Democratic and Popular Algerian Republic

Ministry of higher education and Scientific Research

20 August 1955 University – Skikda

Faculty of Sciences

Department of Computer Science

THESIS

Presented for the diploma of

Master in Computer Science

Option**: GLAA**

Anti-counterfeit Blockchain based system with Ontology

Presented in 04/07/2023 Order N°: .....

By **AHMED LAGGOUN**

Board of Examiners:

Dr. NABET Aicha	Skikda	**President**
Dr. KASRI Soumaya	Skikda	**Supervisor**
Dr. BOUTINE Rachid	Skikda	**Examiner**

Promotion 2022/2023

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

DEDICATION

*First, I dedicate my dissertation work to the sake of **Allah**, my Creator and my Master. My great teacher and messenger, Mohammed (May **Allah** bless and grant him), who taught us the purpose of life.*

I dedicate my dissertation to all my family. A special feeling of gratitude to my loving parents, whose words of encouragement and push for tenacity ring in my ears. My sisters and my brothers have never left my side and are very special.

I also dedicate this dissertation to all friends who have supported me throughout the process. I will always appreciate all they have done.

May ALLAH grant them Jannah Firdaus. Ameen

ACKNOWLEDGEMENTS

*Prima facie, I am grateful to **ALLAH** for the guidance, good health, wellness and willpower that were necessary to complete this thesis.*

*I would like to thank **my Parents**, who always believed in me. It is thanks to their support and prayers that I accomplished this work, they already know how much I owe them.*

*I also would like to thank my supervisor, Doctor **Kasri Soumaya**, whose expertise was invaluable in formulating the research methodology. Your insightful feedback pushed me to sharpen my thinking and brought my work to a higher level..*

I also place on record, my sense of gratitude to one and all, who directly or indirectly, have lent their hand in this venture. Finally, I also want to thank the jurors for agreeing to review and judge my work.

LAGGOUN AHMED

ABSTRACT

The rise of counterfeit products poses significant challenges for both consumers and businesses, jeopardizing consumer safety and causing substantial economic losses. This thesis presents the design and implementation of an anti-counterfeit blockchain-based system that aims to tackle this issue by ensuring the authenticity of products within the supply chain. The system leverages the power of blockchain technology and ontology to establish a trusted and transparent environment for product verification.

The proposed system consists of multiple components that work in synergy. A web application serves as the central hub for account management and product addition, allowing companies to create accounts and add their products. Prior to adding a product, the ontology layer validates the product's information against predefined rules using SWRL. Subsequently, the Metamask wallet verification layer ensures that only authorized users can add inventories to the blockchain.

To verify the authenticity of a product, a mobile application equipped with a QR code scanner enables consumers to quickly and conveniently check the product's validity. Upon scanning the QR code, the application retrieves information stored on the blockchain and provides real-time confirmation of the product's authenticity.

Through the integration of blockchain technology, ontology, and mobile applications, this anti-counterfeit system offers numerous benefits. It boosts consumer confidence through the provision of transparency and traceability., empowering consumers to make informed purchasing decisions. For businesses, it enables them to protect their brand reputation, mitigate financial losses, and combat counterfeit activities effectively.

The results of this thesis demonstrate the feasibility and effectiveness of using blockchain and ontology in combating counterfeit products. Future research directions may focus on en-

hancing system performance, exploring interoperability with existing supply chain systems, and investigating the integration of emerging technologies such as Internet of Things (IoT) for enhanced product tracking and verification.

Overall, this thesis contributes to the growing body of knowledge in the field of anti-counterfeiting systems and provides a solid foundation for further advancements in ensuring product authenticity and consumer protection.

Keywords: Blockchain, Ontology, QR Code, Trust, Counterfeiting.

ملخص

يشكل ظهور المنتجات المقلدة تحديات كبيرة لكل من المستهلكين والشركات ، مما يعرض سلامة المستهلك للخطر ويتسبب في خسائر اقتصادية كبيرة. تقدم هذه الأطروحة تصميم وتنفيذ نظام قائم على سلسلة الكتل لمكافحة التزييف يهدف إلى معالجة هذه المشكلة من خلال ضمان أصالة المنتجات داخل سلسلة التوريد. يستفيد النظام من قوة تقنية سلسلة الكتل والأنطولوجيا لإنشاء بيئة موثوقة وشفافة للتحقق من المنتج.

يتكون النظام المقترح من مكونات متعددة تعمل في تآزر. يعمل تطبيق الويب كمحور مركزي لإدارة الحسابات وإضافة المنتجات ، مما يسمح للشركات بإنشاء حسابات وإضافة منتجاتها. قبل إضافة منتج ، تتحقق طبقة الأنطولوجيا من صحة معلومات المنتج مقابل القواعد المحددة مسبقًا باستخدام. بعد ذلك ، تضمن طبقة التحقق من محفظة ميتام-ماسك أن المستخدمين المصرح لهم فقط يمكنهم إضافة قوائم جرد إلى سلسلة الكتل.

للتحقق من أصالة المنتج ، يتيح تطبيق الهاتف المحمول المجهز بماسح ضوئي لرمز الاستجابة السريعة للمستهلكين التحقق بسرعة وسهولة من صلاحية المنتج. عند مسح رمز الاستجابة السريعة ضوئيًا ، يقوم التطبيق باسترداد المعلومات المخزنة على سلسلة الكتل ويوفر تأكيدًا في الوقت الفعلي على أصالة المنتج.

من خلال دمج تقنية سلسلة الكتل وعلم الوجود وتطبيقات الهاتف المحمول ، يوفر نظام مكافحة التزييف هذا العديد من الفوائد. يعزز ثقة المستهلك من خلال توفير الشفافية وإمكانية التتبع ، وتمكين المستهلكين من اتخاذ قرارات شراء مستنيرة. بالنسبة للشركات ، فهي تمكنهم من حماية سمعة علامتهم التجارية ، وتخفيف الخسائر المالية ، ومكافحة الأنشطة المزيفة بشكل فعال.

تظهر نتائج هذه الأطروحة جدوى وفعالية استخدام سلسلة الكتل وعلم الوجود في مكافحة المنتجات المقلدة. قد تركز اتجاهات البحث المستقبلية على تحسين أداء النظام ، واستكشاف قابلية التشغيل البيني مع أنظمة سلسلة التوريد الحالية ، والتحقيق في تكامل التقنيات الناشئة مثل إنترنت الأشياء لتحسين تتبع المنتج والتحقق منه.

بشكل عام ، تساهم هذه الأطروحة في زيادة المعرفة في مجال أنظمة مكافحة التزييف وتوفر أساسًا متينًا لمزيد من التقدم في ضمان أصالة المنتج وحماية المستهلك.

كلمات مفتاحية: سلسلة الكتل، الأنطولوجيا، رمز الأستجابة السريعة، الثقة، التزوير.

CONTENTS

LIST OF FIGURES

ACRONYMS

PoW *Proof of Work*

PoS *Proof of Stake*

PoA *Proof of Authority*

RPoW *Reusable Proof Of Work*

DApps *Decentralized Applications*

EVM *Ethereum Virtual Machine*

SWRL *Semantic Web Rule Language*

OWL *Web Ontology Language*

RFID *Radio Frequency Identification*

ML *Machine Learning*

NFC *Near Field Communication*

AR *Augmented Reality*

QR Code *Quick Response Code*

GENERAL INTRODUCTION

The recent emergence of blockchain technology and its integration with ontology has a profound impact on various aspects of our society. Just as technology has revolutionized numerous sectors, it has also opened new avenues for enhancing the authenticity and security of supply chains. In the context of combating counterfeiting, the fusion of blockchain and ontology offers remarkable potential. This amalgamation not only ensures the traceability and transparency of inventories within the supply chain but also empowers consumers with increased confidence and trust. By leveraging the unique features of blockchain and ontology, such as decentralized data storage and semantic reasoning, the system addresses the critical challenges associated with counterfeiting, ultimately improving the safety and reliability of inventories.

In the realm of technology, "blockchain" has emerged as a ubiquitous term, transforming various markets in an era marked by skepticism and dissatisfaction with traditional intermediaries such as banks, institutions, and governments. Offering the allure of disintermediation and transparency, blockchain technology holds immense appeal and intrigue. Similar to how computers revolutionized data processing and the internet revolutionized information sharing, blockchain has the potential to revolutionize transactional processes. The pioneering implementation of blockchain technology can be seen in the form of cryptocurrency Bitcoin, which operates as an innovative payment network and decentralized form of currency, functioning without the need for a central authority. This groundbreaking technology is characterized by its openness and freedom, promising a paradigm shift in the way transactions are conducted.

There is no need to save information with third parties under a Blockchain approach. Records exist on many computers with similar information, so breaches make no sense, and if the data of a computer's Blockchain is breached, the system rejects such breach. Even if a hacker breaks into a network and tries to steal money from an account, multiple redundant and identical copies

of the same ledger are stored in the whole world. In the event of a breach, there are numerous others in the form of backups that can provide the funds from the hacked account. In other words, Blockchain data is spread across multiple nested computers. More than half of the security systems network must be infiltrated for hacking attempts to be effective.

Problematic and motivation

Counterfeiting is a major issue in many businesses, and the combination of blockchain technology with ontology offers a possible answer. Counterfeiting offers serious hazards to organizations and people, resulting in financial losses and possibly personal injury. Current anti-counterfeiting solutions frequently fall short in terms of openness, traceability, and confidence. This is where the suggested anti-counterfeit blockchain technology with ontology comes in. Our technology intends to transform the way items are certified and validated by using the immutability of blockchain and the semantic power of ontology. Because blockchain is decentralized and transparent, every step of the supply chain can be safely documented and traceable, reducing the possibility of counterfeit items entering the market. Furthermore, the incorporation of ontology allows for a more in-depth knowledge of product traits and characteristics, which improves the accuracy and efficiency of the anti-counterfeit system. We want to deliver a comprehensive and dependable solution that enables businesses while protecting consumers from the negative impacts of counterfeiting through this study.

Aim of the work

The goal of this project is to use blockchain technology and ontologies to create an anti-counterfeit system that assures the validity and integrity of items. Our solution intends to address counterfeiting concerns by developing a system architecture that integrates blockchain and ontologies. Each transaction and piece of inventory information is recorded as a block on the blockchain, which acts as a decentralized and unchangeable ledger. These blocks are cryptographically connected, producing a chain that ensures inventory traceability and transparency. Our method eliminates the need for a central authority by leveraging a peer-to-peer network of nodes to ensure that all participants have access to the same verified and up-to-date inventory data. This research intends to help to the battle against counterfeiting by delivering a safe and dependable solution that enables companies and protects consumers from market counterfeit items.

Organization of the thesis

Our master thesis is organized as follows:

- **General Introduction:** We will begin our thesis with a general introduction on the context of this work, problematic and motivation, and aim of the work.
- **Chapter 1 Fake Inventories and Trust System:** This chapter introduces the concept of a Fake Inventory and its impacts in different fields, the current anti-counterfeit measures, including the need of trust system and its advantages.
- **Chapter 2 Blockchain and Ontology:** This chapter introduces the general concepts of Blockchain technology, as well as its history and structure. Furthermore, a broad understanding of ontology and how it is applied in blockchain technology, as well as its obstacles and limits, ending with smart contracts.
- **Chapter 3 Design of The System:** This chapter describes the proposed system's design and hence the development phases.
- **Chapter 4 Implementations and Results:** This chapter discusses the implementation tools, the code, and the results obtained.
- **General Conclusion:** We conclude our thesis with a broad conclusion and gives insights into future work.

CHAPTER 1

FAKE INVENTORIES AND TRUST SYSTEM

1.1 Introduction

Counterfeiting has become one of the most important problems facing people's lives, as it not only threatens the safety of the individual, but also causes huge losses to companies that may exceed one trillion dollars.

While many anti-counterfeiting systems such as authentication or security labels have been implemented, they do not guarantee the authenticity of the product in many cases, as they are limited. For this, a more transparent and reliable way must be developed to solve this problem. One potential solution is the trust system, this system can provide a confidence between consumers and companies in a secure and transparent way to ensure the authenticity of the products. In this chapter, we will discuss the problem of counterfeiting and explore the current used anti-counterfeit measures. We will also analyze the weaknesses of these measures and highlight the need for a trust system to solve counterfeiting problem. Furthermore, we will discuss the benefits of a trust system over traditional measures and introduce our own anti-counterfeit system based on blockchain and ontology in the next chapter.

1.2 Counterfeit Products and Their Impact

Counterfeiting is the process of making products that are identical to original products

with lower quality standards and cheaper prices. This problem can cause huge losses to people and producing companies as well as the economy of countries. The volume of losses due to counterfeiting in 2013 amounted to approximately $1 Trillion and is expected to reach $3 Trillion in 2022[1].

Counterfeit products range from high-end consumer luxury goods such as watches, perfumes or leather goods, to business-to-business products such as machines, chemicals or spare parts, to common consumer products such as toys, pharmaceuticals, cosmetics and foodstuffs. In fact, any IP-protected product can be counterfeited. Some counterfeit products, such as pharmaceuticals, spare parts and toys, are of low quality, and thus create significant health and safety threats.

1.2.1 The impact of counterfeit products

Counterfeiting have a significant and wide-ranging impact on consumers, businesses, economic, environment and society as a whole.

1.2.1.1 The consumers impact of counterfeiting

This problem has a significant impact on consumers, particularly in their health and safety. Counterfeit products, such as pharmaceuticals, cosmetics, and food and beverages, may contain harmful substances that can cause serious health problems or even death. In addition, counterfeit electronic products may be poorly designed and pose a risk of fire or other safety hazards, here is some examples:

- Counterfeit airbags and their components can cause severe malfunctions ranging from non-deployment, under inflation, over inflation to explosion of metal shrapnel during deployment in a crash.
- Counterfeit lithium-ion laptop batteries pose significant risk of extreme heat, self-igniting, and exploding.
- Counterfeit helmets and baby carriers can break.
- Counterfeit prescription drugs may not contain the active ingredient or could lead to accidental overdose.
- Counterfeit cosmetics can cause severe skin reactions.

Consumers who unknowingly purchase counterfeit products may also experience financial loss, as counterfeit products are often sold at prices lower than the genuine products. Furthermore, they may face difficulties in obtaining refunds or compensation for any harm caused by the counterfeit products.

1.2.1.2 The businesses impact of counterfeiting

Counterfeiting not only affects consumers, It create serious problems for authentic businesses, but too many people are unaware of the full extent of the impact of counterfeit goods on brands. Here we will touch on the most harmful points consequences of counterfeiting.

- **Loss of sales:** Starting simple, if you are selling a product, and a counterfeiter competes with you by offering customers a copy of your own product at a lower price, you lose some sales to these cheaper items.

  Not long ago, customers were better able to identify fakes and knew what they were getting when presented with counterfeits. But, in the online world the lines between real and fake are less clear. Counterfeiters are able to operate very effectively online, by stealing a company's designs and branding and even by mixing their knock-off goods with their online product reviews. The images used of the product are sometimes the brand owners real photos, leaving the customer playing a guessing game between what is real and fake.

  Counterfeits have become real competitors, and they can gut a company's cash flow if left to operate freely. Over €26bn is lost annually from counterfeiters in the fashion industry alone, and this problem extends across a multitude of sectors[2].

- **A reputation under attack:** Another side-effect of counterfeiting is that companies take a hit to their reputation. Since many customers are unaware that the product in their hands is a fake, when the knock-off item fails to work correctly, or it falls apart quickly, or it doesn't meet their expectations, then the customer will blame the authentic company.

  Word of mouth is, as we know is one of the most powerful forms of marketing. So, when these customers receive poor copies of a product, word can spread that it is the real product that is not up to scratch.

  Customers will also leave negative reviews online, further cementing this new reputation crisis and furthering the idea that the real brand makes low-quality goods. This does twice as much damage on Amazon; with their combined listings system, a customer can buy a knock-off from the official listing, and come back to the same page to share negative

opinions. These online reviews are used as public indicators for brand quality, so the impact can truly resonate.

A truly egregious example of this is when Fuse Chicken had a counterfeit sent to a reviewer, instead of their actual product. As the company was growing in popularity, the New York Times requested a sample product to write a tech review. When the item arrived, Fuse Chicken got lucky, as the reviewer noticed it was a counterfeit and contacted their company. But, It took an eagle-eyed tech expert to notice the falsification, and it's likely a regular consumer would have been left clueless[3].

- **Authentic companies left to deal with the fallout from counterfeiters:** So, customers look for a product, then buy a counterfeit, and they're rightfully unhappy with it. They demand compensation, either through a refund or a new product, and they go directly to the authentic company to find it. A number of affected companies find themselves in a situation where they're dealing with an irritated customer, complaining about the poor quality of their item, and the customer service agent won't even realise that the product they're talking about is a counterfeit.

  Dealing with returns proves to be a real headache for companies too. They receive faulty copies of their products which have nothing to do with the goods they themselves make and which they haven't warrantied. Or, in the case of Beauty Blender, the customers can assume the counterfeit price is the regular price.

  Shelley Swallow[1] said:

  > *"When we have customers, or new users, who think they're getting a Beauty Blender for a much cheaper price, when they call in and want to talk about that experience, it creates a challenge."*

  Companies are finding themselves caught between a rock and a hard place; between trying to avoid losing time and resources dealing with the sub-par imitations of their products, and doing their best to keep their customer base happy.

- **Compromising long-term trust between businesses:** The damage done by counterfeiters reachers further than relations with consumers. Distributors, retailers and other partners working with companies will often lose trust in legitimate businesses due to the actions of counterfeiters.

[1] Customer Relations Manager of Beauty Blender Company.

If a company have a price agreement with distributors or retailers, and they see listings on online platforms which offer the product at a lower price, their first instinct is often to think they're being ripped off by the company itself. Why would a distributor buy a product wholesale if a brand appeared to simply undercut them on price?

This can be compounded further when partners have exclusivity deals in place; a distributor with exclusive rights to sell in a location will feel betrayed if it looks like the brand itself is in direct competition with them. These effects can seriously harm relationships built over many years of trust and cooperation with other companies[5].

- **Loss of time and money fighting counterfeiters:** When a company discovers they're being targeted by counterfeiters, they generally want to fight back. But this is time-consuming and can be extremely expensive. Resources get pulled away from product development, advertising and anything else the company may have been excited to invest in, and instead must fund lawyers and lawsuits to defend their intellectual property and protect their copyrights. Legitimate companies are forced to spend hundreds of thousands of dollars a year, trying to force back a tidal wave of counterfeits sold on places like Amazon, eBay and Alibaba. CEOs, innovators and company founders must spend their time reacting against these infringements, instead of leading their companies into the future[5].

1.2.1.3 The economic impact of counterfeiting

Counterfeiting can result in lost sales and revenue for legitimate enterprises, which can lead to job losses, reduced investments, and lower tax collections. Counterfeit items frequently violate safety and quality laws, resulting in additional economic consequences in the form of greater healthcare bills and diminished consumer confidence.

Furthermore, counterfeiting can stifle innovation and the creation of new products and technology because counterfeiters can simply replicate existing products without investing in research and development. This can have long-term consequences for economic growth and competitiveness.

Counterfeiting can also distort trade flows and harm legitimate producers, which has international trade repercussions. This might result in diminished foreign investment and trade opportunities, affecting the economic growth and development of countries and regions.

To address the economic effects of counterfeiting, effective and collaborative measures such as expanded intellectual property enforcement, strengthened border controls, and the development of powerful anti-counterfeit technology and systems are required. This can serve to

safeguard firms, customers, and economies from the negative impacts of counterfeiting while also encouraging long-term economic growth and development[6].

1.2.1.4 The social impact of counterfeiting

Counterfeiting has substantial societal consequences that can affect both persons and communities. Counterfeit items frequently fail to meet safety and quality standards, posing major health and safety concerns to users. Counterfeit drugs, for example, may include hazardous substances or wrong dosages, which can have major health repercussions.

Furthermore, counterfeiting can contribute to organized crime and terrorism since counterfeiters may utilize the proceeds from counterfeit sales to fund unlawful actions. This has the potential to promote violence, social discontent, and insecurity.

Counterfeiting also damages consumer and corporate trust, undermining market confidence and making it harder for customers to discern between real and counterfeit items. This can have far-reaching societal consequences, since it can foster a widespread feeling of mistrust and cynicism in society.

Furthermore, it can have an influence on people's lives and communities, particularly in underdeveloped nations where it is common. Counterfeiting may lead to employment and income loss for legitimate producers and workers, as well as diminished investment and economic prospects.

To combat the social impact of this problem, comprehensive solutions that account both the economic and social ramifications of counterfeiting must be developed and implemented. Greater coordination between governments, corporations, and civil society groups may be required to raise consumer awareness, enhance access to information, and strengthen accountability and transparency in the marketplace[5].

1.2.1.5 The environmental impact of counterfeiting

Counterfeiting has major environmental repercussions in addition to negative economic and societal consequences. The manufacture and distribution of counterfeit items sometimes entails the use of low-quality and inexpensive materials, which can lead to increased pollution and waste.

Counterfeit electronics and electrical components, for example, are frequently fabricated using low-quality materials and without sufficient environmental controls. When these goods are incorrectly disposed of, dangerous chemicals such as lead, mercury, and cadmium can have

major environmental and health consequences in the short and long term.

Counterfeit goods also add to the rising issue of electronic garbage (e-waste), which is a major environmental burden throughout the world. E-waste frequently includes dangerous elements and can pollute soil and water, causing major health and environmental dangers.

To combat the environmental effect of counterfeiting, greater environmental responsibility and sustainability must be promoted across the supply chain. This might include creating and implementing more sustainable production techniques, better waste management and recycling procedures, and raising consumer and industry understanding of the environmental effects of counterfeit products[9].

1.2.2 The causes of counterfeit products

The production and distribution of counterfeit products can be attributed to a variety of factors, including:

- **Profit Motives:** The desire to profit is the major motivation for counterfeiting. Counterfeiters attempt to duplicate high-demand items at a lesser cost and resell them at a bigger profit margin.
- **Weak Intellectual Property Protection:** In certain countries, weak intellectual property rules and enforcement make it simpler for counterfeiters to develop and sell counterfeit goods without fear of legal repercussions.
- **Globalization:** The advent of global trade and e-commerce has made it simpler for counterfeiters to reach consumers worldwide. Counterfeitors may readily advertise and sell their counterfeit goods using internet markets and social media channels.
- **Consumer Demand:** Because of the low pricing, consumers are frequently drawn to counterfeit items. The desire to purchase luxury items at a low cost might fuel demand for counterfeit goods.
- **Lack of knowledge:** Due to a lack of knowledge or information regarding the hazards and implications of counterfeit items, consumers may inadvertently acquire counterfeit products.
- **Corruption:** Bribery and corruption may both contribute to the creation and distribution of counterfeit goods. Counterfeiters may persuade officials to look the other way or give them with legal protection.

- **Organized criminal:** Organized criminal networks are often involved in the creation and marketing processes of counterfeit goods. These networks may be involved in other illicit operations, such as drug trafficking and human trafficking, and may utilize counterfeiting earnings to support these activities.

Understanding the reasons of counterfeiting is critical for establishing successful counter-strategies. Counterfeiting efforts should concentrate on improving intellectual property laws and enforcement, increasing consumer awareness, and tackling the underlying issues that lead to the creation and dissemination of counterfeit goods[7].

1.3 Current Anti-Counterfeit Measures

Anti-counterfeit measures are procedures or technologies that are used to prevent, detect, or discourage the manufacture, distribution, and sale of counterfeit goods. There are several sorts of anti-counterfeit measures in use today, including:

- **Security Features:** Holograms, watermarks, serial numbers, and barcodes are examples of security features. These characteristics are used on products and packaging to make counterfeiting more difficult.
- **Authentication labels:** Authentication labels are one-of-a-kind labels that are attached to items and packaging. These labels are intended to be difficult to copy and can be used to authenticate a product's legitimacy.
- **Track and Trace Technologies:** Track and trace technologies track a product through the supply chain by using unique identifiers and serial numbers. Manufacturers and merchants may use these technologies to detect and remove counterfeit items from the market.
- **Legal Measures:** Civil and criminal punishments are available for counterfeiters under legal means. Governments all across the globe have enacted legislation to safeguard intellectual property rights and penalize counterfeiters[8].

1.3.1 Modern day anti-counterfeit solutions

In addition to classic anti-counterfeiting technology such as holograms, RFID tags, and watermarked packaging, corporations and manufacturers may utilize a variety of innovative solutions to safeguard their products against counterfeiting. Here are a couple such examples:

1. **Blockchain technology:** Blockchain technology may be used to generate a digital record of a product's legitimacy, making counterfeiters' reproduction impossible. This technology may be used to trace a product's history through the supply chain, allowing businesses and consumers to verify the legitimacy of a product.

2. **AI and machine learning (ML) algorithms:** AI and ML algorithms may be used to evaluate vast volumes of data, such as product photographs or packaging, to discover distinctive patterns and attributes of real items. This makes counterfeit items simpler to recognize and detect.

3. **NFC (Near Field Communication) tags:** NFC tags are little chips that may be installed in a product and scanned with a smartphone. These tags can be used to hold product information like a unique identification or a link to a website that certifies its legitimacy.

4. **Augmented reality (AR):** AR technology may be used to create interactive experiences for consumers, such as virtual tours of a manufacturing site or product demos. This can assist educate people about a product and make it easier to identify genuine items.

5. **Encryption:** By encrypting information and producing a digital signature, encryption may be used to ensure the validity of a product. This can help to validate the authenticity of a product and make counterfeiting more difficult.

These contemporary methods may be used with classic anti-counterfeiting technology to provide an extra layer of security for items. Companies and manufacturers may better safeguard their products from counterfeiting and maintain consumer safety and happiness by staying up to date on emerging anti-counterfeiting technologies[8].

1.3.2 The increasing demand for anti-counterfeiting solutions in recent years

In recent years, the need for anti-counterfeiting solutions has has grown tremendously as counterfeiting has become an increasingly prevalent concern in the worldwide economy. The advent of e-commerce, as well as the increased ease of producing and distributing counterfeit items, have all contributed to the problem. As a result, businesses and manufacturers are seeking for solutions to safeguard their products and trademarks against counterfeiting.

According to a MarketsandMarkets report, the global anti-counterfeiting packaging market was valued at USD 117.2 billion in 2021 and is projected to reach USD 211.3 billion by 2026,

growing at a CAGR 12.5% from 2021 to 2026. The market is witness to growth due to increasing focus of manufacturers on brand protection to reduce counterfeiting[4].

The increasing adoption of anti-counterfeiting technology by companies and manufacturers is a positive trend, as it helps protect consumers and businesses from the dangers and harms of counterfeit goods. It also helps protect the integrity of legitimate brands and combat organized crime groups often involved in counterfeiting activities.

1.4 Need for a Trust System

Despite the introduction of many anti-counterfeit measures, counterfeit items continue to be a serious concern in the worldwide economy. Because of the limitations and problems of these procedures, there is a need for a more accurate and transparent method of tracking items and ensuring their authenticity. This is where a trust mechanism may help.

A trust system is a safe and transparent mechanism to track and authenticate things. Consumers and companies may have confidence that the things they buy and sell are authentic by implementing a trust system.

Greater transparency, higher security, and improved traceability are among the benefits of a trust system above standard anti-counterfeit procedures. These advantages make a trust system an appealing choice for combatting counterfeit goods.

In the next chapter, we will introduce our own anti-counterfeit system based on blockchain and ontology, which utilizes a trust system to provide a more reliable and transparent way to track products and ensure their authenticity.

1.5 Advantages of a Trust System

A trust system has a number of advantages over typical anti-counterfeiting systems. Among these benefits are the following:

1. **Greater transparency:** A trust system may increase supply chain transparency, allowing consumers and companies to readily follow the transit of items from the manufacturer to the end user. This transparency can serve to boost consumer and corporate confidence and guarantee that the items being offered are authentic.

2. **Enhanced security:** A trust system can use sophisticated security features like encryption and digital signatures to secure data kept within the system from illegal access or ma-

nipulation. This can assist to prevent counterfeit items from accessing the market and provide a secure method of product authentication.

3. **Improved traceability:** A trust system may enhance product traceability, allowing firms to quickly track product movement and detect any possible concerns in the supply chain. This can aid in the prevention of counterfeit items entering the market as well as the identification of the source of any problems that do develop.

4. **Cost-effective:** A trust system may be less expensive than typical anti-counterfeit methods such as package security features or authentication labels. Once installed, the system may be quickly scaled to handle more items or users, making it a more efficient and cost-effective option.

Overall, a trust system can provide a more dependable and transparent method of tracking products and ensuring their authenticity, making it an appealing alternative for countering counterfeit goods.

1.6 Conclusion

To summarize, counterfeit items pose a huge danger to consumers, businesses, and economies across the world. While existing anti-counterfeit methods have been deployed, they have limitations and problems, and are frequently insufficient to ensure product authenticity. As a result, there is a need for a more dependable and transparent method of tracking items and ensuring their validity. A trust system can help consumers and companies create trust by providing a safe and transparent mechanism to tackle counterfeit items. In the next chapter, we will present our unique anti-counterfeit solution based on blockchain and ontology, which has various benefits over existing approaches.

CHAPTER 2

BLOCKCHAIN AND ONTOLOGY

2.1 Introduction

Blockchain technology has evolved as a strong and inventive tool for enabling safe and transparent digital transactions in a variety of areas, including healthcare, supply chain management, and finance, in recent years. This technique was designed originally for use in cryptocurrencies, but its potential uses have expanded beyond digital money. Blockchain technology offers a viable alternative for confirming product authenticity and avoiding counterfeiting in the context of anti-counterfeit measures. Blockchain's distinct characteristics, such as its decentralized and unchangeable record, make it perfect for tracing and confirming product origins along the supply chain.

Furthermore, combining ontology, a field of computer science concerned with the formal representation of information, with blockchain technology can improve the security and accuracy of the anti-counterfeit system. Blockchain-based anti-counterfeit systems can better detect fraud and maintain transparency by leveraging ontology to generate a consistent and shared understanding of product information.

In this chapter, we will look at the benefits of blockchain technology and ontology in anti-counterfeiting efforts, as well as how these two technologies may be combined to provide a more robust and effective solution for assuring product authenticity.

2.2 Blockchain Technology

To identify areas where blockchain technology may be applied, one can look for situations where a middleman is necessary to facilitate trust. Trust is crucial in various areas, such as the transfer of money, voting, land records, IP rights, and identity. By serving as a trusted record-keeping system, blockchain software can replace the middleman[10].

2.2.1 Definition of blockchain technology

The concept of Blockchain was appeared in 2008 with the creation of Bitcoin, famously authored by an unknown individual using the pseudonym Satoshi Nakamoto. Since then, numerous definitions for Blockchain have been put forth, including the following examples:

- A blockchain is a distributed database or ledger that is shared among the nodes of a computer network. As a database, a blockchain stores information electronically in digital format. Blockchains are best known for their crucial role in cryptocurrency systems, such as Bitcoin, for maintaining a secure and decentralized record of transactions. The innovation with a blockchain is that it guarantees the fidelity and security of a record of data and generates trust without the need for a trusted third party.[11]

- A blockchain is a peer-to-peer digital ledger of transactions that may be publicly or privately distributed to all users (and therefore is said to be decentralized and distributed). Blockchain technology uses cryptography and a consensus mechanism to verify transactions, which ensures the legitimacy of a transaction, prevents double- spending, and allows for high-value transactions in a trustless environment. A blockchain offers transparency and eliminates the need for intermediaries or third-party administrators.[12]

- A blockchain refers to a decentralized and digital record of transactions that is replicated in real-time across a network of nodes or computers. Before adding a new transaction as a block at the end of the chain, all transactions must undergo cryptographic validation through a consensus mechanism executed by the nodes. The absence of a central authority that approves transactions is why blockchain is also known as a trustless peer-to-peer mechanism.[13]

- A blockchain is a continuously growing chain of interconnected blocks that store transactions. This platform uses a decentralized approach, enabling the distribution of information so that every piece of data has shared ownership.[14]

2.2.2 History of blockchain

The impact of blockchain technology on various sectors, including finance, manufacturing, and education, makes it one of the most significant innovations of the 21st century. However, what many don't realize is that the history of blockchain dates back to the early 1990s.

1991: Stuart Haber and W. Scott Stornetta publish a paper on a cryptographically secured chain of blocks, which laid the groundwork for what would become the blockchain technology.[15]

1992: Merkle Trees were incorporated into the design, which makes blockchain more efficient by allowing several documents to be collected into one block. Merkle Trees are used to create a 'secured chain of blocks.' It stored a series of data records, and each data records connected to the one before it. The newest record in this chain contains the history of the entire chain.[16]

2004: computer scientist and cryptographic activist Hal Finney (Harold Thomas Finney II) introduced a system called RPoW, Reusable Proof Of Work. The system worked by receiving a non-exchangeable or a non-fungible Hashcash based proof of work token and in return created an RSA-signed token that could then be transferred from person to person.[17]

2008: Satoshi Nakamoto conceptualized the theory of distributed blockchains. He improves the design uniquely to add blocks to the initial chain without requiring them to be signed by trusted parties. The modified trees would contain a secure history of data exchanges. It utilizes a peer-to-peer network for timestamping and verifying each exchange. It could be managed autonomously without requiring a central authority. These improvements were so beneficial that makes blockchains as the backbone of cryptocurrencies.[16]

2009: On January 3rd 2009, the Bitcoin network was created when Satoshi Nakamato (the project's mysterious creator) mined the "Genesis" block. The 50 bitcoin coinbase reward is unredeemable, as it was omitted from the transaction database. This means any attempt to spend it would be rejected by the network. Whether this was intentional or not still remains unknown.[18]

2013: Vitalik Buterin, a programmer and a co-founder of the Bitcoin Magazine stated that Bitcoin needed a scripting language for building decentralized applications. Failing to gain agreement in the community, Vitalik started the development of a new blockchain-based distributed computing platform, Ethereum, that featured a scripting functionality, called smart contracts.

Smart contracts are programs or scripts that are deployed and executed on the Ethereum blockchain, they can be used for example to make a transaction if certain conditions are met. Smart contracts are written in specific programming languages and compiled into bytecode, which a decentralized Turing-complete virtual machine, called the Ethereum virtual machine

(EVM) can then read and execute.

Developers are also able to create and publish applications that run inside Ethereum blockchain. These applications are usually referred to as DApps (decentralized applications) and there are already hundreds of DApps running in the Ethereum blockchain, including social media platforms, gambling applications, and financial exchanges.

The cryptocurrency of Ethereum is called Ether, it can be transferred between accounts and is used to pay the fees for the computational power used when executing smart contracts[17].

2015: The first version of the Ethereum blockchain is released, and the first Initial Coin Offering (ICO) takes place on the Ethereum platform.[19]

2018: Many countries begin to regulate cryptocurrencies, and several major companies announce plans to use blockchain technology for various purposes.[20]

2021: The value of Bitcoin and other cryptocurrencies reaches new highs, and many major financial institutions begin to offer cryptocurrency-related services. Governments and regulatory bodies also increase their scrutiny and oversight of the industry.[21]

2.2.3 Blockchain architecture

The Blockchain technology is structured as a sequence of blocks containing transactions arranged in a specific order. Each block stores transaction data and is linked to the preceding and succeeding blocks, forming a chain. The architectural elements of Blockchain were initially standardized and later modified by different companies, resulting in various projects such as Bitcoin, Ethereum, Hyperledger, among others. These projects share the fundamental components of Blockchain architecture, as depicted in Figure 2.1.

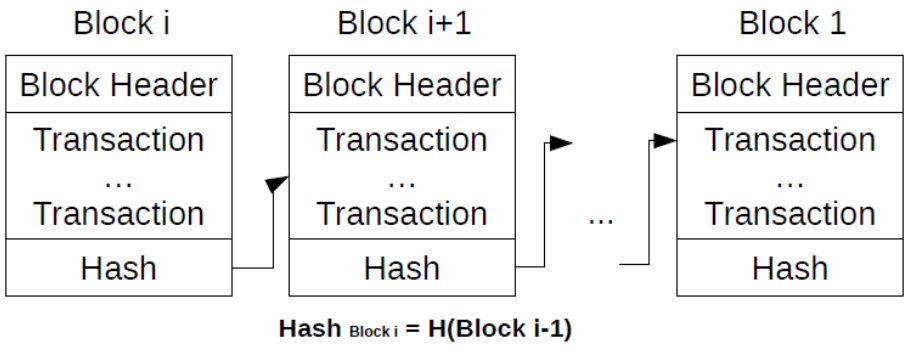


Figure 2.1: Blockchain structure [22]

2.2.3.1 Transaction

Transaction is a digital record of a transfer of data or value between two or more parties.

Transactions are at the core of blockchain technology and are the building blocks of the distributed ledger.
A transaction in a blockchain network typically includes the following components:

- **Inputs:** The inputs represent the funds or assets being transferred in the transaction. In a cryptocurrency transaction, the inputs consist of unspent transaction outputs (UTXOs) from previous transactions.
- **Outputs:** The outputs represent the destination addresses where the transferred funds or assets are being sent. Each output specifies the amount being sent and the public key or address of the recipient.
- **Digital signature:** To verify the authenticity and integrity of a transaction, it is digitally signed using the private key of the sender.
- **Transaction ID:** Each transaction is assigned a unique identifier, called a transaction ID or TXID, which is used to track and verify the status of the transaction on the blockchain network.

Once a transaction is initiated, it is broadcast to the network of nodes for validation and confirmation. The nodes in the network use consensus mechanisms to verify the validity of the transaction and add it to the blockchain ledger. Once a transaction is confirmed and added to the blockchain, it becomes immutable and cannot be altered or deleted.
Overall, transactions in blockchain technology provide a secure, transparent, and decentralized way to transfer data and value between parties without the need for intermediaries or central authorities.

2.2.3.2 Blocks

Blocks in a blockchain contain a set of transactions that have been validated and verified by the network of nodes. Once a block is added to the chain, it cannot be altered or deleted, ensuring that the data stored in the blockchain is secure and tamper-proof.
Each block in a blockchain contains a header, which contains metadata about the block, such as the timestamp, the hash of the previous block, and a nonce (a random number used in the mining process). The block also contains a list of transactions, each of which includes information about the sender, recipient, amount, and other details.
To add a new block to the blockchain, a process called mining is used. Miners use powerful computers to solve complex mathematical problems that validate transactions and create new

blocks. Once a block is mined, it is added to the blockchain and becomes a permanent part of the ledger.

Overall, blocks are a crucial component of blockchain technology, as they enable the secure and transparent recording of transactions on a distributed network without the need for a central authority or intermediary. Figure 2.2 shows components of block.

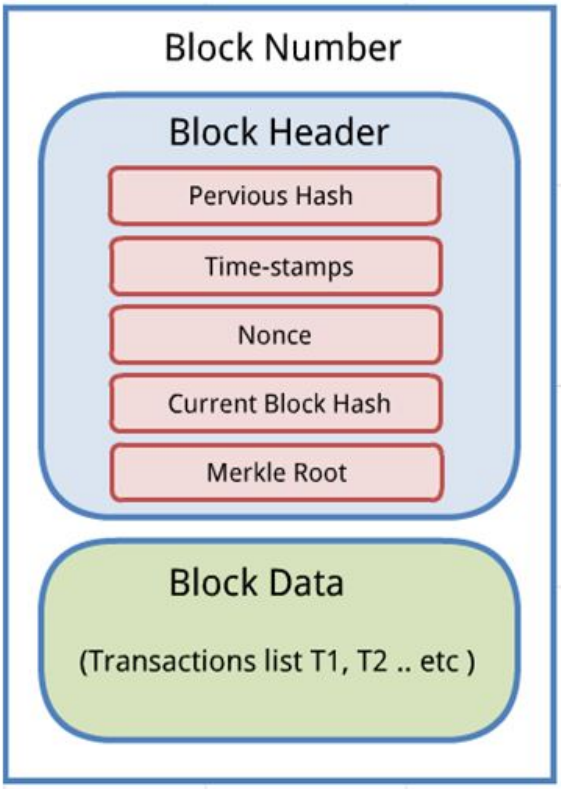


Figure 2.2: Components of Block

2.2.3.3 Consensus

Blockchain technology relies heavily on consensus. It is the mechanism through which nodes in a blockchain network agree on the present state of the blockchain ledger. This agreement assures that all network participants have the same view of the ledger and that no fraudulent or illegal transactions are included[23]. Consensus methods are meant to withstand assaults like as double-spending and Sybil attacks, which might jeopardize the ledger's integrity.

Several consensus algorithms, including Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), and Practical Byzantine Fault Tolerance (PBFT), have been developed and implemented in various blockchain systems. Each algorithm has advantages and disadvantages, and the consensus mechanism chosen relies on the unique use case and desired trade-offs between security, scalability, and decentralization.

The PoW algorithm, which is employed in the Bitcoin blockchain, is well-known for its great

security but also for its enormous energy consumption. The Ethereum blockchain employs the PoS algorithm, which attempts to minimize energy usage by replacing the energy-intensive mining process with a more efficient staking mechanism. The DPoS method, which is employed in blockchains such as EOS and BitShares, validates transactions using a limited set of trustworthy nodes and is noted for its scalability and speed. Finally, in permissioned blockchains, the PBFT method is employed for high-speed transactions with a reduced number of trustworthy nodes[24].

2.2.3.3.1 Proof of work (PoW)

Proof of Work is often referred to as the Nakamoto Consensus[25]. It is an algorithm that takes exponential time to compute as the complexity increases and can be confirmed in a single hash, as shown in Algorithm 1. It was popularized by Bitcoin. Proof of Work achieves consensus by utilizing compute resources. Generally speaking, the consensus is that everyone agrees on the same issue, such as an election. It chooses who will produce the next block. In PoW, the more computation resources invested, the more likely the next block will be generated.

Proof of Work, on the other hand, is not merely a random function weighted by computation capability. The majority of computation on block generation is provided by global mining, which keeps the blockchain safe. However, PoW is thought to consume too much electricity. It is because of the enormous incentive of the Bitcoin price. We think that a blockchain with PoW can be designed for low energy costs and reasonable security[26].

Algorithm 1 Proof of Work [26]

Get miner's identity id
Get current block's $hash$
Get next block's $data$
Calculate current difficulty D based on history blocks
Let $nonce = 0$
repeat
    if Got other's new block **then**
        Exit and restart this algorithm based on the new block hash
    end if
    Let $nonce = nonce + 1$
until Hash$(hash||data||id||nonce) < D$
Broadcast the new block worldwide

2.2.3.3.2 Proof of stake (PoS)

Proof-of-Stake (PoS), a concept first proposed by Sunny Kind and Scott Nadal in 2012, is a consensus mechanism design that allows network validators to participate in the transaction

validation process used to maintain the network if they stake or lock the network's token. The incentives of validators and the network are aligned by forcing validators to lock tokens; a validator wants the network to continue running and would not want to attack the network since the value of the validator's underlying locked capital would be diminished if a problem occurred. Proof-of-Stake uses less energy than other methods. Proof-of-Work is so-called because, rather than requiring processing capacity to solve a mathematical problem, it distributes the authority to validate transactions among multiple validators depending on the proportion of tokens locked by each validator[27].

2.2.3.3.3 Proof of authority (PoA)

Proof of Authority (PoA) is a family of permissioned blockchain consensus algorithms that have gained popularity due to increased performance over traditional BFT algorithms due to lighter message exchanges. PoA was first proposed as a component of the Ethereum ecosystem for private networks.

The authorities are a group of N trustworthy nodes that PoA algorithms rely on. Each authority is identifiable by a unique id, and a majority of them, meaning at least N/2 + 1, are presumed to be honest. To order the transactions issued by clients, the authorities conduct a consensus. Consensus in PoA algorithms is based on a mining rotation scheme, a commonly used method for distributing block generation duty fairly across authority. Time is separated by stages, each of which has a mining leader elected[28].

2.2.3.4 The Hash

Hash function or also known as one-way function (one-way-function), message digest, fingerprint, compression function, and message authentication code (MAC)[29]. A hash function is a mathematical function that converts input of variable sizes into output, which is generally a set length hexadecimal. Hash is often written using a mix of numbers (0 to 9) and letters (a to f). The hash value or message digest is the result of the hash function.[30].

The hash connects each block on the Blockchain to the preceding block. As a result, the entire Blockchain transaction cannot be modified or removed. As a result, the Blockchain is protected from hacking.

To minimize space, direct note stores simply the hash rather than the assets or note on the blockchain. The increased number of nodes in blockchain allows for higher energy and storage expenses while also increasing security[31].

The blockchain data structure is unchangeable and can only be added to. Every data in this

Blockchain is linked to one another; if one of the data blocks changes, it will affect the next data.

2.2.3.5 Miners or nodes

The nodes, known as miners, compete to solve a difficult cryptographic puzzle using a brute force search technique. The first miner to find a way to include pending transactions in a block gets rewarded with newly minted crypto-coins. This incentive compensates for the operational costs of mining, which are mostly caused by the use of power[32].

2.2.3.6 The Chain

The blocks are linked together by each including the hash digest header of the previous block, constituting the blockchain.If a previously released block was modified, the hash would be different.As a result, all subsequent blocks will have different hashes since they contain hashing from the prior block. This allows for the detection and rejection of changed blocks. Figure 2.3 depicts a generic block chain[33].

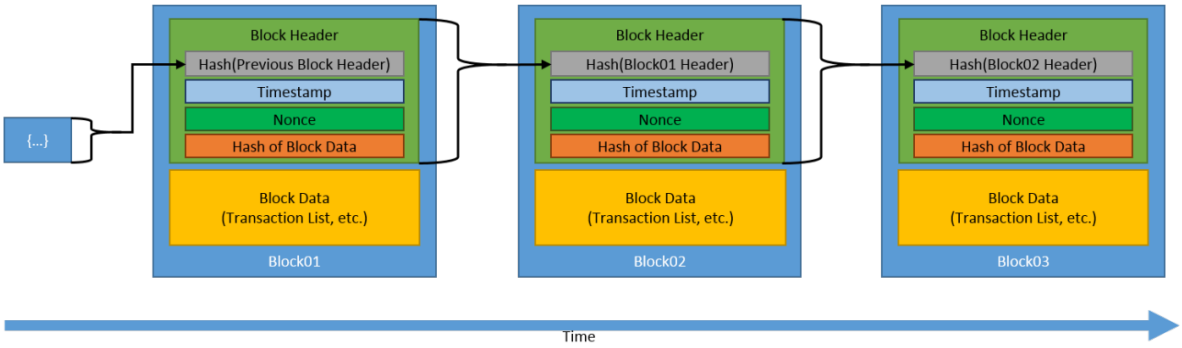


Figure 2.3: Generic chain of blocks[34].

2.3 Ontology

Ontology is a philosophical field that studies the nature of existence and the links between things. Ontology is a systematic and formal technique of representing information about a domain in computer science. Ontology is significant in the context of blockchain technology because it defines the meaning and connections of data recorded on the blockchain.

Ontology provides a method for modeling the entities and relationships in a system, such as data types, attributes, and the rules that govern their interactions. Ontology helps to guarantee that data is consistent and interoperable across multiple systems and applications by providing a standard vocabulary and set of rules.

Ontology can be used to specify the categories of items, their attributes, and the rules controlling their distribution and authentication in the context of anti-counterfeit systems. An ontology for pharmaceutical items, for example, may include information about the producer, the active chemicals, and the suggested dose, as well as criteria for confirming the product's legitimacy.

Anti-counterfeit systems may ensure that information is accurate, consistent, and trustworthy by employing ontology to describe the structure and meaning of data on the blockchain. This can assist to develop confidence among stakeholders and ensure the system's integrity[36, 37].

2.3.1 How ontology is used in Blockchain technology

One use of ontology in blockchain technology is in supply chain management. It is feasible to readily identify and monitor items as they move through the supply chain by developing a consistent ontology for the products being tracked. This can assist to prevent counterfeiting and assure the authenticity of items, which is vital in industries like as medicines and luxury goods.

Another use of ontology in blockchain technology is in the field of identity management. It is feasible to develop a more secure and trustworthy identity verification system by employing an ontology-based approach to establishing and maintaining identities. This is especially crucial in applications such as financial services, where precise and reliable identity verification is critical for regulatory compliance and fraud prevention.

Ontology can potentially play an important role in the development of blockchain-based decentralized applications (dApps). These dApps run on a blockchain network with no centralized point of control. Using ontology, developers can create a standardized framework for specifying the rules and interactions between the application's various components. This can assist to make the development process more efficient and productive while also ensuring that the dApp runs smoothly and safely. Developers may use ontology to create dApps that give unique and inventive answers to a variety of real-world situations[38, 39].
becomes possible to create more intelligent and sophisticated systems that can make better use of the data they collect.

2.3.2 Benefits of ontology in anti-counterfeiting systems

Ontology, as a knowledge representation and semantic web technology, has several benefits when applied to anti-counterfeiting systems. One of the primary advantages is its ability to facilitate interoperability between different systems and data sources, enabling seamless data

sharing and collaboration between stakeholders. This interoperability can help to increase the efficiency and accuracy of anti-counterfeiting measures, as well as reduce the risk of errors and discrepancies in data.

Another advantage of using ontology in anti-counterfeiting systems is that it allows for automated reasoning and decision-making. Ontology may help automated systems make more educated judgments based on available data by specifying the relationships and rules between distinct ideas and things. This can assist to lessen the danger of human mistake while also speeding up the identification and reaction to counterfeit items.

Overall, ontology has substantial benefits in anti-counterfeiting systems, and it may assist to increase the efficiency, accuracy, and efficacy of anti-counterfeiting methods. Ontology may provide a standardized and interoperable framework for data sharing, decision-making, and machine learning by utilizing the power of semantic web technologies, allowing stakeholders to collaborate more effectively to tackle the rising issue of counterfeit products.

2.3.3 Challenges and limitations of ontology in blockchain-based anti-counterfeiting systems

Anti-counterfeiting systems that use ontologies provide various advantages in the battle against counterfeit goods. Nonetheless, these systems present a number of challenges and limitations that must be considered.

The development of an ontology capable of effectively capturing the intricate relationships between the various entities participating in a supply chain is one of the most challenging issues. This necessitates a thorough awareness of the business processes involved, as well as extensive domain-specific knowledge. Furthermore, the ontology must be designed in such a way that it allows for scalability and interoperability with other systems.

Another challenge is the maintenance and updating of the ontology. As the supply chain and the products involved evolve over time, the ontology needs to be regularly updated to ensure that it remains accurate and relevant. This can be a time-consuming and expensive process.

Furthermore, we must consider the risks associated with using an ontology-based system. One such danger is the potential breach of data privacy. This is because, in order to identify counterfeit items, the system may need to access sensitive information such as product designs, manufacturing methods, and distribution routes. Sharing such information may be a source of worry for certain firms.

Despite these challenges, ontology-based anti-counterfeiting systems have great potential in

preventing the spread of counterfeit products. The benefits of using ontology in blockchain-based anti-counterfeiting systems outweigh the challenges, making it a promising technology in the fight against counterfeit products.

2.4 Smart Contracts

A smart contract is a computer program that performs the terms contained inside it and is stored on the decentralized blockchain network. The contract is only activated when an external event or a predetermined condition triggers it. Smart contracts can be written in Solidity, Serpent, or LLL, contract-oriented high-level languages designed for the Ethereum Virtual Machine (EVM). The Ethereum Virtual Machine (EVM) is a Turing complete program that operates on the Ethereum network. Given enough time and memory, it allows anyone to run any program written in any programming language. We picked Solidity to create our smart contracts for our implementation. Solidity is statically typed and, among other things, enables inheritance, libraries, and sophisticated user-defined types. Figure 2.4 shows how the solidity compiler (solc) creates byte-level code for an input smart contract that can be deployed using the EVM[35].

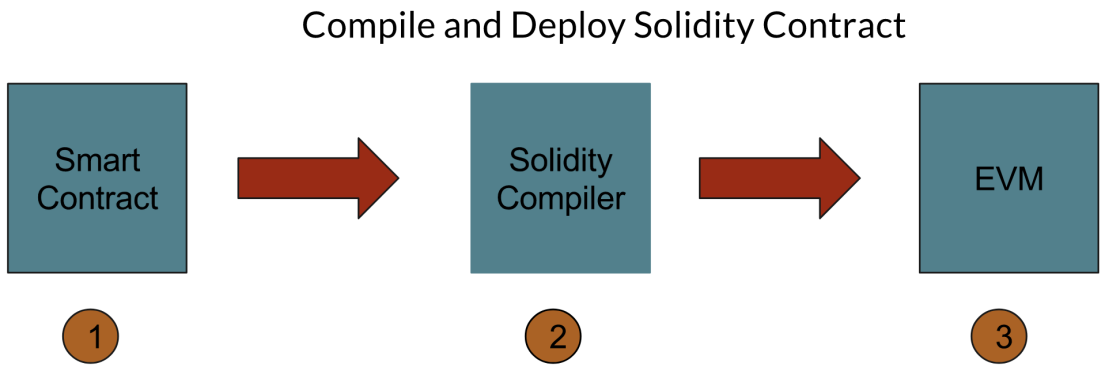


Figure 2.4: Deploying a Smart Contract on Blockchain[35].

2.5 Conclusion

In this chapter, we looked at how blockchain and ontology technologies may be utilized to tackle the problem of counterfeit goods. The obstacles and constraints of incorporating ontology in blockchain-based anti-counterfeiting systems, such as the necessity for a uniform ontology and the difficulty of ontology integration, were explored.

In conclusion, the integration of blockchain and ontology has the potential to transform the battle against counterfeit items. Despite some challenges, the benefits of using these tech-

nologies far outweigh the costs. We may anticipate to see more new solutions that will improve the marketplace's safety and reliability by harnessing the capabilities of blockchain and ontology.

CHAPTER 3

DESIGN OF THE SYSTEM

3.1 Introduction

In this chapter we will dig into the complexities of constructing the anti-counterfeit blockchain-based system that we presented in previous chapters . The solution was created to solve the growing issues of detecting counterfeit items by leveraging blockchain technology and ontology. We will examine the system's many components in depth, including the web application that allows for account administration and inventories addition, the mobile application that allows for inventory verification. Furthermore, we will look at the design considerations and roadblocks found throughout the development process. The inspiration for this chapter derives from the necessity to thoroughly understand the system's architecture and how it functions in order to create a safe and trustworthy supply chain solution. By the end of this chapter, you will have a deeper understanding of how the system was designed and how it operates.

3.2 Motivation

What are the challenges in detecting counterfeit products?

Consumers and companies confront various obstacles in detecting counterfeit items, including a lack of openness in the supply chain, difficulty determining product authenticity, and the ease with which products may be replicated. These difficulties can result in reputational impact, financial loss, and potential harm to end users as we discussed before.

How can blockchain be used to create a trusted anti-counterfeit system?

By providing a decentralized, tamper-proof ledger that records all transactions between members, blockchain technology can help to construct a trustworthy and transparent supply chain system. This generates an indelible record of the product's path from origin to end user, confirming its validity and integrity.

What is the role of ontology in developing anti-counterfeiting solutions?

Ontology plays a significant role in developing anti-counterfeiting solutions by providing a formal framework for describing the relationships between various entities in the system, such as products and certificates. This enables the creation of a standardized and machine-readable format for the data, facilitating its efficient sharing and processing.

How can SWRL rules help ensure that only authentic products are added to the blockchain?

SWRL rules can help ensure that only authentic products are added to the blockchain by specifying the conditions that a product must meet to be considered genuine. For example, a rule may state that a product must have a unique serial number and be manufactured by an authorized factor to be added to the blockchain.

What are the potential benefits of using a blockchain-based anti-counterfeiting system with ontology?

Using a blockchain-based anti-counterfeiting system with ontology can bring numerous benefits, such as increased transparency and trust in the supply chain, better protection against counterfeit products, reduced costs related to fraud and product recalls, and improved customer satisfaction. By utilizing the immutable nature of blockchain and the semantic knowledge representation of ontology, businesses can create a reliable and secure ecosystem where every participant can easily and quickly verify the authenticity of products.

3.3 System Architecture

In this section we will take an overview of the structure of the system and then delve into its parts in detail. we can better appreciate how the various components work together to create a trusted and transparent anti-counterfeit system.

3.3.1 Global overview

Our system is made up of several components that work together to produce a trusted and transparent system. The system's core is blockchain technology, which is used to store information about inventories (products, certificates) and their legitimacy in a decentralized and tamper-proof way. This data is maintained through a web application that allows businesses to establish accounts and add items after verifying them against an ontology model using SWRL rules. Consumers may then use a smartphone app to scan the QR code of the inventories and verify its validity using the information recorded on the blockchain.

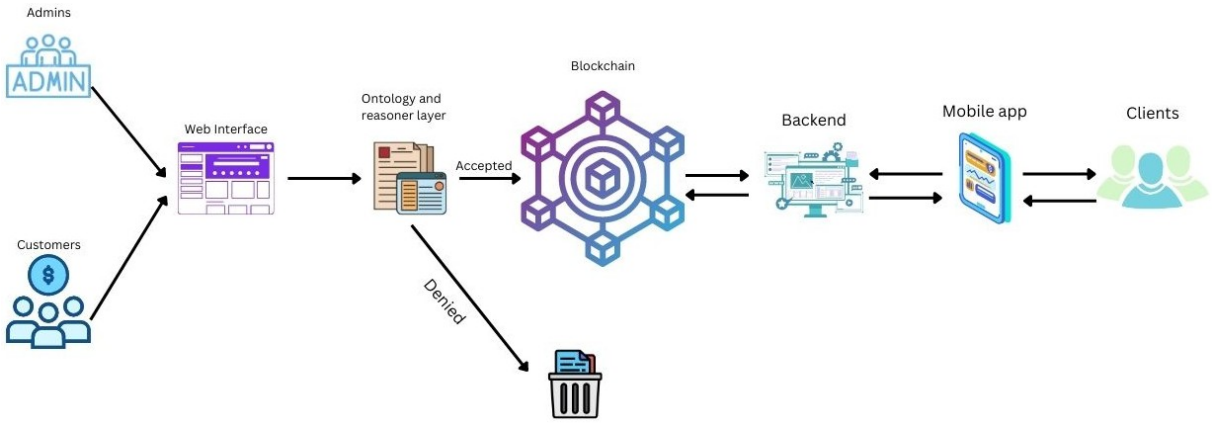


Figure 3.1: Global overview of the system.

The figure 3.1 shows three main actors which are:

- **Admins:** Have a direct access to the system through dashboard, Their role is represented by the approval or rejection of the customers's accounts based on proof of the identity, web application management and provide a customer support.
- **Customers:** They can access the system through a web interface, It provides the ability to sign up and place an order to join the system, They also can add their products or certificates to be confirmed.
- **Clients:** We provides a mobile application that have the ability to scan the QR codes and

send its information to the system through a backend then it returns whether the inventory is genuine or not based on its existence on blockchain.

3.3.2 Architecture of each subsystem

The system represents a set of modules, so we will illustrate all these units.

3.3.2.1 Registration

This process is concerned with customers who want to authenticate their products or certificates.

First of all, The customer has to sign up as usually with email and password, After that, he has to place an order to confirm his identity by providing a real documents. When the identity is successfully verified, a smart contract is published in the blockchain by that he can add his inventories to be authenticated, as shown in Figure 3.2.

Figure 3.2: Registration process.

In the joining order process, the customer should verify his MetaMask wallet as an additional layer to validate the customer and it will be used whenever he going to add an inventory.

3.3.2.2 Adding an inventory

Once the customer has created a verified account, they can start adding their inventories through a series of stages that have been designed to ensure the authenticity and validity of the information being added. The first stage involves validating the MetaMask wallet that has been linked during the account creation process this stage ensures only the customer can add any inventory under their company name, and no one can manipulate the system. If the wallet validation failed, the process will be denied. If it's successful, the process will move on to the next stage which involves adding the main information of inventory, such as product specification and certificate details to validate it through ontology layer using pre-defined SWRL rules (we will delve in it in separated section). If it's failed, the process will be failed. Otherwise, it will move on to the last stage which is hashing the information and then store it as plain text in blockchain with its hashed value to ensure security and tamper-proofing.

Once all the stages have been completed, the system generates a QR Code that can be included in the inventory (product or certificate). This QR Code contains all the necessary information about the inventory that has been added to the blockchain, and clients can use it to verify the authenticity of the product using the mobile application as shown in Figure 3.3. This multi-stage process ensures that only authentic and verified information is added to the system and that the supply chain remains transparent and secure.

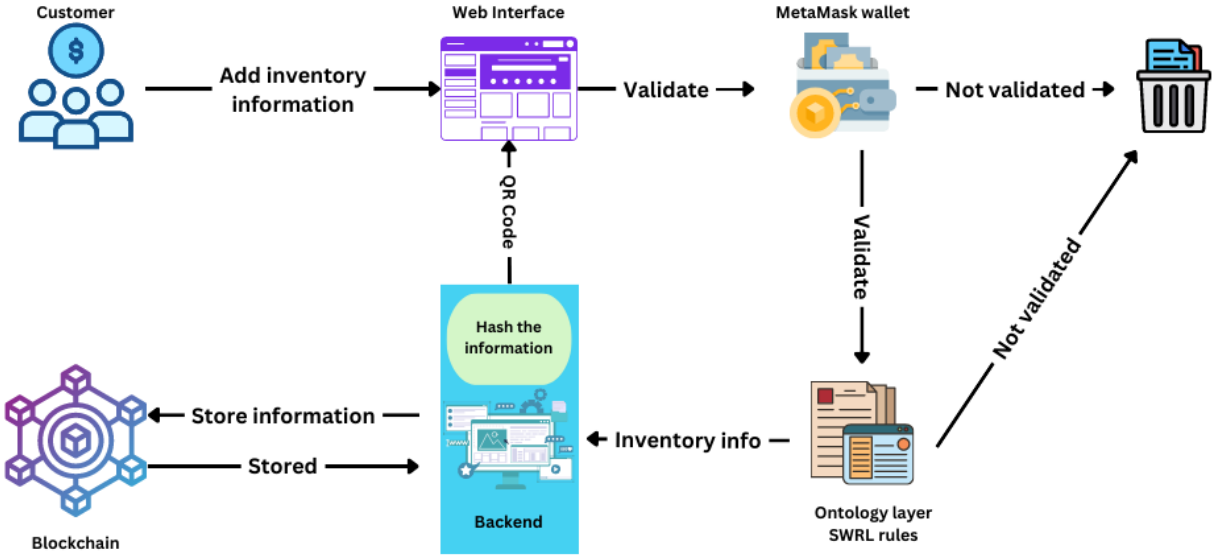


Figure 3.3: Adding inventory process.

3.3.2.3 Ontology layer

We have integrated an ontology layer in our system to enhance its effectiveness and combat counterfeiting. The role of this layer is to ensure the correctness of the information entered to ensure that a real inventory is recorded and to reduce as much as possible the possibility of forgery. The following table 3.1 shows some of the counterfeiting scenarios that our system can effectively prevent.

Scenario	Actor
corrupt employee/teacher issues a certificate without submitting the assignments or required studies done.	Employee/teacher
Hacker get into the university/company system and add certificate.	Hacker
Company officials insert wrong data of the origin product.	Company officials
Company officials insert one origin product, and then after the system generates the QR-code the official print it on many products with the same specs.	Company officials

Table 3.1: Possible fraud scenarios

To delve into how the ontology layer works, Figure 3.4 shows the structured knowledge framework that captures and defines domain-specific concepts, such as product attributes, certifications and the different actors. Table 3.2 illustrate the different properties with its types in the mentioned concepts.

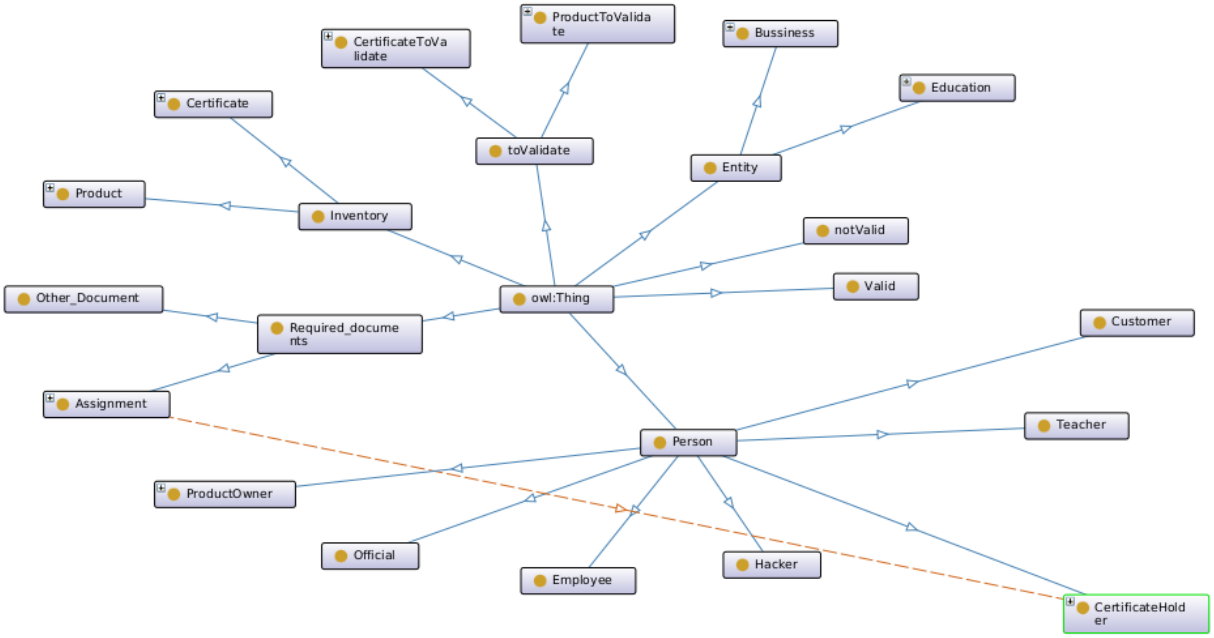


Figure 3.4: Ontology structure

Property	Type	Domain
Brand	xsd:string	Product or ProductToValidate
Model	xsd:string	Product or ProductToValidate
IdProduct	xsd:integer	Product or ProductToValidate
CompanyName	xsd:string	Company
CertificateTitle	xsd:string	Certificate or CertificateToValidate
IdCertificate	xsd:integer	Certificate or CertificateToValidate
IssueTimestamp	xsd:integer	Certificate or CertificateToValidate
FirstName	xsd:string	Person
LastName	xsd:string	Person

Table 3.2: Object properties

As we can see, we defined a conception for each inventory that our system deals with (certificates and products) and two other concepts for valid inventory and for invalid. We used the Pellet reasoner[1] to take advantage of its effectiveness in classification based on pre-defined rules. The SWRL rule presented in Listing 3.1 is designed to validate certificates in our system. The rule specifies the conditions that must be satisfied for a certificate (**?c**) and its corresponding certificate to validate (**?cv**) to be considered valid.

[1]Pellet: is an open-source Java based OWL 2 reasoner.

```
CertificateToValidate(?cv) ^ Certificate(?c) ^ holdBy(?cv, ?a) ^ holdBy(?c,
    ?a) ^ IdCertificate(?c, ?idC1) ^ IdCertificate(?cv, ?idC2) ^
   Certificate_title(?c, ?t1) ^ Certificate_title(?cv, ?t2) ^
   issue_timestamp(?c, ?date1) ^ issue_timestamp(?cv, ?date2) ^ swrlb:equal
   (?date1, ?date2) ^ swrlb:equal(?idC1, ?idC2) ^ swrlb:
   stringEqualIgnoreCase(?t1, ?t2) -> Valid(?cv)
```

Listing 3.1: SWRL rule for validating certificates

The rule specifies the conditions that must be satisfied for a certificate (**?c**) and its corresponding certificate to validate (**?cv**) to be considered valid.
The rule checks various properties of the certificates, including the holder (**holdBy**), the ID of the certificate (**IdCertificate**), the title of the certificate (**Certificate_title**), and the issue timestamp (**issue_timestamp**). It ensures that the holdBy relationship is established for both the certificate and its corresponding certificate to validate, and that the ID, title, and issue timestamp of both certificates are equal.
Additionally, the rule utilizes built-in SWRL functions such as **swrlb:equal** to compare values and **swrlb:stringEqualIgnoreCase** to perform case-insensitive string comparison.
If all the conditions specified in the rule are met, the conclusion is reached that the certificate to validate (**?cv**) is valid, and the **Valid** relation is inferred.

The second SWRL rule, depicted in Listing 3.2, is designed to validate products in our system. This rule ensures that a product (**?p**) and its corresponding product to validate (**?pv**) are considered valid based on certain conditions.

```
Product(?p) ^ ProductToValidate(?pv) ^ manufacturedBy(?p, ?m) ^
   manufacturedBy(?pv, ?m) ^ IdProduct(?p, ?idP1) ^ IdProduct(?pv, ?idP2) ^
    Brand(?p, ?b1) ^ Brand(?pv, ?b2) ^ swrlb:stringEqualIgnoreCase(?b1, ?b2
   ) ^ Model(?p, ?m1) ^ Model(?pv, ?m2) ^ swrlb:equal(?idP1, ?idP2) ^ swrlb
   :stringEqualIgnoreCase(?m1, ?m2) -> Valid(?pv)
```

Listing 3.2: SWRL rule for validating products

The rule checks several properties of the products, including the manufacturer (**manufacturedBy**), the ID of the product (**IdProduct**), the brand (Brand), and the model (**Model**). It verifies that the **manufacturedBy** relationship is established for both the product and its corresponding product to validate. It also ensures that the ID and brand of both products are equal, as well as the model.
Similar to the previous rule, this rule utilizes built-in SWRL functions such as **swrlb:equal** for value comparison and **swrlb:stringEqualIgnoreCase** for case-insensitive string matching.
If all the conditions specified in the rule are met, the conclusion is reached that the product to validate (**?pv**) is valid, and the **Valid** relation is inferred.

These two rules plays the crucial role in our ontology layer for ensuring the validity and authenticity of inventories.

3.3.2.4 Verifying an inventory

When a client wants to verify the authenticity of an inventory (product or certificate), they can use the mobile application to scan the QR code on the inventory. The application will then extract the information encoded in the QR code and send it to the backend and then it will contact the blockchain and search for the corresponding inventory information on the blockchain. If the inventory information exists on the blockchain and is validated as authentic based on the SWRL rules and ontology model, the application will display a message indicating that the product is genuine. If the product information is not found on the blockchain or fails the validation process, the application will display a message indicating that the product may be counterfeit. Figure 3.5 sums it all up.

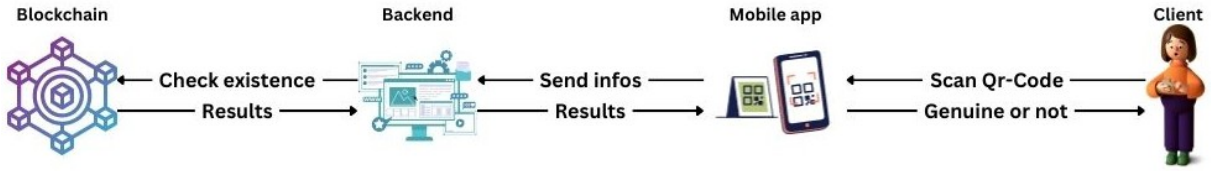


Figure 3.5: Verifying inventory process.

3.4 Conclusion

In conclusion, the design of our anti-counterfeit blockchain-based system with ontology has been carefully thought out to address the challenges of counterfeiting the inventories (products and certificates). By utilizing blockchain technology and ontology, the system ensures the transparency and authenticity of inventories, while SWRL rules provide additional validation of the product's authenticity. while the blockchain network ensures the security and immutability of the data stored. The system architecture and various components work together seamlessly to create a robust anti-counterfeit system that benefits both businesses and consumers alike. In the next chapter we will talk more clearly about the implementation of the system.

CHAPTER 4

IMPLEMENTATIONS AND RESULTS

4.1 Introduction

In the previous chapter, we covered the architecture and various components of the AuthentiQatoR system. This chapter will focus on the implementation of the system itself. Firstly, we will present the tools and programming languages used to construct our system. Following that, we will delve into the key components that make up our system.

4.2 Development Tools

Our system was developed on Laptop Lenovo z41-70 80k5 Processor Intel® Core™ i7-5500U CPU @ 2.40GHz, 4 Core(s), 8 Logical Processor(s), RAM 16 Go of memory and SSD 120 GB. Operating system Kali GNU/Linux 2022.2 x64.

4.2.1 Web app development tools

The tools we used to develop the web app are listed below.

4.2.1.1 Visual Studio Code

Visual Studio Code is a lightweight but powerful source code editor which runs on your desktop and is available for Windows, macOS and Linux. It comes with built-in support for JavaScript, TypeScript and Node.js and has a rich ecosystem of extensions for other languages

and runtimes (such as C++, C#, Java, Python, PHP, Go, .NET)[40]. We chose VSCode as a IDE because it's free and open-source.

Figure 4.1: Visual studio code logo[40].

4.2.1.2 Laravel

Laravel is a web application framework with expressive, elegant syntax. A web framework provides a structure and starting point for creating your application, allowing you to focus on creating something amazing while we sweat the details.

Laravel strives to provide an amazing developer experience while providing powerful features such as thorough dependency injection, an expressive database abstraction layer, queues and scheduled jobs, unit and integration testing, and more[41].

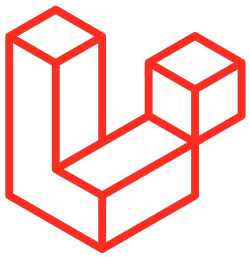


Figure 4.2: Laravel's logo[41].

4.2.1.3 Ganache

Ganache is a personal blockchain for rapid Ethereum and Corda distributed application development. You can use Ganache across the entire development cycle; enabling you to develop, deploy, and test your dApps in a safe and deterministic environment.

Ganache comes in two flavors: a UI and CLI. Ganache UI is a desktop application supporting Ethereum and Filecoin technology. It offers:

- console.log in Solidity
- Zero-config Mainnet and testnet forking
- Fork any Ethereum network without waiting to sync

- Ethereum JSON-RPC support
- Snapshot/revert state
- Mine blocks instantly, on demand, or at an interval
- Fast-forward time
- Impersonate any account (no private keys required!)
- Listens for JSON-RPC 2.0 requests over HTTP/WebSockets
- Programmatic use in Node.js
- Pending Transactions [42]

Figure 4.3: Ganache's logo[42].

4.2.1.4 Brownie

Brownie is a Python-based development and testing framework for smart contracts targeting the Ethereum Virtual Machine. It offers:

- Full support for Solidity and Vyper
- Contract testing via pytest, including trace-based coverage evaluation
- Property-based and stateful testing via hypothesis
- Powerful debugging tools, including python-style tracebacks and custom error strings
- Built-in console for quick project interaction
- Support for ethPM packages [43]

Figure 4.4: Brownie's logo[43].

4.2.1.5 Protégé

Protégé is a free, open-source platform that provides a growing user community with a suite of tools to construct domain models and knowledge-based applications with ontologies[47].

Figure 4.5: Protégé's logo[47].

4.2.2 Android app development tools

The tools we used to develop the android app are listed below.

4.2.2.1 Android studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development. Based on the powerful code editor and developer tools from IntelliJ IDEA , Android Studio offers even more features that enhance your productivity when building Android apps, such as:

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code

- Extensive testing tools and frameworks
- Lint tools to catch performance, usability, version compatibility, and other problems
- C++ and NDK support
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine [44]

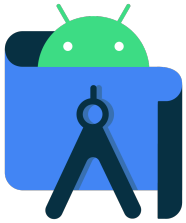


Figure 4.6: Android studio logo[44].

4.2.2.2 Flutter

Flutter is an open source framework by Google for building beautiful, natively compiled, multi-platform applications from a single codebase.[45].

Figure 4.7: Flutter's logo[45].

4.2.2.3 MetaMask

MetaMask is an extension for accessing Ethereum enabled distributed applications, or "Dapps" in your browser. The extension injects the Ethereum web3 API into every website's javascript context, so that dapps can read from the blockchain. MetaMask also lets the user create and manage their own identities (via private keys, local client wallet and hardware wallets like Trezor™), so when a Dapp wants to perform a transaction and write to the blockchain, the user gets a secure interface to review the transaction, before approving or rejecting it[46].

Figure 4.8: MataMask's logo[46].

4.3 Implementation and Realization of The System

This section of the chapter provides a detailed description of our AuthentiQatoR system, supported by its various interfaces.

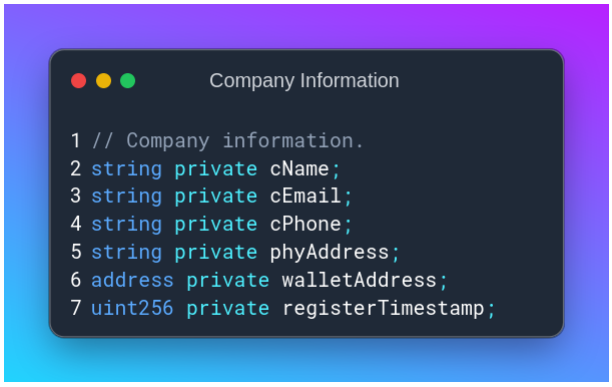
4.3.1 System overview

The anti-counterfeit blockchain-based system we built is a complete solution that handles the issues of counterfeit products and certificates. It is made up of multiple linked components that function in unison.

The blockchain network is at the center of the system, acting as a decentralized and tamper-proof ledger for storing inventories information. This blockchain network provides data immutability and transparency, making it very secure and trustworthy.

Every company or concerned party that wants to protect its inventories must register in the system to have a smart contract in blockchain network. This smart contract contains three main parties as shown in Figure 4.9.

(a) (b)



(c)

Figure 4.9: Company contract

The Figure 4.9 consists of three sub-figures that provide a detailed description of the company smart contract code used in our system.

In the first sub-figure, Figure 4.13a, the Certification struct is defined. It includes fields such as the title of the certificate, the first name and last name of the certificate holder, a description of the certificate, and the timestamp when the certificate was issued.

In the second sub-figure, Figure 4.13b, the Product struct is defined. It contains fields related to a specific product, including the brand, model, description, manufacturer name, and the timestamp when the product was manufactured.

The third sub-figure, Figure 4.9c, provides information about the company. It includes fields such as the company name, email, phone number, physical address, wallet address, and the timestamp when the company registered within the system.

When a company gets a verified account it means there's a copy of this smart contract deployed in the blockchain network under the name of this company. Now it can add its products or certificates through the steps that we discussed earlier.

As we saw before, the first step to add an inventory is verifying the MetaMask wallet as the first layer of validation. the system ensures that the MetaMask wallet associated with the com-

pany account is verified before going on to the next step of adding inventory to the blockchain. The following code snippets demonstrates the process.

Get message to verify

```php
// Generate message to verify.
public function getVerifyMessage()
{
    $nonce = \Str::random();
    $message = "Sign this message to confirm you own this wallet address. This action will not cost any gas fees.\n\nNonce: " . $nonce;

    session()->put('sign_message', $message);

    return $message;
}
```

Figure 4.10: Generate message to verify.

The first method **Figure 4.10** generate a message contains random string and then send it to the user to sign it via MetaMask wallet in other side it store the message in the session.

The second method **Figure 4.11** receives the signed message from the user and the address after that it sign the same message that has been stored in the session using the address through the same hashing method and then compare it with signed one from the user, if they're equal it means the wallet is verified otherwise it's not.

Verify the signed message

```php
// Verify the message.
public function verifyMessage(Request $request): String
{
    $result = $this->verifySignature(session()->pull('sign_message'), $request->input('signature'), $request->input('address'));

    $customerId = Auth::guard('customer')->user()->id;
    if ($result) {
        $r = $this->authWallet($request->input('address'), $customerId);
        return $r;
        return redirect()->back()->with('success', 'Your wallet has been successfully verified.');
    }
    return redirect()->back()->with('error', 'Your wallet has not been verified!');
}
protected function verifySignature(string $message, string $signature, string $address): bool
{
    $hash = Keccak::hash(sprintf("\x19Ethereum Signed Message:\n%s%s", strlen($message), $message), 256);
    $sign = [
        'r' => substr($signature, 2, 64),
        's' => substr($signature, 66, 64),
    ];
    $recid = ord(hex2bin(substr($signature, 130, 2))) - 27;

    if ($recid != ($recid & 1)) {
        return false;
    }


    $pubkey = (new EC('secp256k1'))->recoverPubKey($hash, $sign, $recid);
    $derived_address = '0x' . substr(Keccak::hash(substr(hex2bin($pubkey->encode('hex')), 1), 256), 24);

    return (\Str::lower($address) === $derived_address);
}
```

Figure 4.11: Verify signed message.

In the second layer of validation, the user will be asked to enter the inventory's details in custom web page to validate it through the second layer of ontology. The code below Figure 4.12

shows how the system passes inventory's details to the ontology layer to be classified through pellet reasoner and await for results, if it's validated then the system move to the next step of process, otherwise the process will be fail and the inventory will be denied.

Send data to ontology layer

```php
public function validateProduct($data)
    {

        $process = new Process(
          [
            'python3',
            storage_path() . '/onto/isValidProduct.py',
            $data['brand'],
            $data['model'],
            $data['idProduct'],
            $data['companyName']
          ]);
        $process->run();

        if (!$process->isSuccessful()) {
            throw new ProcessFailedException($process);
        }

        $result = $process->getOutput();
        return $result;
    }
```

(a)

Send data to ontology layer

```php
public function validateCertificate($data)
    {

        $process = new Process(
          [
            'python3',
            storage_path() . '/onto/isValidCert.py',
            $data['title'],
            $data['hFirstName'],
            $data['hLastName'],
            $data['certId'],
            $data['date']
          ]);
        $process->run();

        if (!$process->isSuccessful()) {
            throw new ProcessFailedException($process);
        }

        $result = $process->getOutput();
        return $result;
    }
```

(b)

Figure 4.12: Send data to ontology layer

Figure 4.13 is a code snippets of python that shows how pellet reasoner classify inventories and return whether it's valid or invalid.

Classifying Product

```python
def getCompany(onto,cname):
    for i in onto.Company.instances():
        if cname in i.CompanyName:
            return i
        else:
            continue

    return ''

def verifyProduct(brand, model, idProduct, companyName):
    onto = get_ontology("<path>/trustOstology.owl").load()

    company=getCompany(onto,cname=companyName)

    if company != '':
        cert = onto.ProductToValidate(
                        str(idProduct)+'product',
                        namespace = onto,
                        manufacturedBy = [company],
                        IdProduct=[idProduct],
                        Brand=[brand],
                        Model=[model]
                    )
        onto.save(file="<path>/trustOstology.owl")
        sync_reasoner_pellet(infer_property_values = True, infer_data_property_values = True)
        list=onto.search(type=onto.Valid)
        if cert in list:
            return 'Valid'
        else:
            return 'inValid'
    else: return 'inValid'
```

(a)

Classifying Certificate

```python
def getStudent(onto,holder_last_name,holder_first_name):
    for i in onto.CertificateHolder.instances():
        if holder_last_name in i.LastName and holder_first_name in i.FirstName:
            return i
        else:
            continue

    return ''

def verifyCertificate1(title,last_name,first_name,number,timestamp):
    onto = get_ontology("<path>/trustOstology.owl").load()

    student1=getStudent(onto,holder_last_name=last_name, holder_first_name=first_name)

    if student1 != '':
        cert = onto.CertificateToValidate(
                        str(number)+'cert',
                        namespace = onto,
                        holdBy = [student1],
                        IdCertificate=[number],
                        Certificate_title=[title],
                        issue_timestamp=[timestamp]
                    )
        onto.save(file="<path>/trustOstology.owl")
        sync_reasoner_pellet(infer_property_values = True, infer_data_property_values = True)
        list=onto.search(type=onto.Valid)
        if cert in list:
            return 'Valid'
        else:
            return 'inValid'
    else: return 'inValid'
```

(b)

Figure 4.13: Classifying inventories with pellet

Once the inventory verified through the two layers it will be ready to be added to the blockchain,

the system takes all required data of inventory that has been entered and passes it to data controller to make sure it doesn't contains any suspicious inputs, then all data is combined in one string and that string is hashed to get a unique id of the inventory in the blockchain, after that the system passes all inventory's data and the last hash (uid) to the blockchain to be stored. Once it stored successfully, the system generate a QR code contains the hash (uid), company smart contract address and the type of inventory (product or certificate), this QR code will be returned to company account manager to be added to inventory. **Figure 4.14** demonstrates the process.

Storing inventory

```php
public function storeInventory(Request $request)
{
    $company = CompanyContract::find($request->company_id);
    if (!$company)
        return redirect()->back()->with('error', 'Whoops! something went wrong.');

    // Validate inputs.
    $validator = Validator::make($request->all(), [
        'type'       => ['required', 'in:1,2'],
        'brand'      => ['required_if:type,1'],
        'model'      => ['required_if:type,1'],
        'descriptionP'  => ['required_if:type,1'],
        'title'      => ['required_if:type,2'],
        'holderFName'   => ['required_if:type,2'],
        'holderLName'   => ['required_if:type,2'],
        'descriptionC'  => ['required_if:type,2'],
        'issueDate'     => ['required_if:type,2'],
    ]);
    if ($validator->fails())
        return redirect()->back()->withErrors($validator->errors());


    $web3 = new Web3Helper();
    $companyName = $company->company_name;
    // If the inventory is product.
    if ($request->type == '1') {
        $timestamp = Carbon::now()->timestamp;
        $brand = $request->brand;
        $model = $request->model;
        $description = $request->descriptionP;

        $uid = Hash::make($brand . $model . $description . $companyName . $timestamp);
        $callData = [$uid, $brand, $model, $description, $companyName, $timestamp];
        $extraParams = ['from' => $company->wallet_address];
        $res = $web3->sendCompanyContract($company->contract_address, 'addProduct', $callData);
        if ($res['result'] == 'success') {
            // Generate Qr-Code.
            $qrContent = array(
                'uid'   => $uid,
                'type'  => 1,
                'ccaddress' => $company->contract_address
            );
            $fileName = 'companies/Qr-Codes/' . $companyName . '/' . time() . '_product' . '.png';
            $qrCode = QrCode::format('png')
                ->size(300)
                ->errorCorrection('H')
                ->merge(storage_path('app/public/images/ic-product.png'), 0.5, true)
                ->generate(json_encode($qrContent), storage_path('app/public/' . $fileName));
            $inv = Inventory::create([
                'company_contract_id'   => $company->id,
                'company_name'          => $companyName,
                'inventory'             => 'storage/' . $fileName,
                'type'                  => 1,
            ]);
            return redirect()->back()->with('success', 'Your product added successfully!' . ' ' . $uid);
        } else
            return redirect()->back()->with('error', 'Whoops! something went wrong');
    }
    // If the inventory is certificate.
    elseif ($request->type == '2') {
        // Do same concept for storing products.
}
```

Figure 4.14: Storing inventory.

For now, company role has been done and let's consider it has some products registered in blockchain and the client want to scan the authenticity of a product, he should scan the QR code in the products using our android application. **Figure 4.15** is a code snippet of the android

application that take care of this process.

Scanning Qr-Code

```
void _onQRViewCreated(QRViewController controller) {
    this.controller = controller;
    this.controller?.resumeCamera();

    controller.scannedDataStream.listen((scanData) {
      this.controller?.pauseCamera();
      apiController.postWithSecretKeys(
          url: APIs.scanInventory, // Endpoint.
          data: {"scanned_qr": scanData.code?.trim()}) // Data extracted from qr-code.
          .then((value) {
        ScannedQrModel scannedQrModel = ScannedQrModel.fromJson(value.data);
        if(scannedQrModel.status == 'success'){ // The scanned inventory found in the blockchain
          Map<String, dynamic> jsonProduct = jsonDecode(scannedQrModel.data??'');
          if(scannedQrModel.type == 1) // Product
          {
            // Show alert dialog indicating it's genuine.
            ProductModel product = ProductModel.fromJson(jsonProduct);
            AwesomeDialog(
                context: context,
                animType: AnimType.scale,
                customHeader: Image.asset('assets/images/ic_product.png'),
                body: Column(...)
            ).show();
          }else if(scannedQrModel.type == 2) // Certificate
          { // Do same concept for product.

          }
        }else{ // Show alert dialog indicating it may be counterfeited.
          AwesomeDialog(
            context: context,
            dialogType: DialogType.warning,
            animType: AnimType.scale,
            title: 'Error',
            desc: 'Whoops! The scanned inventory is not registered in our system',
          ).show();
        }
      });
    });
  }
```

Figure 4.15: Scanning Qr Code.

As we can see there's three main parties in this method. Firstly, it extract the data from the QR code, then it send the extracted data to the endpoint of the system, the backend take care of calling blockchain and getting results, once done the backend returns that results to our application. Finally, the application shows alerts dialog indicating whether it's genuine or not depending on the returns results.

4.3.2 System interfaces

In this section, we will provide an overview of the interfaces of our system. You will get a glimpse of the main web and android application pages through screenshots, starting with the home page. These interfaces play a crucial role in enabling users to interact with the system,

manage their accounts, add products, and perform product verification. By exploring these interfaces, you will gain a better understanding of the user experience and the functionality offered by our system. Let's dive into the interface designs and explore the key features of each page. Figure 4.16 shows the logo of our system.

Figure 4.16: AuthentiQatoR's logo.

4.3.2.1 Home page

The home page (Figure 4.17) serves as the entry point to our anti-counterfeit blockchain-based system, providing users with an overview of its purpose and functionality. Upon landing on the home page, users are greeted with a concise description of the system, highlighting its ability to combat counterfeit products and ensure supply chain transparency. The page features prominently displayed buttons for registration and login, offering users the option to either create a new account or access their existing accounts. These buttons serve as gateways to the respective user interfaces, empowering individuals and businesses to engage with the system and leverage its anti-counterfeiting capabilities.

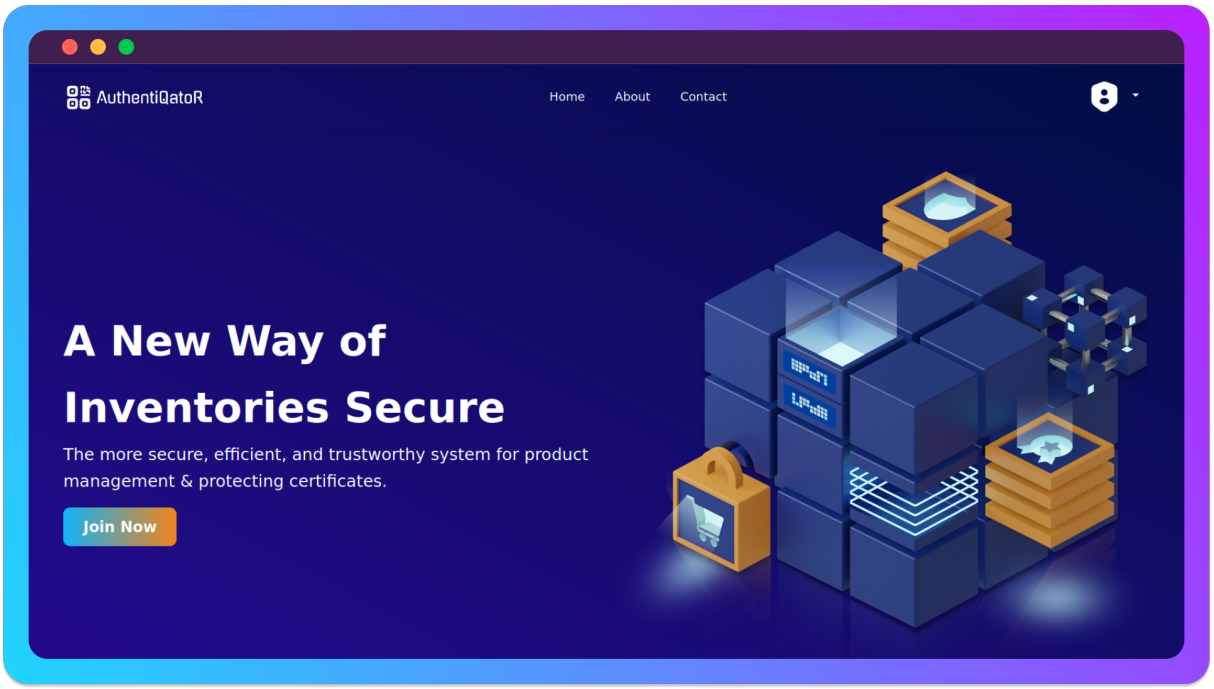


Figure 4.17: Home page.

4.3.2.2 Register/Login pages

The register page (Figure 4.18) is where new users can easily join our anti-counterfeit blockchain-based system. It presents a user-friendly interface with fields to input necessary information for account creation. You'll find fields for name, email, username, phone number, and password, enabling users to securely provide their personal details. By filling out these fields, users can register an account, granting them access to the system's features and empowering them to contribute to combating counterfeit products.

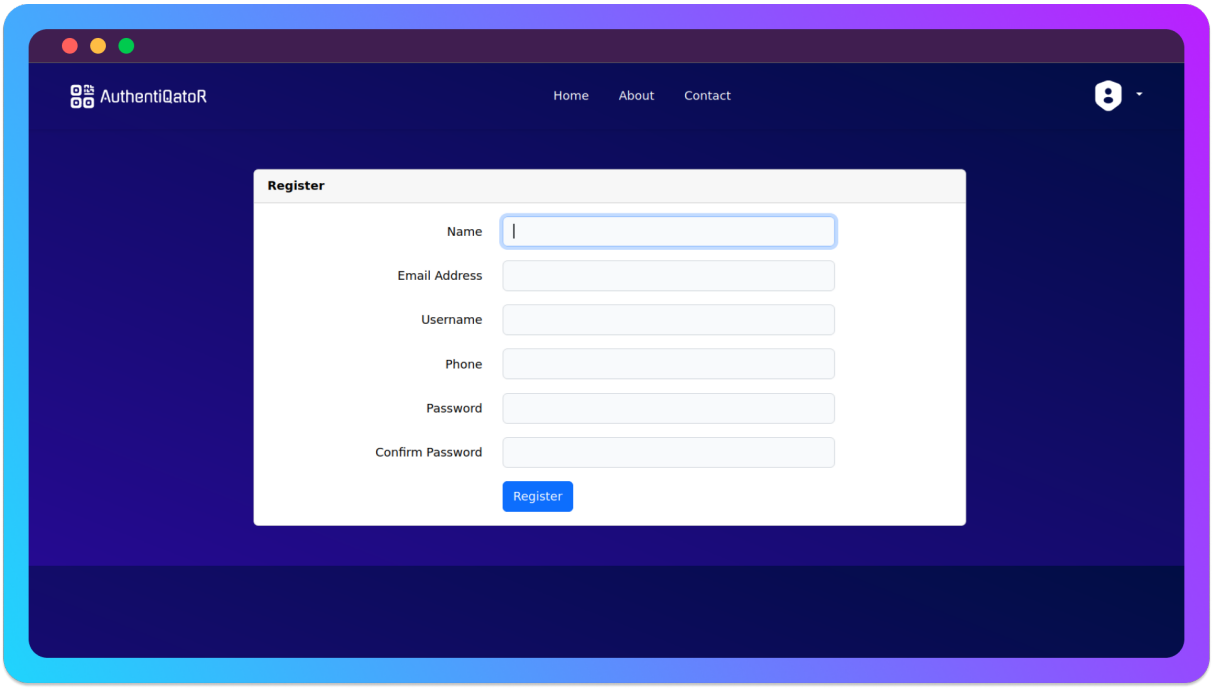


Figure 4.18: Register page.

On the login page (Figure 4.19), existing users can effortlessly authenticate themselves and access their accounts. This streamlined interface prompts users to enter either their email or username, along with the corresponding password. By accurately providing this information, users can securely log in to the system and access their personalized dashboards and functionalities. The login page ensures that only authorized individuals can access the system, ensuring account security and maintaining a protected environment for anti-counterfeiting efforts.

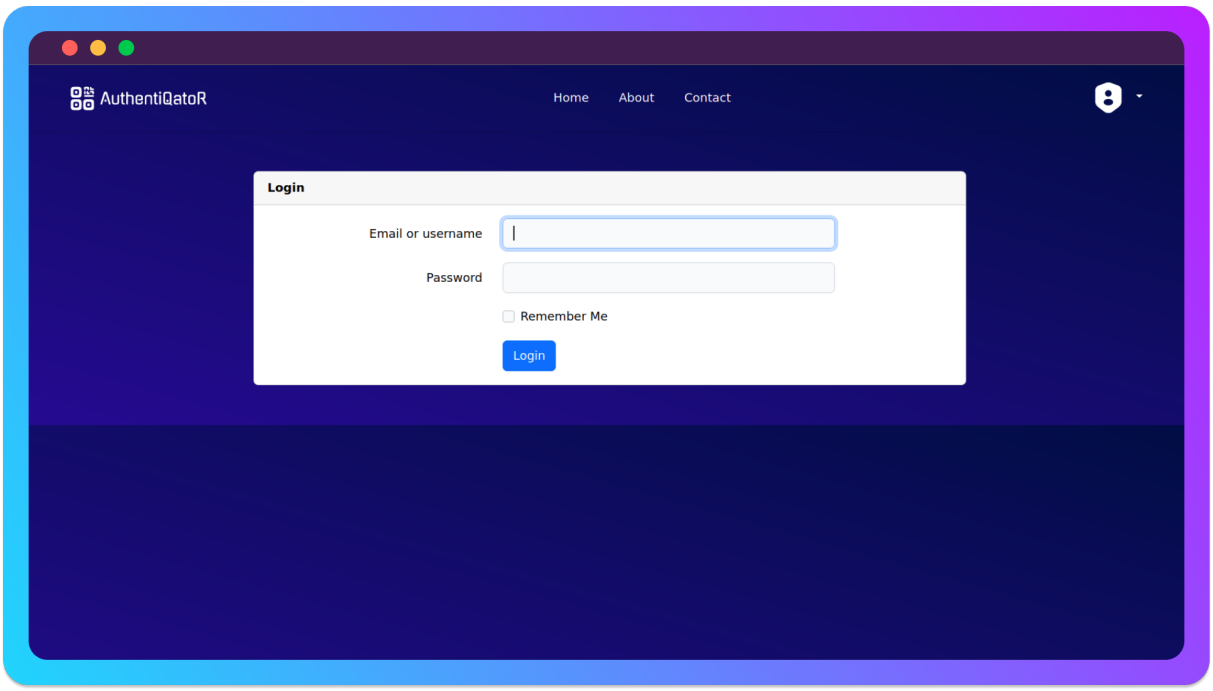


Figure 4.19: Login page.

4.3.2.3 Adding inventory

The verifying wallet interface (Figure 4.20) plays a crucial role in ensuring the utmost security and authenticity of user accounts within our system. When you click on the "Verify Wallet Address" button, our website will prompt you to seamlessly connect your Metamask wallet to our platform. In case you don't have Metamask installed, we'll guide you through the process of downloading and installing it beforehand. Once successfully connected, you'll encounter a clear and concise message requesting you to sign a specific message using your Metamask wallet. This signature verification process serves as a robust mechanism to confirm beyond doubt that you are the legitimate owner of the wallet address you're utilizing.

To proceed with the verification, all you need to do is simply approve the message within your Metamask wallet. This ensures a secure and trust-based validation of your ownership of the wallet address. We want to emphasize that our platform respects and prioritizes your privacy at all times. We never have access to your private keys or any other sensitive information stored within your Metamask wallet. The verification process is designed solely to confirm the authenticity and ownership of the wallet address you're using on our platform.

Furthermore, we provide you the option and flexibility to choose the precise account to which you want to add your inventory within the wallet interface. This enables you to manage and arrange your inventory-related tasks in accordance with your preferences and organizational needs. We recognize the significance of providing a unified and user-centric experience, and we

are committed to making your contact with the wallet interface simple, safe, and trouble-free.

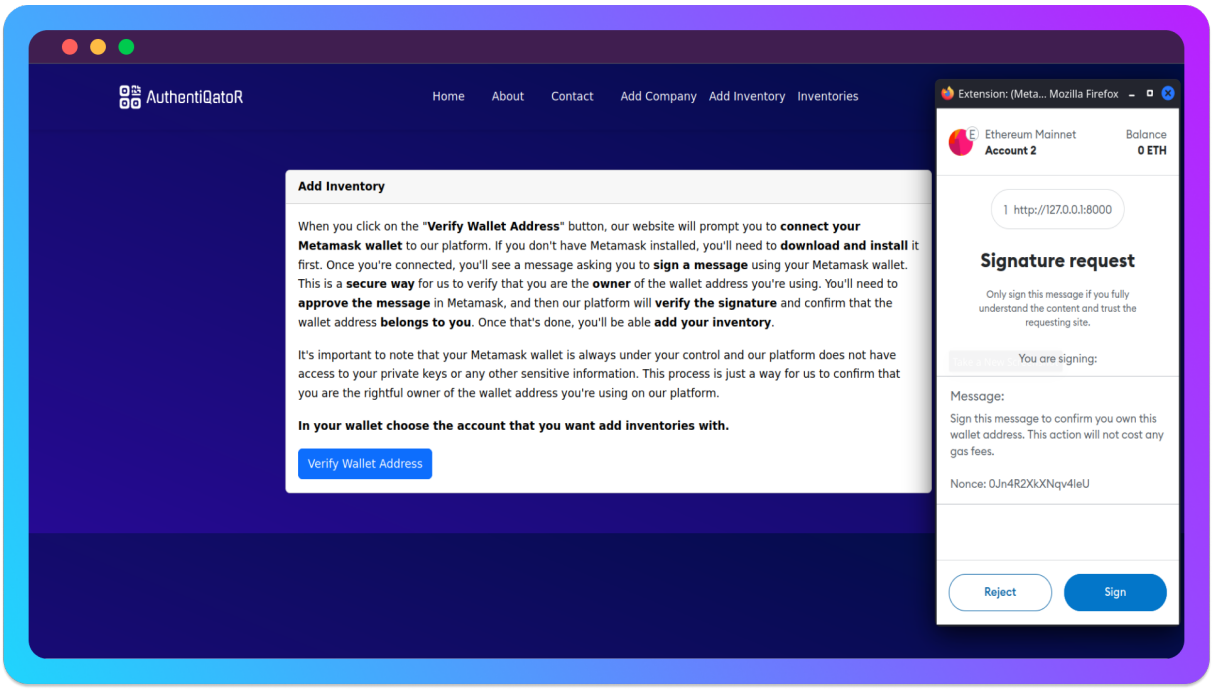


Figure 4.20: Verifying wallet page.

Once your wallet is verified, you'll be directed to an intuitive interface where you can enter the details of your inventory, whether it's a product or certificate. This user-friendly interface features a structured form that allows you to input essential information such as brand, model and description for products, or title, holder's first name, holder's last name, description and issue date for certificates. By accurately filling out these fields, you contribute to the integrity and traceability of your inventory within our system. The interface includes helpful features like input validations and clear instructions to ensure a seamless experience. Your input plays a vital role in maintaining the transparency and trustworthiness of the products or certificates associated with your account in our system. Figure 4.21 shows the concerned interface for adding an inventory.

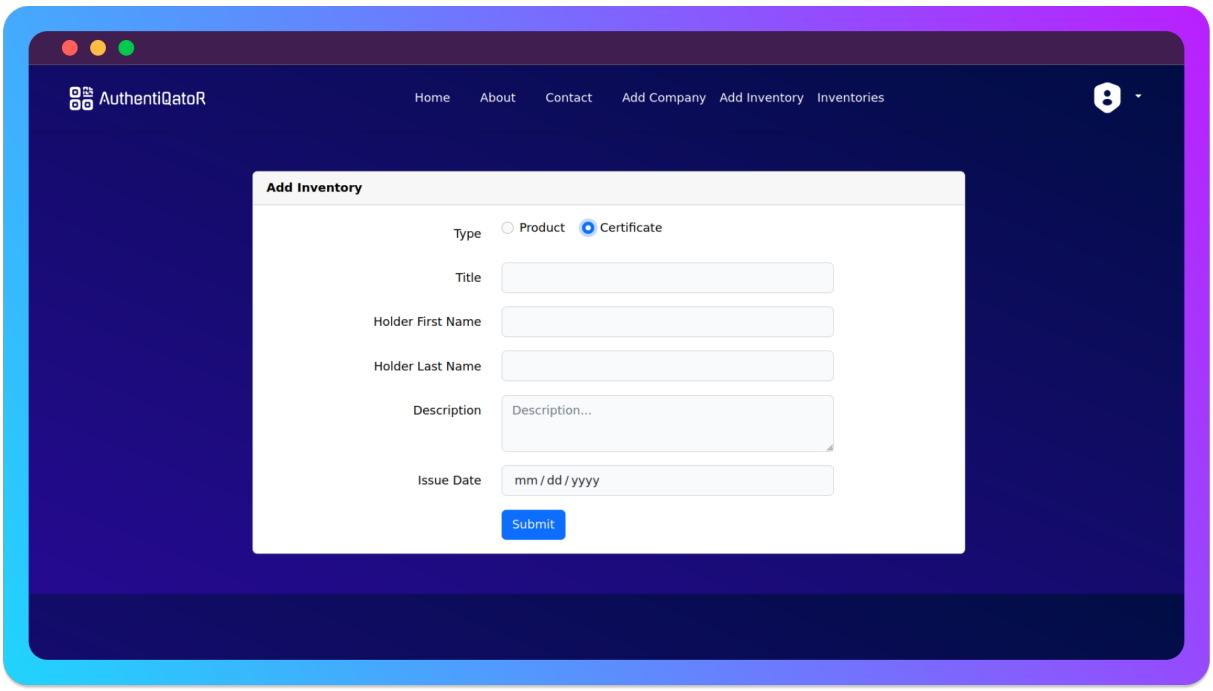


Figure 4.21: Adding inventory page.

Once you have successfully added your inventory, you will be presented with a comprehensive interface displaying all the inventories you have added. This interface provides a convenient overview of your products or certificates, accompanied by their corresponding QR codes. Each inventory is presented with key details such as company name, type, Qr code and buttons to view or download the QR code, allowing for easy identification and management. The QR codes associated with each inventory serve as a powerful tool for product verification. Users can simply scan the QR code using the mobile application, which instantly retrieves information from the blockchain and confirms the authenticity of the product or certificate. This interface offers a centralized hub where you can effortlessly access and monitor your added inventories, ensuring a streamlined and efficient management process. Figure 4.22 shows the described interface.

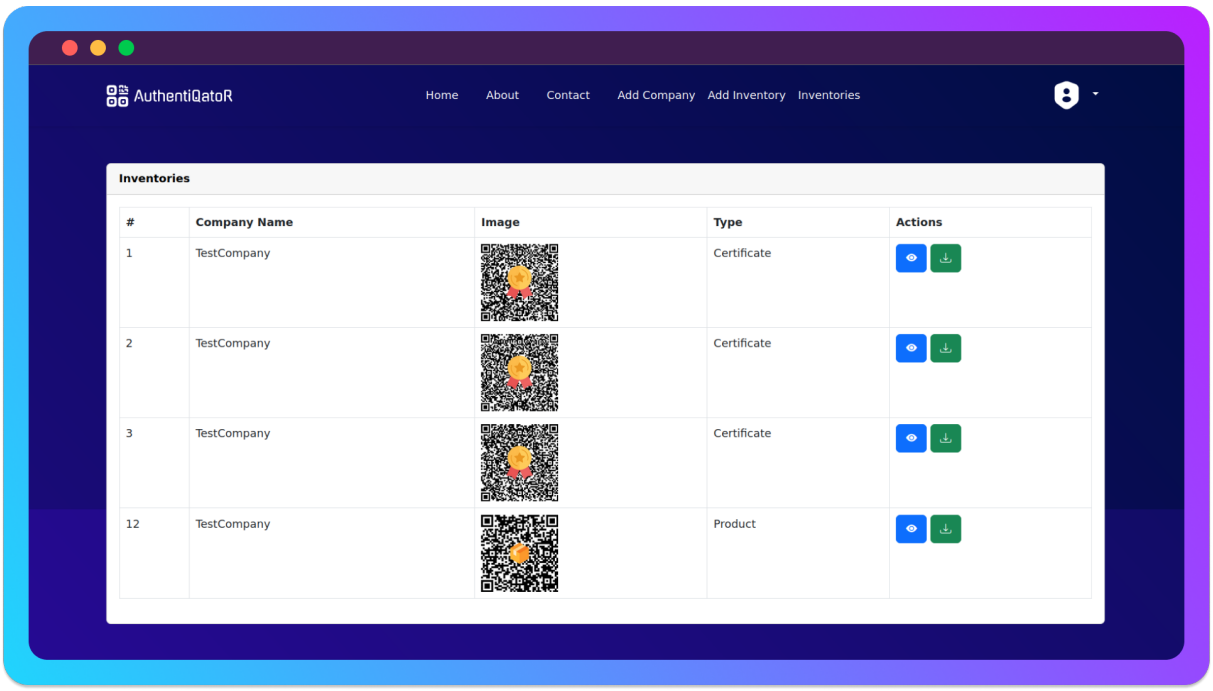


Figure 4.22: List of inventories page.

Within our system, we have incorporated a feature that empowers companies to add sub-companies under their main account. This functionality is designed to facilitate streamlined management and organization of multiple entities within a single platform. The interface for adding a sub-company provides a straightforward and intuitive experience. Companies can input essential details such as the sub-company's name, email, phone number, address, and property contract. By gathering these pertinent details, companies can easily establish and maintain a network of associated sub-companies. This feature enhances collaboration, coordination, and oversight, enabling seamless management of a diverse range of entities within the system. Figure 4.23 shows the described interface.

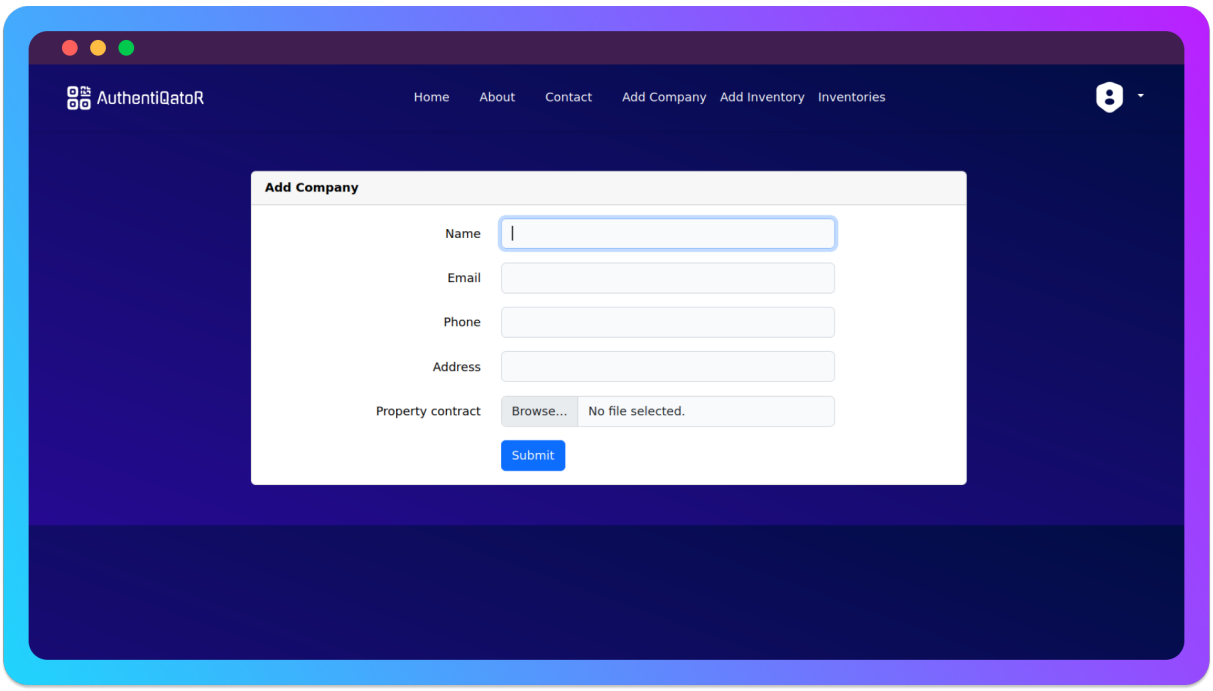
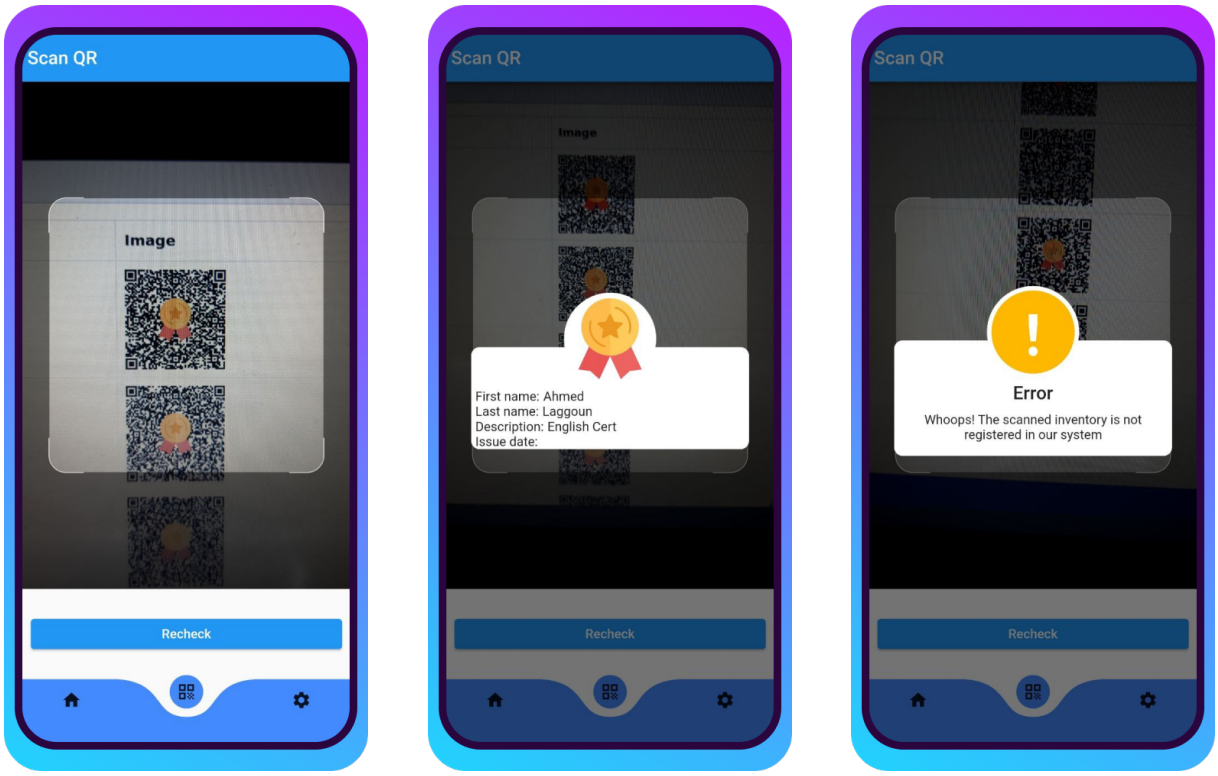


Figure 4.23: Add company page.

The application UI offers customers a seamless and straightforward experience for scanning QR codes and checking product authenticity. Users are met with a clean and user-friendly interface when they launch the program. The QR code scanner (Figure 4.24a), which is prominently displayed on the screen, is the interface's key feature. Users may start the scanning process by simply aiming their device's camera at the QR code printed on a product. The program instantly takes the QR code and analyses the data it contains. The program pulls the needed data from the blockchain and runs a verification check in a couple of seconds. The user is subsequently shown a clear and succinct notice indicating whether the product is genuine (Figure 4.24b) or potentially counterfeit (Figure 4.24c). The interface is designed to prioritize speed and accuracy, allowing users to make informed purchasing decisions and mitigate the risks associated with counterfeit products.



(a) (b) (c)

Figure 4.24: Android application interface

4.4 Conclusion

Finally, this chapter delves into the design and implementation of our blockchain-based anti-counterfeit system. We investigated the numerous development tools and technologies used throughout the system's development, including the programming languages, frameworks, and libraries that were critical to its success. Code samples were shown to offer insight into the underlying logic and operation of essential components such smart contract code and the verification process. In addition, we investigated the system's interfaces and displays, providing a visual picture of the user experience. Our system provides a strong and secure solution for preventing counterfeit items and providing a transparent supply chain by integrating blockchain technology, ontology, and a user-friendly interface.

GENERAL CONCLUSION

The combination of an anti-counterfeit blockchain system with ontology holds great promise for addressing the challenges faced by various industries, particularly in ensuring the authenticity and integrity of inventories. By leveraging blockchain technology, automated data collection and verification processes can be established, ensuring the traceability and immutability of inventory information. This integrated system provides a secure and tamper-resistant platform for verifying the authenticity of inventories, reducing the risk of counterfeit goods entering the market.

Additionally, by incorporating ontology into the blockchain system, a structured and semantic representation of inventory data can be achieved. This enables a deeper understanding of inventory attributes, relationships, and ownership, enhancing the overall anti-counterfeit capabilities. The combination of ontology and blockchain technology offers a decentralized and transparent solution, allowing stakeholders to validate the authenticity of inventories across the entire inventory lifecycle.

The research and development efforts in the field of anti-counterfeit blockchain systems combined with ontology have demonstrated their potential in addressing the pressing issue of counterfeit inventories. This innovative approach provides a reliable and efficient means to combat counterfeiting, ensuring consumer trust and confidence in the authenticity of inventories. Furthermore, it offers opportunities for collaboration among industry players, regulators, and consumers to create a robust ecosystem that safeguards the integrity of goods and protects the interests of all stakeholders involved.

In this thesis, we described the design of our system, which handles the difficulties of safe sharing and inventory authenticity verification. This was an important issue since a lack of confidence and security provided substantial barriers to eliminating counterfeit items. Because

there is currently no reliable source of inventory information, anybody may falsify data, making it impossible for customers to verify the legitimacy of items. As a result, our solution focuses on decentralized storage, safe sharing, and powerful inventory data application, offering greater security, confidentiality, and trust for everyone involved in the fight against counterfeits.

As a result, we've proposed our AuthentiQatoR system, which is built on the Blockchain and Ontology, for securing inventories data by storing, and sharing it in transparent and immutable way. It also allows customers to manage their inventories, for example, by creating sub-companies under the name of main company and adding inventories depending on it's category. Also, we can even cite one of the strong aspects of the AuthentiQatoR system:

For a clients, this means that they can securely purchase inventories with no fear of its origin.

For a customers, this means that they can store their inventories in transparent and immutable place, which makes it such a trustworthy source for clients. hence, maintain the confidence of clients and the reputation of the company.

Future work

The current work primarily attempts to manage inventory data and track its origin. We attempted to provide a possible solution based on blockchain technology as a distributed network and an ontology. Several components of this work, as described below, might be added or enhanced for future development:

- Ownership tracking, means that the chain of ownership of any inventory can be known. This means we can add additional value to inventories. For example, if a famous person buys this product.
- We plan to eliminate the need for the mobile app to read the QR code, and let any QR code reader do the trick.
- User reviews and ratings: implement a user review and rating system where customers can provide feedback and rate their inventory experiences. This feature helps potential buyers make more informed decisions based on the experiences of others.
- Combine this project with AI.

BIBLIOGRAPHY

[1] *Counterfeiting is on the rise, and projected to exceed $3 trillion in 2022*. https://scm.ncsu.edu/scm-articles/article/counterfeiting-is-on-the-rise-projected-to-exceed-3-trillion-in-2022, Accessed April 18, 2023.

[2] *Clothing, accessories and footwear*. https://euipo.europa.eu/ohimportal/en/web/observatory/ip-infringements_clothing-accessories-footwear, Accessed April 24, 2023.

[3] *Fuse Chicken Vs. Amazon Is The David Vs. Goliath Lawsuit To Watch In 2018*. https://www.forbes.com/sites/wadeshepard/2018/01/14/fuse-chicken-vs-amazon-is-the-david-vs-goliath-lawsuit-to-watch-in-2018/?sh=4f26a3da5115, Accessed April 24, 2023.

[4] MarketsandMarkets, (2021)*Anti-counterfeit Packaging Market (Mass Encoding, RFID, Tamper Evidence, Hologram, Forensic Markers), End-use Industry (Food & Beverage, Pharmaceuticals, Electrical & Electronics. Automotive, Luxury goods), and region - Global Forcast to 2026* (PK 4055). https://www.marketsandmarkets.com/Market-Reports/anti-counterfeit-packaging-advanced-technologies-and-global-market-129.html, Accessed April 26, 2023.

[5] Pio, Luigi & Cavaliere, Luigi Pio Leonardo & Krishnabhaskarmangalasserri, & Venkateswaran, P & Byloppilly, Rameshwaran & Effendy, Femmy & More, Amrita & Rajest, Suman & Rajan, Regin. (2021). *THE IMPACT OF BRAND COUNTERFEITING*

ON CONSUMER BEHAVIOR IN THE FASHION SECTOR. Türk Fizyoterapi ve Rehabilitasyon Dergisi/Turkish Journal of Physiotherapy and Rehabilitation. 32. 19831-19847.

[6] Fink, C., Maskus, K. E., & Qian, J. (2016). *The economic effects of counterfeiting and piracy: A review and implications for developing countries* (Policy Research Working Paper No. 7586). World Bank Group.

[7] *Knock-off the Knockoffs: The Fight Against Trademark and Copyright Infringement*, Illinois Business Law Journal, September 21, 2009.

[8] European Union Intellectual Property Office, *Anti-counterfeiting technology guide – *, European Union Intellectual Property Office, 2021.

[9] *Environmental Impact of Counterfeiting and How can it be resolved?*, `https://blog.countercheck.com/environmental-impact-of-counterfeiting-and-how-can-it-be-resolved-a5b503c8b01` Accessed June 12, 2023.

[10] Tiana Laurence. *"Introduction to blockchain technology"*. Van Haren, 2019.

[11] Adam Hayes. *"Blockchain Facts: What is it, How it works, and how it can be used"*.`https://www.investopedia.com/terms/b/blockchain.asp`, Accessed: March 11, 2023.

[12] J Michael, ALAN Cohn, and Jared R Butcher. *"Blockchain technology"*. In: The Journal 1.7 (2018).

[13] *"Blockchain: A technical primer"*. `https://www2.deloitte.com/content/dam/insights/us/articles/4436_Blockchain-primer/DI_Blockchain_Primer.pdf`. Accessed:Mar 11, 2023.

[14] A. Shahnaz, U. Qamar, and A. Khalid. *"Using blockchain for elec- tronic health records"*. In: IEEE Access 7 (2019), pp. 147782–147795.

[15] Haber, S., & Stornetta, W. S. (1991). *"How to time-stamp a digital document"*. Journal of Cryptology, 3(2), 99-111.

[16] *"History of blockchain"*. `https://www.javatpoint.com/history-of-blockchain`. Accessed: Mar 11, 2023.

[17] *History of blockchain*. https://academy.binance.com/en/articles/history-of-blockchain. Published Dec 6, 2018, Accessed: Mar 11, 2023.

[18] Nakamoto, S. (2009). *Genesis Block*. https://www.blockchain.com/btc/block/0. Accessed: Mar 11, 2023.

[19] *"Blockchain: A very short history of ethereum everyone should read"*. https://bernardmarr.com/blockchain-a-very-short-history-of-ethereum-everyone-should-read/. Accessed: Mar 12, 2023.

[20] *"Cryptocurrency Regulations Around The World"*. https://complyadvantage.com/insights/cryptocurrency-regulations-around-world/. Accessed: Mar 12, 2023.

[21] *"Bitcoin (BTC) price per day"*. https://www.statista.com/statistics/326707/bitcoin-price-index/. Accessed: Mar 12, 2023.

[22] Joao Sousa, Alysson Bessani, and Marko Vukolic. *"A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform"*. In: 2018 48th annual IEEE/IFIP international conference on dependable systems and networks (DSN). IEEE. 2018, pp. 51–58.

[23] Narayanan, A., Bonneau, J., Felten, E., Miller, A., & Goldfeder, S. (2016). *"Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction"*. Princeton University Press.

[24] Larimer, D. (2018). *"Delegated proof of stake (DPoS) consensus"*. EOS.IO Technical White Paper.

[25] Satoshi, N. (2008). *"Bitcoin: A peer-to-peer electronic cash system"*.

[26] Kan, J. (n.d.). *"Economic Proof of Work"*. Retrieved April 28, 2023.

[27] FTSE Russell. (Jan 2022). *"Proof-of-Stake: A crypto path to lower energy consumption and yield"*. Retrieved from https://content.ftserussell.com/sites/default/files/education_proof_of_stake_paper_v6_0.pdf, Accessed April 28, 2023.

[28] Stefano D A, Leonardo A, Roberto B, Federico L, Andrea M and Vladimiro S. *PBFT vs Proof-of-Authority: Applying the CAP Theorem to Permissioned Blockchain*.

[29] R. Damanik, *"Pengkodean Pesan Teks Dengan Proses Penerapan Algoritma Kriptografi Secure Hash Algorithm (SHA)*, J. Inform. Kaputama, vol. 1, no. 1, pp. 48–57, 2017.

[30] R. Munir, *"Pengantar Kriptografi"*, Bandung: Informatika, 2006.

[31] B. S. Riza, *Blockchain Dalam Pendidikan: Lapisan Logis di Bawahnya*, ADI Bisnis Digit. Interdisiplin J., vol. 1, no. 1, pp. 41–47, 2020.

[32] Hansjörg Albrecher, Dina Finger, Pierre-Olivier Goffard. *"Blockchain mining in pools: Analyzing the trade-off between profitability and ruin"*. 2022. hal-03336851v2.

[33] Dylan Yaga et al. (2019). *Blockchain technology overview"*. In: arXiv preprint arXiv:1906.11078.

[34] Anastasios Kalogeropoulos. *"A Reference Architecture for Blockchain-based Resource-intensive Computations managed by Smart Contracts"*. PhD thesis. Technische Universität München Munich, Germany, 2018.

[35] Amitai P, Avneesh P, Parth S and Vinit A. (n.d). *"Blockchain Consensus: An analysis of Proof-of-Work and its applications"*.

[36] Gruber, T.R. (1993). *"Toward principles for the design of ontologies used for knowledge sharing"*. International Journal of Human-Computer Studies.

[37] Noy, N.F. and McGuinness, D.L. (2001). *"Ontology Development 101: A Guide to Creating Your First Ontology"*. Stanford Knowledge Systems Laboratory Technical Report.

[38] J. Jaramillo and J. Saucedo, "*Ontology-driven development of decentralized applications using blockchain technology*, Journal of Ambient Intelligence and Humanized Computing, vol. 11, no. 12, pp. 5309-5321, 2020.

[39] A. Lazouski, *"Ontology-based modeling and simulation of blockchain systems"*, in Proceedings of the 2019 IEEE/ACM International Symposium on Code Generation and Optimization (CGO), pp. 20-29, 2019.

[40] *Documentation for Visual Studio Code*. `https://code.visualstudio.com/docs`, Accessed April 03, 2023.

[41] *Documentation for Laravel*. `https://laravel.com/docs/9.x#meet-laravel`, Accessed April 03, 2023.

[42] *Documentation for Ganache*. `https://trufflesuite.com/docs/ganache/#what-is-ganache`, Accessed April 03, 2023.

[43] *Documentation for Brownie*. `https://eth-brownie.readthedocs.io/en/stable/index.html`, Accessed April 05, 2023.

[44] *Documentation for Android Studio*. `https://developer.android.com/studio/intro`, Accessed April 05, 2023.

[45] *Documentation for Flutter*. `https://flutter.dev/`, Accessed April 05, 2023.

[46] *Documentation for MetaMask*. `https://addons.mozilla.org/en-US/firefox/addon/ether-metamask/`, Accessed April 05, 2023.

[47] *Documentation for Protégé*. `https://protege.stanford.edu/software.php`, Accessed May 26, 2023.