralization of wealth to a small group of investors and individuals, or a loss of privacy due to more expansive data collection. Others, such as Elon Musk and Jack Dorsey, have argued that web3 only currently serves as a buzzword. It act as intermediate between forms and blockchain decentralized peer to peer network as well as storage.

3.2.9 Flask

Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools. Applications that use the Flask framework include Pinterest and LinkedIn.

3.2.10 MySQL

MySQL is an open-source relational database management system. Its name is a combination of ”My”, the name of co-founder Michael Widenius’s daughter, and ”SQL” the abbreviation of Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data

from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an operating system to implement a relational database in a computer's storage system, manage users, allows for network access and facilitates testing database integrity and creation of backups.

3.2.11 Android Studio

Android Studio is the official IDE for Google's Android operating system, built on JetBrains' Intellij IDEA software and designed for Android development. It is available for download on Windows, macOS and Linux based operating systems. It is a replacement for the Eclipse Android Development Tools as the primary IDEa for native Android application development.

CHAPTER 4
SYSTEM ANALYSIS

System Analysis by definition is a process of systematic investigation for the purpose of gathering data, interpreting the facts, diagnosing the problem and using this information to either build a completely new system or to recommend the improvements to the existing system.

A satisfactory system analysis involves the process of examining a business situation with the intent of improving it through better methods and procedures. In its core sense, the analysis phase defines the requirements of the system and the problems which user is trying to solve irrespective of how the requirements would be accomplished.

4.1 Requirements Analysis

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. These features, called requirements, must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.

4.1.1 Python 3

Python was developed by Guido van Rossum in the early 1990s and its latest version is 3.7.1, we can simply call it as Python3. Python 3.0 was released in 2008. and is interpreted language i.e it's not compiled and the interpreter will check the code line by line. This article can used to learn very basics of Python programming language.Python is a general-purpose interpreted, interactive,

object-oriented, and high-level programming language.Like Perl, Python source code is also available under the GNU General Public License (GPL). Python is named after a TV Show called ëMonty Pythonís Flying Circusí and not after Python-the snake.

4.1.2 Libraries:

• mysql-connector : Python installers in native package formats are available for Windows and for Unix and Unix-like systems:
Installing Connector/Python with pip Use pip to install Connector/Python on most any operating system:
pip install mysql-connector-python
• datetime - The datetime module supplies classes for manipulating dates and times.While date and time arithmetic is supported, the focus of the implementation is on efficient attribute extraction for output formatting and manipulation.
• HTTPProvider - The provider is how web3 talks to the blockchain. Providers take JSON-RPC requests and return the response. This is normally done by submitting the request to an HTTP or IPC socket based server, which attempts to connect to “http://localhost:8545” or ”http://localhost:7545”
• json - JSON is a syntax for storing and exchanging data. JSON is text, written with JavaScript object notation.
•smptlib - The smtplib module defines an SMTP client session object that can be used to send mail to any internet machine with an SMTP or ESMTP listener daemon.
•qrcode - Python has a library “qrcode” for generating QR code images. It can be installed using pip. pip install qrcode.

4.2 Proposed System

We proposed block chain based management system that activates the consumer and enterprise vendor to track and identify the real product using a Smartphone. Counterfeit products are detected using QR Code reader, where a QR Code of the product linked to a BCBM system.

Manufacturers that are approved by the admin can add product details to the block chain and use the qr code generated for the products. Customers can scan the qr code on the product. If the unique code from qr code of customer matches the unique code in block chain database, product details are returned and else alerted for counterfeit product. Customer can then view entered products or send a feedback based on the result.

4.3 Feasibility Study

A feasibility study is a high-level capsule version of the entire System analysis and Design Process. The study begins by classifying the problem definition. Feasibility is to determine if the project is worth undertaken. Once an acceptance problem definition has been generated, the analyst develops a logical model of the system. A search for alternatives is analyzed carefully. There are 3 parts in feasibility study.

4.3.1 Technical Feasibility

The technical feasibility is the trickiest part of a feasibility study. This is because, at this point in time, not too many detailed design of the system, making it difficult to access issues like performance, costs on (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis. Understand the different technologies involved in the proposed system before commencing the project we have to be very clear about what are the technologies that are to be required for the development of the new system. Find out whether the organization currently possesses the required technologies. Is the required

technology available with the organization? Current technical resources are sufficient for the new system.Unlike traditional applications, web systems are accessible anytime, anywhere via any device with an Internet connection. Now internet,mobiles etc are available in less cost.So that the project is feasible within the limits of current technology.

4.3.2 Operational Feasibility

Proposed project is beneficial only if it can be turned into information systems that will meet the organizations operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? Here are questions that will help test the operational feasibility of a project:

- Is there sufficient support for the project from management from users? If the current system is well liked and used to the extent that persons will not be able to see reasons for change, there may be resistance.
- Are the current business methods acceptable to the user? If they are not, Users may welcome a change that will bring about a more operational and useful systems.
- Have the user been involved in the planning and development of the project? Early involvement reduces the chances of resistance to the system and in general and increases the likelihood of successful project. Since the proposed system was to help reduce the hardships

4.3.3 Economical Feasibility

Economical feasibility analysis is the most commonly used method for determining the efficiency of a new project. It is also known as cost benefit analysis also used to conduct Return Of Investment(RIO)analysis, along with tangible and intangible benefits. It helps in identifying profit against investment expected from a project. Cost and time are the most essential factors involved in this field of study.

CHAPTER 5
SYSTEM DESIGN

5.1 Architecture Diagram

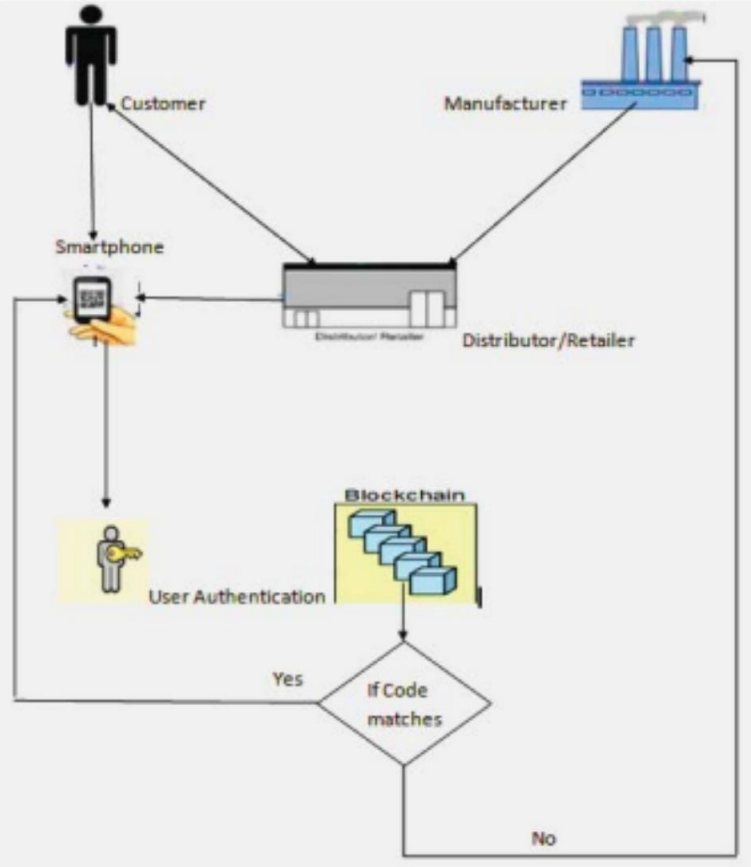


Fig. 5.1: Architecture Diagram

Fig. 5.1 is the architecture of the proposed system of Counterfeit detection. Here, we can understand that the system will detect counterfeit products using QR code reader, where a QR code of the product is tied to a block chain system so that you can scan the QR code using your smart phone. It will notify you whether the product is fake or not.

A customer can use the smartphone and scan the qr code of the product and after the completion of authentication process, a unique code from the customer will compare the code against the entries in block chain database. If it matches, product details are sent to customer. Otherwise, alerted for fake product and customer can send a feedback on the product. They can also check on the entered products of each manufacturer.

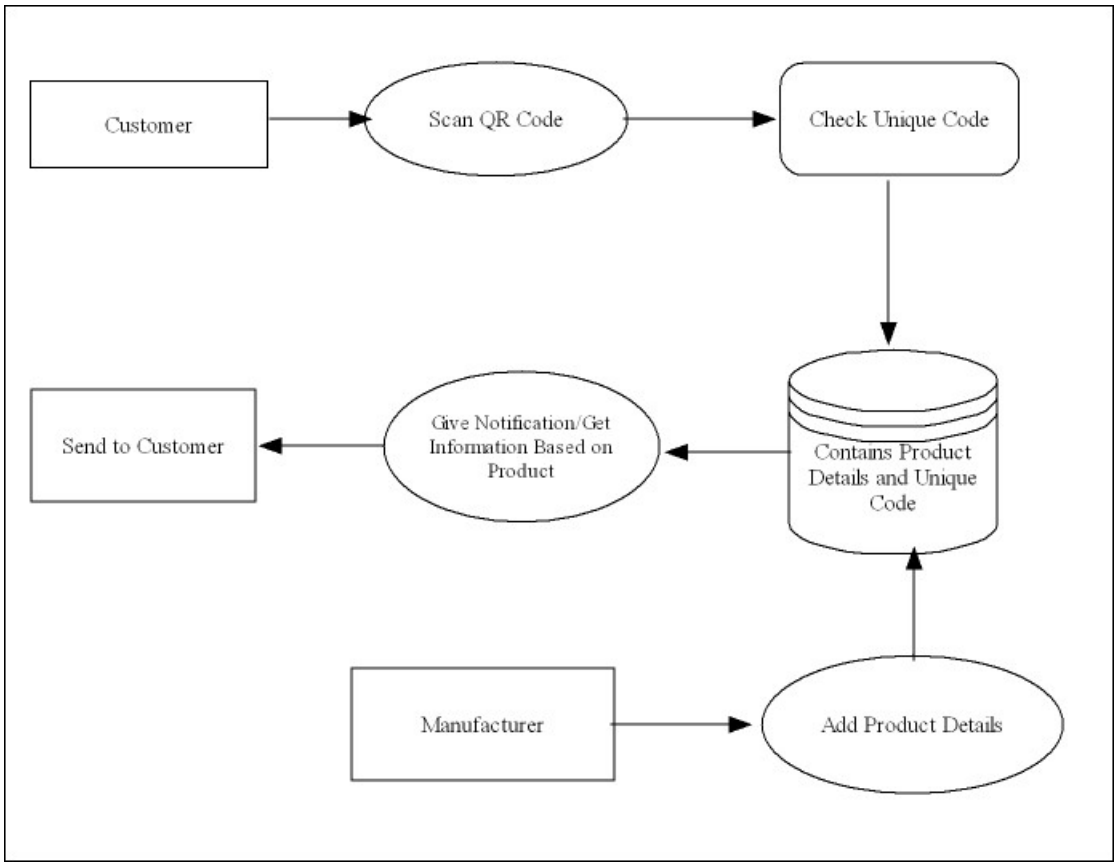


Fig. 5.2: Architecture Block Diagram

The manufactures can register and when approved by the admin, can login to add product details to the database. Each time a product is added to the block chain database, a qr code for the product will be generated. Manufactures can also view the products they entered into the database. Manufactures and admins can view the feedback given by the customers.

5.2 Module Design

Counterfeit Detection consists of three modules:

- Admin
- Manufacturer
- Customer

5.2.1 Admin

Admin has a separate account where in admin logs in via a username and password, and admin can approve or reject manufacturers, view the registered customers and their feedbacks.

5.2.2 Manufacturer

Manufacturers can register for login credentials and when approved by admin, can login with those credentials to add products, view them and view customer feedbacks.

5.2.3 Customer

Customers can use the application to scan qr code of products and if qr code matches of that with the product details in block chain database, product details are returned else alerted for fake product. Customers can then register, login and give feedback or view products entered in block chain.

5.3 System Design

On a lower level, BLOCK CHAIN BASED COUNTERFEIT PRODUCT DETECTION has three modules - Admin, Manufacturer, Customer- each have the user interface and Ganache - Smart Contract integration and storage. All these modules combine to set up this project on a local machine. Since the application is developed in the ethereum blockchain, hosting this project online costs above INR 20,000. So I haven't tried hosting this project online. I have

used ganache UI to emulate the blockchain on my local machine. Ganache is a free windows application which can be used to emulate blockchain on 10 free accounts.

5.4 Smart Contracts

A smart contract is a computer program or a transaction protocol respectively, which is intended to automatically execute, control or document respectively legally relevant events and actions according to the terms of a contract, of an agreement or of a negotiation.

5.5 Table Design

The Table 5.1 of student table is not a typical database table. It is a structured data stored in blockchain. It is defined in the smart contract.This structure stores the details of product such as product name, product manufacturing location and time, ingredients, manufacturer name and id, product number, product image and Hash of file stored.

Name	Type	Null	Constraint
Product Id	Integer	NO	PRIMARY KEY
Product Name	STRING	NO	
Manufacturer Name	STRING	NO	
Manufacturer Id	Integer	NO	
Manufacturer Location	STRING	NO	
Manufacture Date	STRING	NO	
Product Ingredients	STRING	NO	
Product Image	STRING	NO	

Table 5.1: Table Design in Smart Contract

5.6 Procedural Design

Software Procedural Design converts and translates structural elements into procedural explanations. SPD starts straight after data design and architectural design. This has now been mostly abandoned mostly due to the rise

in preference of Object Oriented Programming and design patterns. A design methodology combines a systematic set of rules for creating a program design with diagramming tools needed to represent it. Procedural designs best used to model programs that have an obvious flow of data from input to output.

5.7 Ganache and Smart contract

Ganache is the latest version of TestRPC. It is a fast and customizable blockchain emulator. It allows you to make calls to the blockchain without the overheads of running an actual Ethereum node. Solidity is purposefully a thin language and a loosely typed language with a syntax very similar to Javascript. Solidity is highly preferable to develop a smart contract.

5.8 Data Flow Diagram

A data flow diagram is a graphical representation of the "flow" of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (Structured Design). A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will be stored. It does not show information about the timing of process or information about whether processes will operate in sequence or in parallel.

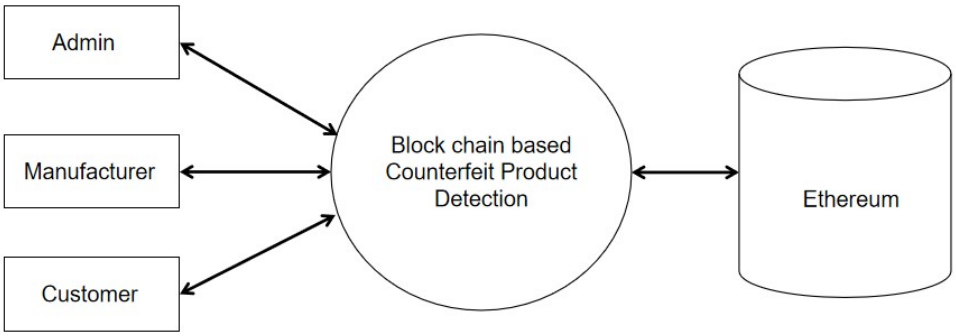


Fig. 5.3: Level 0-DFD

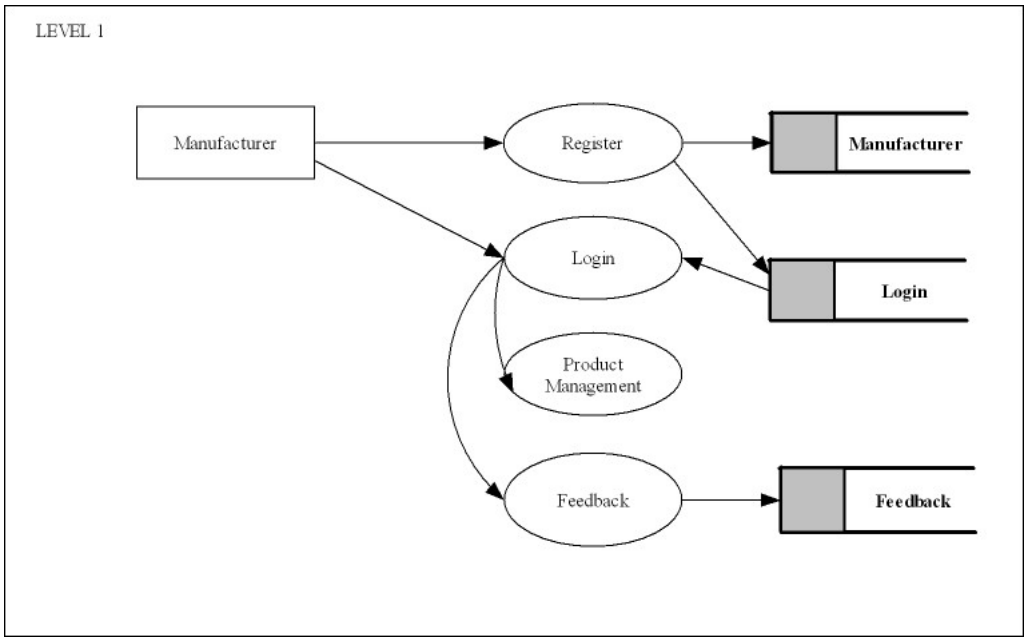


Fig. 5.4: Level 1-DFD

Figure 5.3 shows the level 0 DFD of Counterfeit Product Detection system. End users interacting with ethereum block chain emulator ganache.

Figure 5.4 shows the level 1 DFD, with manufacturer, while figure 5,5 shows level 2 DFD with Customer and figure 5.6 the level 3 DFD with Admin.

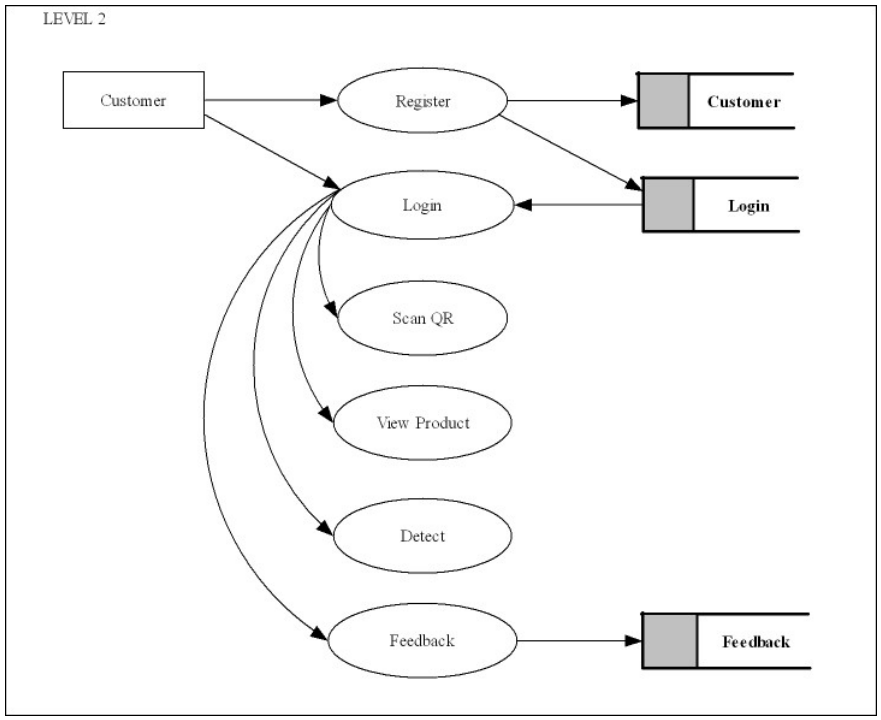


Fig. 5.5: Level 2-DFD

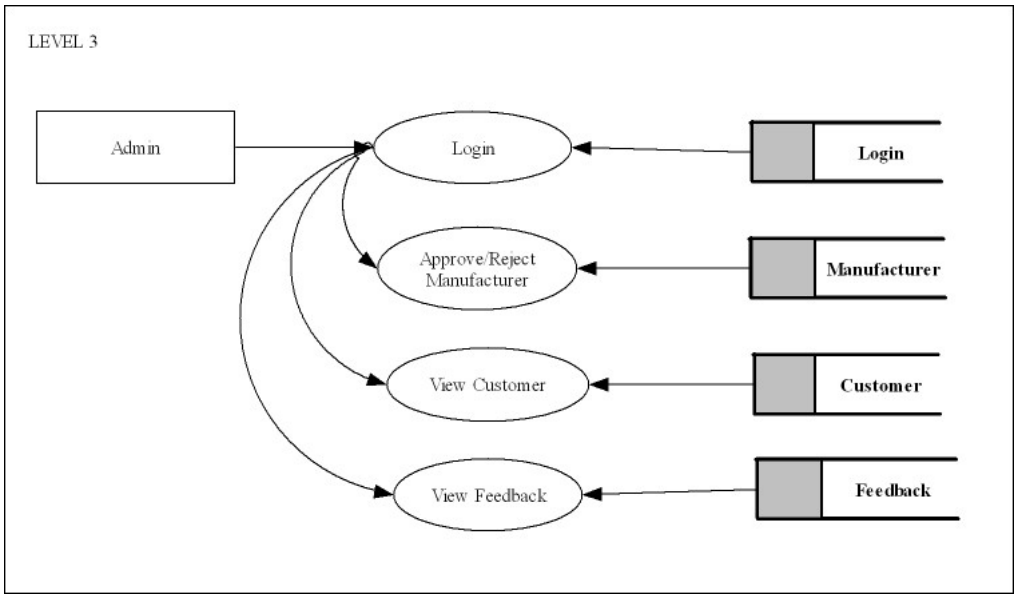


Fig. 5.6: Level 3-DFD

5.8.1 Input Design

Input design is the link between information system and the user. It comprises of determining set of inputs, validates the data, minimizes the data entries and thereby provides multiuser facilities. The input is designed in such a way that it provides security and ease of use with retaining privacy. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database.
In COUNTERFEIT PRODUCT DETECTION, product details from the system is used as input. All input data is validated, and if any data violates any conditions, a message is displayed to the user.

5.8.2 Output Design

A quality yield is one, which meets the prerequisites of the end client and presents the data unmistakably. In any framework consequences of handling are imparted to the clients and to other framework through yields. In yield structure it is resolved how the data is to be uprooted for sure fire need and furthermore the printed version yield. It is the most significant and direct source data to the client. Productive and clever yield configuration improves the framework's relationship to help client dynamic. Planning PC yield ought to continue in a composed, very much idea out way; the correct yield must be created while guaranteeing that each yield component is structured with the goal that individuals will discover the framework can utilize effectively and adequately. In COUNTERFEIT PRODUCT DETECTION dif-

ferent yields are Product Details held up against an individual and advanced archive put away in the blockchain. The Hash of the document put away in blocks is put away in the square chain.Customers and manufacturers can view product details as output from block chain.

5.9 Software Architecture

5.9.1 Blockchain Technology

Blockchain, is also known as distributed ledger technology, makes the nature of any digital thing unchangable and efficient through the use of cryptography hashing and decentralization.

An easy way for understand block-chain is Google-Doc application. When we make a document and sharing it with a group of people, the documents are distributed instead of moved or copied. This makes a decentralized distribution in the form of a chain that gives everyone access to the document at the same time. Any of the users are not locked out awaiting changes from another party, while all alterations to the doc are being recorded in real-time, making changes completely reliable. Blockchain consists of three important concepts: blocks, nodes and miners.

5.9.1.1 Blocks

A block-chain consists of multiple blocks and each block has three elements:

- The data stored in the block.
- A 32-bit whole number which is called a nonce. That is randomly generated when a block is created, which then creates a block header hash.
- The hash will be a 256-bit integer attached to the nonce. It starts with a big number of zeroes (i.e., be very small).

When the initial block of a chain is created, the nonce generates the hash. The data stored in the block is signed and forever attached to the nonce and

the hash unless it is mined.

5.9.1.2 Miners

Mining is the process of creating new blocks on the chain by the miners. In a blocks-chain all blocks have its own novel key and hash, yet additionally interfaces the hash of the past blocks of the chain, so mining a blocks isn't so much simple, extraordinarily on huge chains. Miners utilize external coding projects to take care of the mind boggling math issue of finding a key that produces an acknowledged hash.

Making changes to any blocks prior in the chain requires re-mining the blocks with the modify, yet the entirety of the hinders that come after it. This is the reason it's exceptionally hard to control blocks chain innovation. Consider it is as "well being in math" since finding good nonce requires a various measure of time and figuring power. At the point when a blocks is mined, the change is acknowledged by the entirety of the hubs on the system and the Miner is compensated financially.

5.9.1.3 Nodes

One of the most center ideas in block chain innovation is decentralization. No organization or PC can claim a chain. Rather, it is a dispersed record and the hubs associated with the chain. Hubs can be any kind of electronic gadget that keeps up duplicates of the block chain and keeps the system working.

Each hub has its own duplicate of the block chain and the system must affirm any recently dug obstruct for the chain to be changed, confirmed and demonstrated. Since block chains are straightforward, each activity in the record can be effectively checked and seen. Every client is given a one of a kind distinguishing proof number that shows their exchanges.

5.9.2 Ethereum Blockchain

Ethereum is an open-source, blockchain-based, decentralized software platform used for its own cryptocurrency, ether. It enables SmartContracts and Distributed Applications (DApps) to be built and run without any downtime, fraud, control, or interference from a third party. Ethereum is a distributed public block chain network that focuses on running programming code of any decentralized application. More simply, it is a platform for sharing information across the globe that cannot be manipulated or changed.Ethereum is not just a platform but also a programming language (Turing complete) running on a blockchain, helping developers to build and publish distributed applications.

Ether is a decentralized digital currency, also known as ETH. In addition to being a tradeable cryptocurrency, ether powers the Ethereum network by paying for transaction fees and computational services. Ether is paving the way for a more intelligent financial platform.

CHAPTER 6
SYSTEM IMPLEMENTATION

6.1 Smart Contract

Nick Szabo introduced the term Smart Contract. It is a set of user interfaces and computer protocols designed for securing and formalizing agreements and connections over computer networks. The clauses of traditional legal contracts are captured and converted into a couple of computer logic rules which are run as a computer code. Programs of smart contracts in many cases are used to authorize the perception of the essential clauses by all participating entities.

6.2 Ethereum Accounts

Any substance in ethereum having an inward state is associated with an account, for example an open/private key pair . The open key speaks to the address of the record. This block chain separates two sorts of accounts, for example outside claimed records and agreement account. An outer possessed record is an individual record constrained by a private key. Messages or ether to other outside records can be send by the proprietor of the private key. Though the record that have its own rationale associated into the code that controls it is known as an agreement account.

6.3 Contracts and Transactions

Ethereum insert programmable logic into the blockchain with the help of contracts. Contracts does the following:

- Stores the condition of qualities associated with different agreements or outside elements. For eg. , the digital currency contract spares the record equalizations of any individual who collaborates with the agreement.

- Goes about as outside record with advantaged get to approaches. For eg. an informing administration that checks if certain obliges are met to advance messages.

- Maps and oversees connections among different clients. For instance, the rationale of a certifiable money related agreement can be mapped which will consistently be enforceable inside the ethereum condition.

- Gives capacities to different agreements along these lines demonstrations like programming libraries. Agreements associate with each other by message passing. Substance of the messages can be account addresses, byte exhibits, measures of ether. When an agreement gets such a message it returns information which are consumable by the message sender, as the working of a customary capacity call.

The ethereum execution condition stays a loop until something gets it under way. Activities are activated through the system called exchanges. An exchange can be send by a client from their record to another outer record or to an agreement. In the principal case ether is moved from their parity to the equalization of the outside record client, yet other than that they don't have anything to do. At the point when the recipient is an agreement, i.e.in the second case, the agreement become dynamic and executes its code. An agreement is able of perusing or altering its inward state, expending the got message or on the other hand can even make the execution of an other agreement by sending it as a message. Nature becomes aloof state when the execution of an activity and the entirety of its sub sequence activities is finished. It stays idle until next exchange is gotten.

To join blockchain innovation in the item confirmation framework, one ought to essentially see how blockchain record functions under the hood.

Blockchain has a worked in character component, a cryptographically secure key pair(as referenced in the above area). Every member on the arrange is doled out a specific movement utilizing these keys. A member can be a substance, a gadget or an individual. The members' unique personalities are disguised and they are perceived by these keys. A key pair has no clue about the member's very own information (for example name, contact or expert qualifications) can be related with it. In this project a predefined set of 10 local ethereum account is created in the ganache as shown in the figure 6.1.

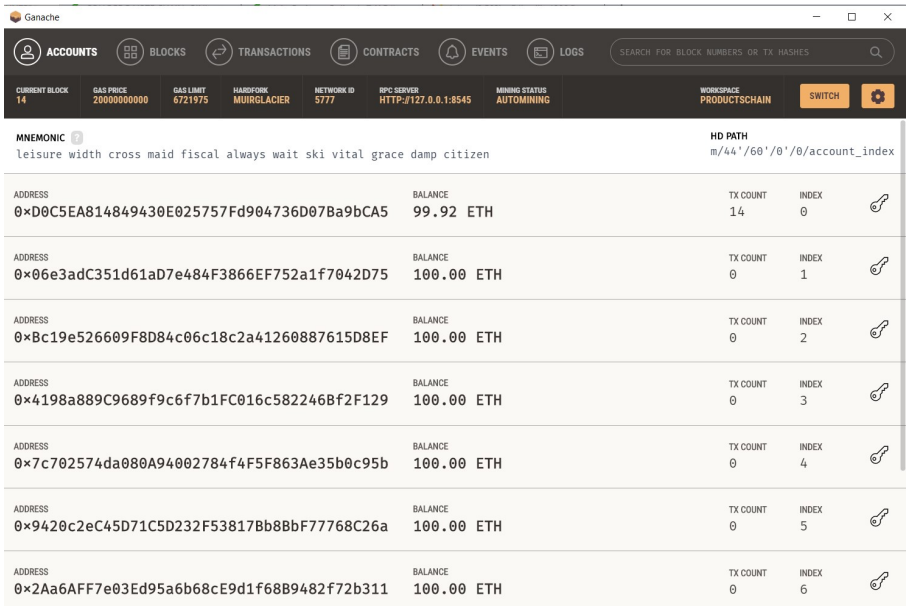


Fig. 6.1: Local Ethereum Accounts in Ganache

In the proposed system, product details are stored in decentralized application that records transactions on ethereum and manufacture details and their products, customer details and feedback are stored in DBMS . The functionality is provided by the smart contract:

- product.sol - To store product details and Hash of the file.

In the context of this work, the participants will be admin,manufacturer and customers each having their own dashboard. Each will be identified by their username and password.

First install NPM, node.js, truffle web3, and wambserver(temporary storage) then open Pycharm IDE with project folder run blockchain.py so that

http://localhost:5000/ to launch the application. The implementation of this system is explained as follows. The contract logic, migrations and tests use True and Ganache as basic environment. These must be installed in the PC. Install node package, http-provider package can be exported from web3.So that interaction between forms and blockchain database can be managed by web3.

Start ganache GUI, a window will open up after deploy contract solidity file. Then select localhost:8545 in networks in order to get connected. This loads all the accounts ganache creates in ethereum gas can be used for transactions.

Open the terminal and go to the project folder, then type the commands truffle migrate or truffle compile. This command will compile and deploy the smart contracts. If no error is found terminal shows, if there is any error, the migration of smart contracts fails and it can only be deployed after rectifying those errors.

The homepage opens with particular login page of customer, manufacturer, admin. Admin can view all customers and their feedback and also approve or reject manufacturers. Approved manufacturers can login and add product details, view them and customer feedbacks. Customer can login into the system by identifying the corresponding person and login as that particular user, then view products and give feedbacks. Customers can use the application on smartphone to scan qr code of product and if it matches with unique code in block chain, get the product details.

The web user interface contains the web3.js library used to connect to the ethereum blockchain. Web designing is done using HTML and web development using node.js.

CHAPTER 7
RESULTS AND DISCUSSION

A network of trusted parties are formed. A blockchain, at the back-end, will store all the required transactions, and once the data is added, it can never be altered. This system now has smart contracts ready and interface for each role. It has a view for admin,manufacturer and customer. The ganache will be giving 10 accounts and their respective private keys as shown in figure 7.1.

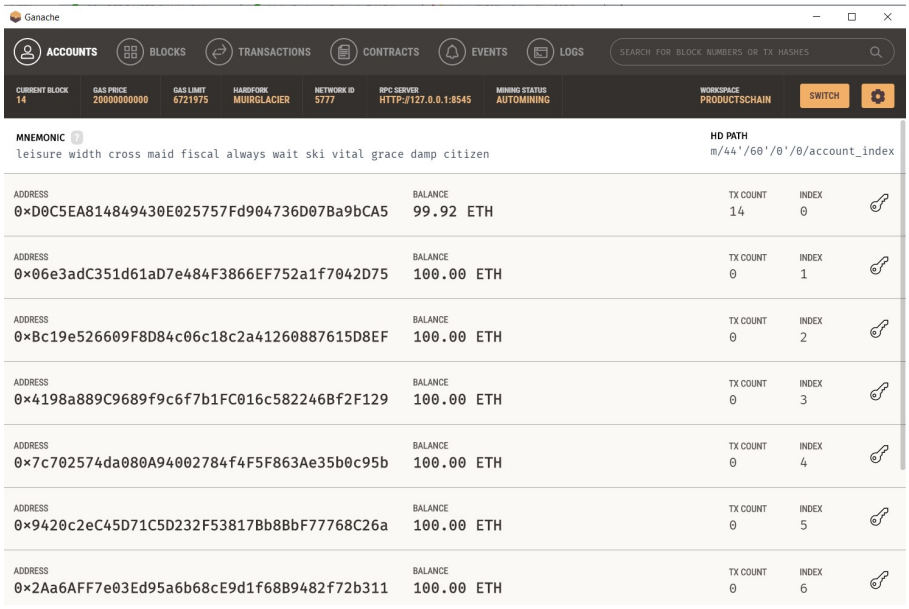


Fig. 7.1: Local Ethereum Accounts in Ganache

The private key of each account can be viewed by clicking on the key symbol against the ganache accounts as seen in figure 7.1. The ganache will automatically start mining blocks as soon as it is launched and the number of blocks can be seen on the top-left corner. On clicking the transactions button all the transaction history can be viewed.

There are 3 major type of users in this portal.They are customer, admin, manufacturer, they can login into the system through a common login page as shown fig.7.2 .

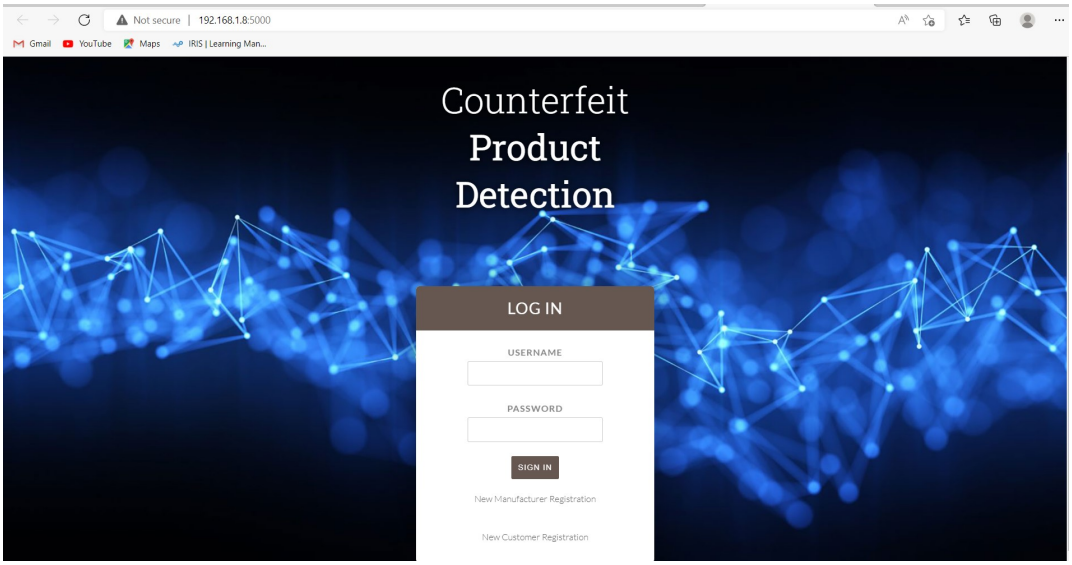


Fig. 7.2: Common login page

Admin can be login into system and approve or reject manufacturers, view customers and their feedbacks as shown in fig 7.3, fig 7.4 and fig 7.5 .

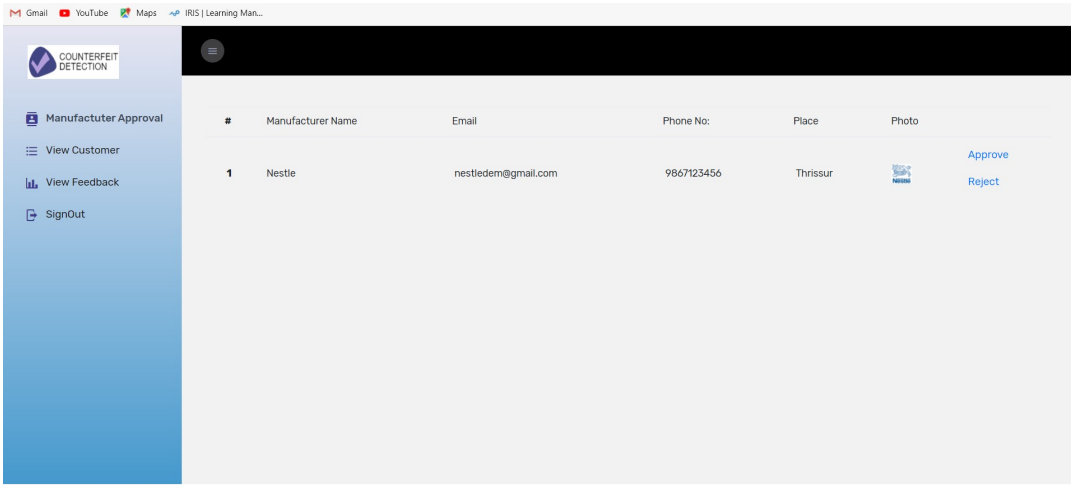


Fig. 7.3: Admin - Approve or Reject Manufacturer

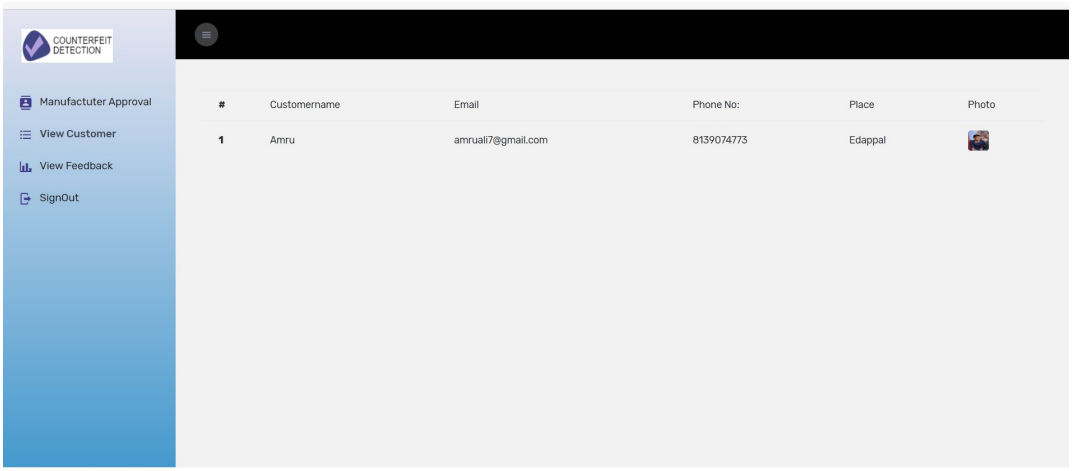


Fig. 7.4: Admin - View Registered Customers

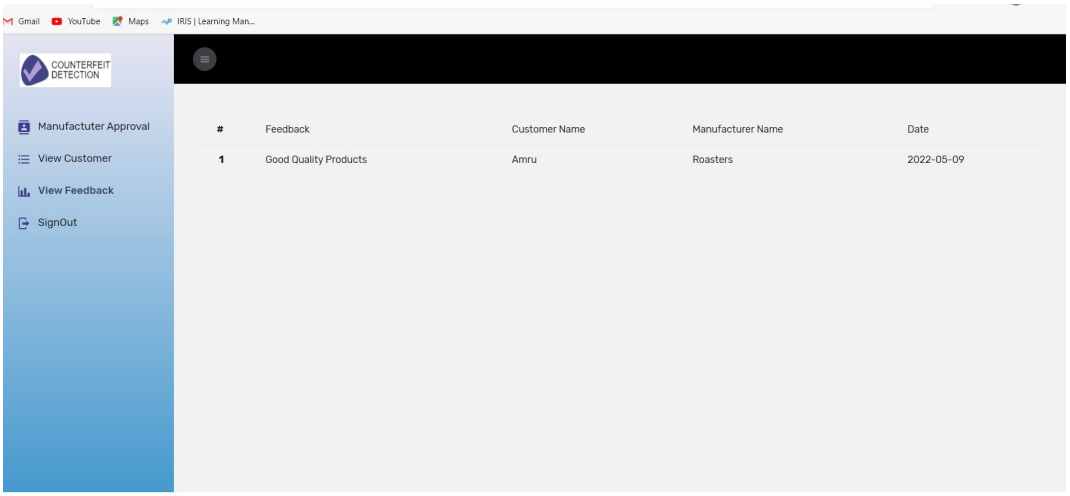


Fig. 7.5: Admin - View Customer Feedbacks

Manufacturers can add products details and add the products to block chain while viewing all products added by the manufacturer, as shown in fig 7.6, fig 7.7.

The added products are stored in blocks that can be viewed by clicking blocks option in ganache, as shown in fig 7.8. Manufacturers can also view the feedback provided to them by the customers, as shown in fig 7.9.

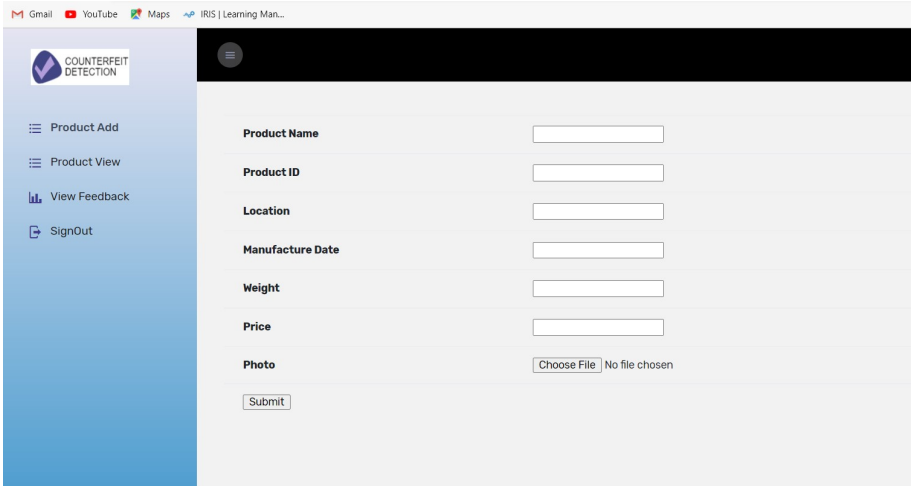


Fig. 7.6: Manufacturer- Add Product Details

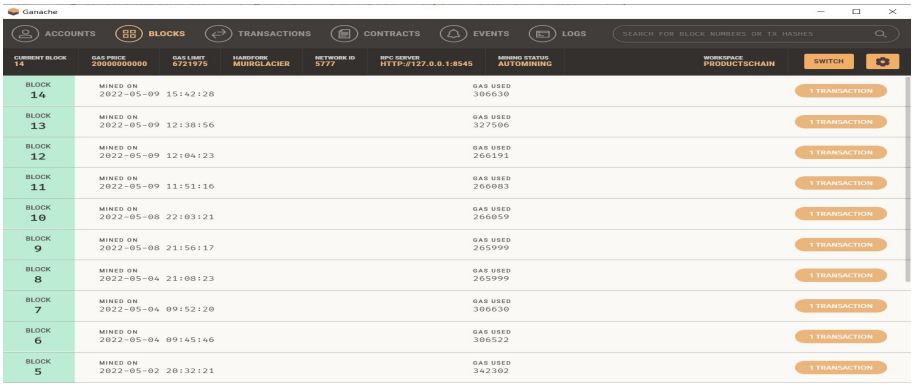


Fig. 7.7: Block Chain- /Blocks added

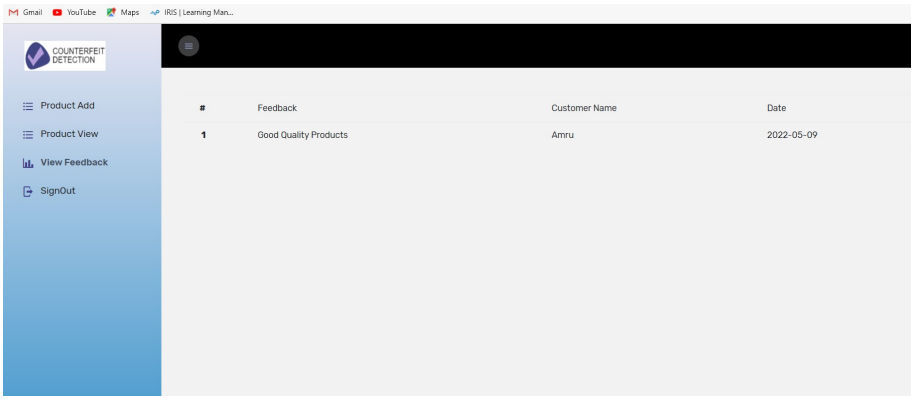


Fig. 7.8: Manufacturer- Customer Feedbacks

Customers can Login to view products in block chain database or give feedback as shown in fig 7.10 and fig 7.11.

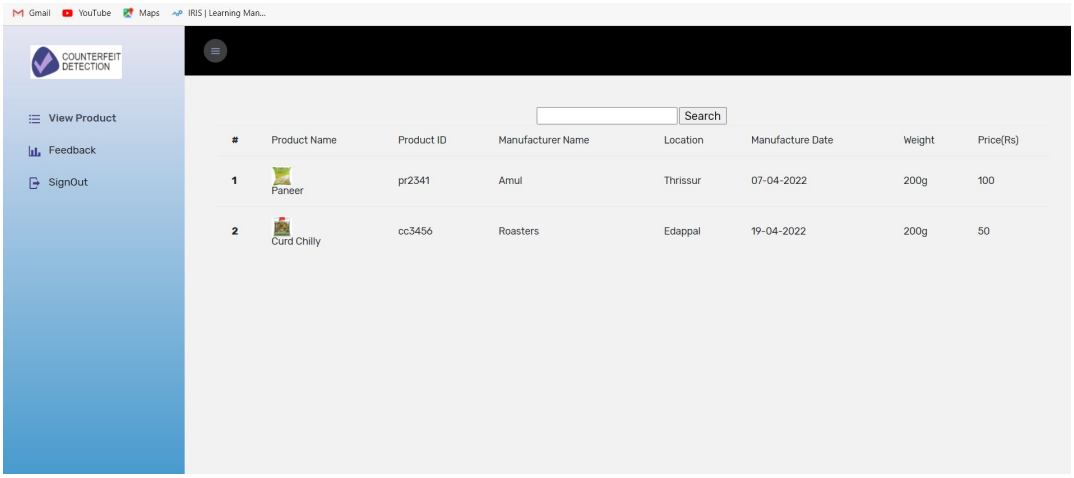


Fig. 7.9: Customer- View Products

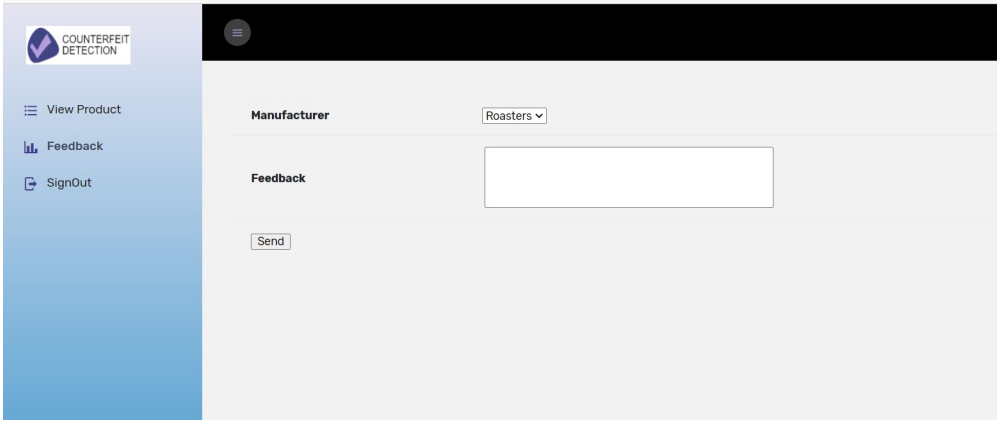
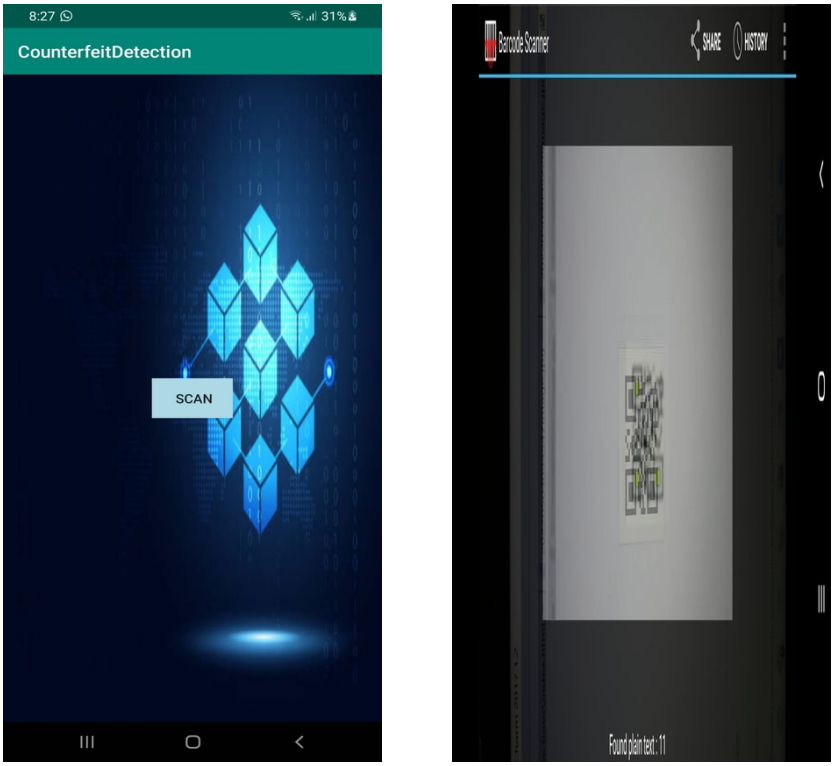


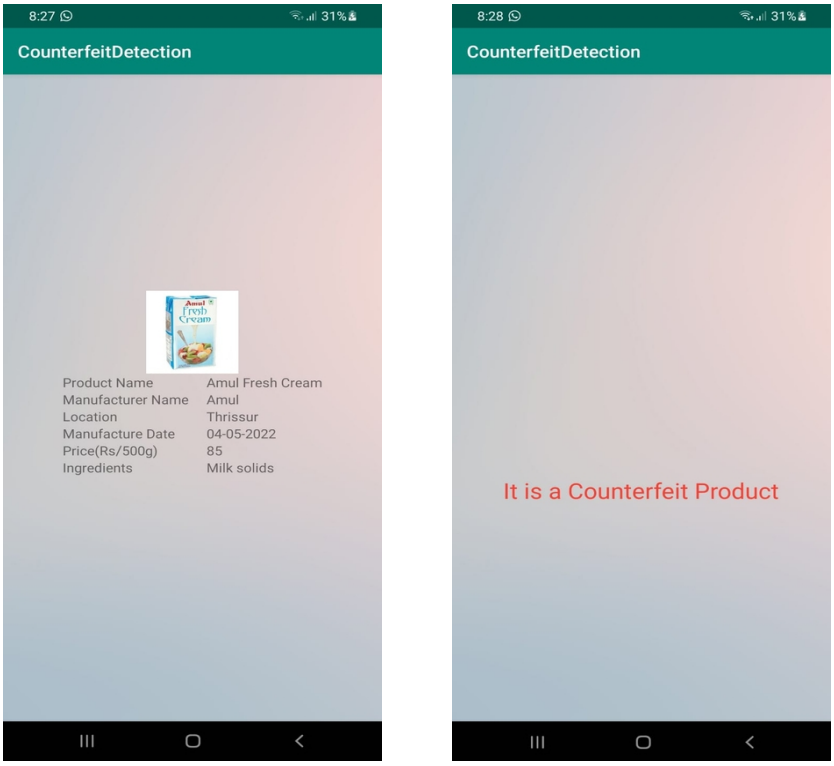
Fig. 7.10: Customer- Send Feedback

Customers can use application to scan qr code and get the product details or counterfeit product alert, as shown in fig 7.12, fig 7.13, fig 7.14, fig 7.15.



(a) View Page (b) Scan QR Code

Fig. 7.11: Android Application



(a) Product Detected (b) Fake Product Detected

Fig. 7.12: Android Application

CHAPTER 8
CONCLUSION

Due to the increasing number of counterfeit products being sold in the black market and online, the challenge of controlling the supply chain is becoming more prevalent. Although the government has enacted various laws and regulations against fake products, it is still not able to prevent them from entering the market. This is why it is important that the government and the private sectors work together to identify and prevent these products from entering the supply chain. Through a block chain management system, manufacturers can store the details of their product sales within a secure and accessible database. This system can also perform vendor-side verification to ensure that the products are real. Only the owner of the device can access the data stored within the block chain management system. This system can then identify the real product and prevent it from entering the supply chain.

CHAPTER 9
SCOPE FOR THE FUTURE ENHANCEMENT

The block chain technology has a big efficiency for enhancement and application in the logistics and supply chain. This decentralized technology may be useful for all applications and economic strengthening will be the future challenging one for securing counterfeits in supply chain.

The main improvements required for the system is to ensure the performance with large scale operation as the scale of the system increases the performance and reliability needs to be tested. The system may require minor redesign in-order to accommodate changes in the Product details.

The system can be customized for a single manufacturer, so that each manufacturer can have a single block chain database.

CHAPTER 10

SCREENSHOTS

```
// Creating a Smart Contract
contract StructDemo{
// Structure of product
struct product{
    // State variables
    uint bid;
    string productname;
    string  location;
    string timestamp;
    string price;
    string ins;
    string photo;
    string mname;
    uint mid;
}
product []emps;
// Function to add
// products details
function addProduct(uint bid, string memory p,string memory m,string memory l,string memory t ,string memory price,
        string memory ins,string memory photo,uint mid) public{
    product memory e =product(bid, p,m,l, t ,price,ins,photo,mid);
    emps.push(e);
}

// Function to get details of product
function getProduct(uint bid) public view returns(string memory, string memory, string memory,
        string memory,string memory,string memory,string memory,uint){
    uint i;
    for(i=0;i<emps.length;i++)
    {
        product memory e =emps[i];
        if(e.bid==bid)
        {
                return(e.productname,e.mname,e.timestamp,e.location, e.price ,e.ins,e.photo,e.mid);
        }
    }
    return("0","0","0","0","0", "0","0",0);
}
```

Fig. 10.1: Smart Contract- Product Details

```
@app.route('/View_Product1')
def View_Product1():
    if session['lg'] == "lin":
        data = []
        with open(compiled_contract_path) as file:
            contract_json = json.load(file)  # load contract info as JSON
            contract_abi = contract_json['abi']  # fetch contract's abi - necessary to call its functions
        contract = web3.eth.contract(address=deployed_contract_address, abi=contract_abi)
        blocknumber = web3.eth.get_block_number()
        print(blocknumber)
        for i in range(blocknumber, 4, -1):
            a = web3.eth.get_transaction_by_block(i, 0)

            print(a)
            decoded_input = contract.decode_function_input(a['input'])
            res = {}
            res['productname'] = decoded_input[1]['pn']
            res['productid'] = decoded_input[1]['pd']
            res['location'] = decoded_input[1]['l']
            res['mdate'] = decoded_input[1]['md']
            res['price'] = decoded_input[1]['price']
            res['weight'] = decoded_input[1]['w']
            res['photo'] = decoded_input[1]['photo']
            res['mid'] = decoded_input[1]['mid']
            if str(decoded_input[1]['mid']) == str(session['m_id']):
                data.append(res)
        return render_template("Manufacturer/View Product.html",data=data)
    else :
        return redirect('/')
```

Fig. 10.2: CounterfeitDetection.py

```
def Product_Management():
    if session['lg'] == "lin":
        if request.method == "POST":
            productname = request.form['textfield']
            productid=request.form['textfield4']
            location = request.form['textfield2']
            weight=request.form['textfield6']
            price=request.form['textfield3']
            mdate = request.form['textfield5']
            photo = request.files['filefield']
            import datetime,qrcode
            date = datetime.datetime.now().strftime('%y%m%d-%H%M%S')
            photo.save(r"C:\Users\fidha\PycharmProjects\CounterfeitDetection\static\products\\" + date + '.jpg')
            photo1 = "/static/products/" + date + '.jpg'
            db = Db()
            q = db.selectOne(
                "select * from `manufacturer` WHERE `manufacturer_id`='" +str(session['m_id'])+ "'")
            if q is not None:
                mname= q['m_name']
            with open(compiled_contract_path) as file:
                contract_json = json.load(file)  # load contract info as JSON
                contract_abi = contract_json['abi']  # fetch contract's abi - necessary to call its functions

            contract = web3.eth.contract(address=deployed_contract_address, abi=contract_abi)
            blocknumber = web3.eth.get_block_number()

            message2 = contract.functions.addProduct(blocknumber + 1, productname, productid, mname, location, mdate, weight, price, photo1, (session['m_

            # Create qr code instance
            qr = qrcode.QRCode(
                version=1,
                error_correction=qrcode.constants.ERROR_CORRECT_H,
                box_size=3,
                border=4,
            )
            # Add data
            qr.add_data(blocknumber + 1)
            qr.make(fit=True)
            import datetime
            img = qr.make_image()
```

Fig. 10.3: CounterfeitDetection.py

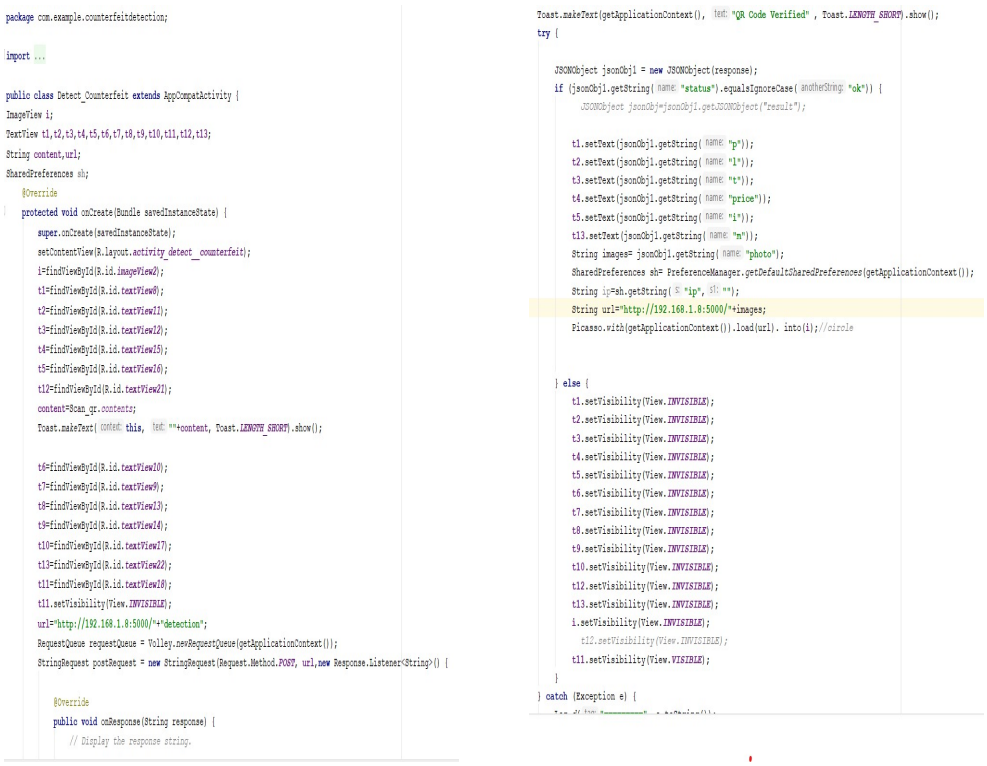
```
        db=Db()
        s = db.select("select * from feedback, customer, manufacturer where feedback.manufacturer_id=manufacturer.manufactu
        return render_template("Manufacturer/View Feedback.html", data=s)
    else :
        return redirect('/')

@app.route('/detection',methods=['post'])
def detection():
    bid=request.form['srno']
    with open(compiled_contract_path) as file:
        contract_json = json.load(file)  # load contract info as JSON
        contract_abi = contract_json['abi']  # fetch contract's abi - necessary to call its functions
    contract = web3.eth.contract(address=deployed_contract_address, abi=contract_abi)
    blocknumber = web3.eth.get_block_number()
    c=0
    for i in range(blocknumber, 4, -1):
        print(i)
        a = web3.eth.get_transaction_by_block(i, 0)
        decoded_input = contract.decode_function_input(a['input'])
        print(decoded_input[1]['bid'],"iiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiiii",bid)

        if str(decoded_input[1]['bid'])==bid:
            print(decoded_input[1]['photo'])
            return jsonify(status="ok",p=decoded_input[1]['p'], m=decoded_input[1]['m'], l=decoded_input[1]['l'],
t=decoded_input[1]['t'], price=decoded_input[1]['price'], i=decoded_input[1]['ins'], photo=decoded_input[1]['photo'])
        else:
            c+=1
    if c>0:
        return jsonify(status="none")


if __name__ == '__main__':
    app.run(host="0.0.0.0")
    # app.run()
```

Fig. 10.4: CounterfeitDetection- Android



(a) (b)

Fig. 10.5: CounterfeitDetection- Android

Fig. 10.6: CounterfeitDetection- Android

BIBLIOGRAPHY

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

[2] Jinhua ma, Shih-Ya Lin, Xin Chen, Hung-min-sun, Yeh-Cheng Chen, Huaxiong Wang *"A Block Chain Based Application For Product AntiCounterfeiting (IEEE)"* 2020

[3] Ghaith Khalil, Robin Doss, Morshed Chowdhury *"A New Novel RFID Based Anti-Counterfeiting Scheme for Retail Environment (IEEE) "* 2020

[4] Kentaroh Toyoda, P. Takis Mathiopoulos, I. Sasase and Tomoaki Ohtsuki *"A Novel Blockchain-Based Product Ownership Management System (POMS) for Anti-Counterfeits in the Post Supply Chain"* 2017

[5] Tejaswini Tambe, Sonali Chitalkar, Manali Khurud, Madhavi Varpe, S. Y. Raut *"Fake Product Detection Using Blockchain Technology"* 2021

[6] https://fintelics.medium.com/fake-product-identification-using-blockchain-technology-73ea162de9be