

Intelligent System for Checking the Authenticity of Goods Based on Blockchain Technology

Oleh Prokipchuk¹, Lyubomyr Chyrun², Myroslava Bublyk¹, Valentyna Panasyuk³, Viktor Yakimtsov⁴ and Roman Kovalchuk⁵

¹ Lviv Polytechnic National University, S. Bandera street, 12, Lviv, 79013, Ukraine

² Ivan Franko National University of Lviv, University street, 1, Lviv, 79000, Ukraine

³ West Ukrainian National University, Lvivska Street, 11, Ternopil, 46004, Ukraine

⁴ Ukrainian National Forestry University, Gen. Chuprynka Street, 103, Lviv, 79057, Ukraine

⁵ Hetman Petro Sahaidachnyi National Army Academy, Heroes of Maidan Street, 32, Lviv, 79012, Ukraine

Abstract

The rapid spread of counterfeit in the market has become a significant issue in the 21st century. Counterfeit goods cause material damage to the producers of original goods and pose a substantial threat to buyers' health. As a result, many companies are already trying to combat this phenomenon in a variety of ways. There are two main ways to combat counterfeiting: the way to find and ban counterfeit manufacturers and the way to identify authentic goods by buyers. The subject of this bachelor's thesis research is the developed information system for the authenticity of goods based on blockchain technology. Here is how this system works. Each product manufacturer is a separate node of the P2P network. Manufacturers create units of goods as individual wallets and give them a certain balance. They then place the public and private keys of the goods in URL format and place the QR codes with this URL on the packaging of the goods, thus transferring ownership of the goods to the buyer. When the buyer scans the secret code, the balance of the interests is transferred back to the manufacturer. When scanning the public code, the system determines product status based on the balance of the product wallet: if the balance is zero, then the product is already consumed, and if not, then the product is not consumed. In this way, the buyer can determine whether this product has ever been used or unpacked before him. If the buyer receives the used product, he can return it to the store. Java programming language tools were used to implement the blockchain. RSA was chosen as the cryptographic algorithm, and the SHA256 algorithm was selected for hashing. The Spring Boot framework was used to optimize the software development process. This article consists of five sections: an analytical review of literature sources, system analysis, selection of tools and technologies for system implementation, description of the created software, and, finally, the economic part. Each section is accompanied by a detailed explanation and a summary of the work done. The developed program is provided as a set of executable JAR files. These files can be run either manually or using the included BAT scripts. The fourth section presents the results of a study of the program's behaviour depending on the variables. At the end of the work are applications that contain the most important blocks of code, a configuration file, and an example of serialization of the blockchain

Keywords 1

Blockchain technology, counterfeit good, information system, product type, counterfeit product, network communication, product life cycle, update blockchain status, user interface, intelligent information retrieval system, product key, manufacturer environment

MoMLeT+DS 2021: 3rd International Workshop on Modern Machine Learning Technologies and Data Science, June 5, 2021, Lviv-Shatsk, Ukraine

EMAIL: mr.prokipchuk@gmail.com (O. Prokipchuk); Lyubomyr.Chyrun@lnu.edu.ua (L. Chyrun); my.bublyk@gmail.com (M. Bublyk); v.panasyuk@tneu.edu.ua (V. Panasyuk); yakimtsov@ntu.edu.ua (V. Yakimtsov); roma_kov@meta.ua (R. Kovalchuk)

ORCID: 0000-0002-3584-4380 (O. Prokipchuk); 0000-0002-9448-1751 (L. Chyrun); 0000-0003-2403-0784 (M. Bublyk); 0000-0002-5133-6431 (V. Panasyuk); 0000-0001-8452-0561 (V. Yakimtsov); 0000-0001-8337-8591 (R. Kovalchuk)



© 2021 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

1. Introduction

With today's prevalence of giant companies on the commodity market, many manufacturers are afraid to compete and resort to more insidious methods of gaining income - counterfeiting. Today, thousands of manufacturers from around the world produce counterfeit goods. However, such goods cause significant harm to humanity. They threaten the material profits of the owners of goods and trust relations with the buyer and cause direct damage to the health of buyers of goods. That is why methods of combating counterfeiting and detecting original goods methods are being developed. One of such methods is an informational system for checking the authenticity of goods based on blockchain technology. This system allows the buyer to verify the authenticity of goods after or before purchase and groups manufacturers into a single network to increase this approach reliability and transparency. The method used has almost no analogues in its field of action, and strategies that work at different levels complement each other. The system is cheaper to implement and more reliable in mass use than its analogues. That is why a system that can reduce the threat of counterfeit goods in today's environment is relevant. The purpose of creating this system is to provide buyers of goods with the opportunity to verify their authenticity. Verification should be possible both before and after the purchase to return the goods in case of counterfeiting detection. The system should unite manufacturers into a single network, avoiding attachment to a particular manufacturer. In this way, each buyer can view any part of the required data. Given this transparency of data, the developed system must provide reliable protection of this information. The system must also provide additional functionality other than the main one, expanding the scope and increasing the benefits of using the system by users. Finally, the chosen method of verification should consist of multiple parts.

To achieve this goal, it is necessary to implement the following tasks:

- To study the region's problems to form the basis of existing ways of counterfeiting goods and methods of combating them. Highlight the advantages and disadvantages of the found approaches and explore the buyer's view on the attractiveness of such methods.
- Design the relationships of system elements to create a flexible and stable complex. It is necessary to build a tree of goals and decide on the type of information system based on the obtained criteria and conduct a system analysis by constructing process diagrams and a hierarchy of tasks.
- Identify the necessary technological and software tools for the implementation of such a system. If necessary, compare analogues and establish priority solutions.
- Develop the system following the established plan and collect statistical data to optimize the work with the system in the future.

The object of study: is the process of authenticating the product using transparent, immutable blockchain data. *The subject of study* is creating a unified decentralized blockchain system that provides the ability to verify the authenticity of goods for all manufacturers participating in this network.

The developed system is presented in the form of ready-to-use executable files. The main standard module is a blockchain module that manufacturers of goods can use. Other developed modules are impractical and are designed only to demonstrate the operation of the main module. The system provides a standard template that can be implemented by manufacturers of goods to access a secure blockchain network, which is why the resulting product has practical value.

2. Related works

2.1. Commodity production and counterfeiting

Buying goods has long been an integral part of the everyday life of each of us. It is difficult to imagine without these results of production of both large companies and small private enterprises.

In the past, the basis of such relations was *natural production*. It is a type of production in which individual groups (such as the family, various social groups) produced all the necessary goods to support the life and needs of this group. In other words, each group was isolated from the others and paid for goods for itself. Over time, this type of production began to displace more efficiently *commodity production*. Here, goods are already produced for their exchange or sale. Accordingly, each product has its consumer value and usefulness, with which you can agree to sell this product [1]. This type of

production has proved to be more efficient than natural production because each manufacturer provides the market with its specialised goods. Respectively, each buyer gets the opportunity to receive higher quality goods. To date, such production has developed dramatically. Many giant companies have entered the market, competing with each other for the end buyer. Due to the impossibility of competition or simply for the sake of easy profit, some producers earn by supplying *counterfeit or counterfeit goods*. Counterfeit are those goods whose production infringes the copyright of the manufacturer. For example, if someone creates a laptop using the Asus logo or certain technologies patented by this company, this product will infringe the copyright of Asus [2]. A more general concept for counterfeit is a forgery. The effect produced here may not violate someone's copyright, but the actual product and positioning will differ. For example, if you bought milk with the label saying "10% fat", although there is only 5 percent fat, it is a fake. We can conclude that counterfeit is always a forgery, and forgery is not necessarily a counterfeit.

2.2. Distribution of counterfeit products and consequences

It would be a mistake to think that counterfeit products are uncommon and quickly punishable by law. This problem has long harmed many countries and companies, causing billions in losses and taking hundreds of lives. For example, in 2017, counterfeit products caused global losses of 323 billion dollars, which is a considerable amount. The largest suppliers of counterfeit products are:

- China ~ 54%;
- Hong Kong ~ 27%;
- Turkey ~ 4%;
- Singapore ~ 2%;
- Germany ~ 1%;
- India ~ 1%;
- Macedonia ~ 0.9%;
- Thailand ~ 0.8%;
- Malaysia ~ 0.7%.

China is a leader in the production of counterfeit products globally, which is not surprising because China is also not a leader in the production of original products [3].

Thus, countries with low labour costs create counterfeits and sell them in the first world, earning vast amounts of money. The countries most affected by such products are the following:

- USA - 24%;
- France - 17%;
- Italy - 15%;
- Switzerland - 11%;
- Germany - 9%;
- Japan - 6%;
- South Korea - 3%;
- Britain - 2%;
- Other countries - 13%.

Data collected for 2014-2016 by the Organization for Economic Co-operation and Development (from now on OECD). Since most counterfeits are created on companies' products in the United States, the latter suffers the most significant losses [4].

We think everyone at least once noticed the goods of improper quality on the shelves of markets or supermarkets or when receiving a delivery from an online store. The buyer is disappointed in the store or in the manufacturer of the product, respectively, reducing the following profits. At the same time, no one can avoid the risk zone of counterfeit goods because this network has penetrated almost all areas of production of goods, so you need to inspect and check the goods when buying carefully. Fig. 1 shows world statistics on industries that suffer losses from counterfeit goods according to OECD [5].

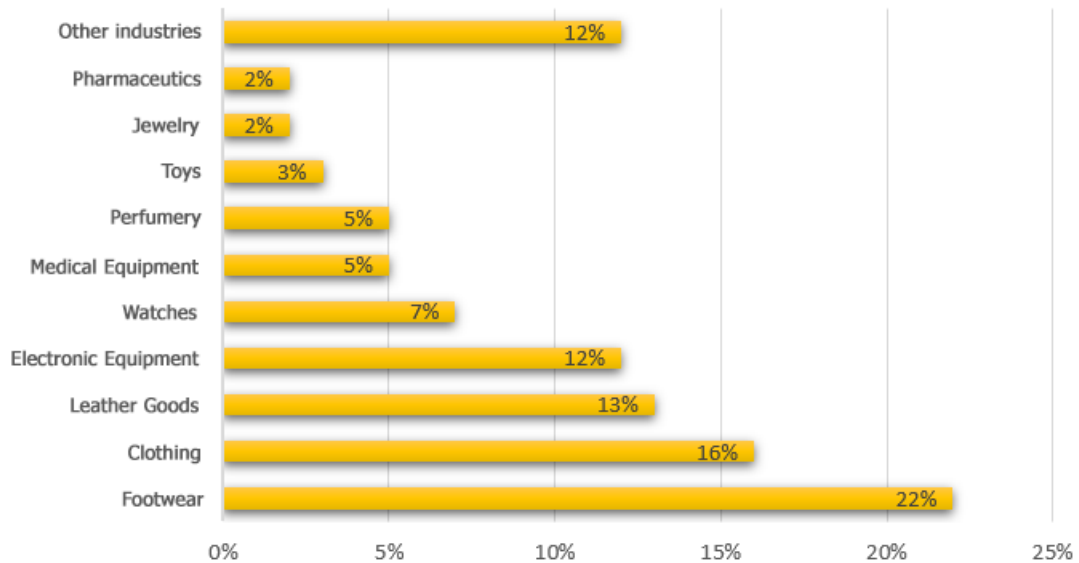


Figure 1: The industries that suffer the most from counterfeiting

Not surprisingly, in the first places of this ranking are the elements of appearance. Many people want to wear fashionable things, but not everyone has the means to do so. Someone deliberately buys fakes, and someone is led to unrealistically low discounts. The companies themselves achieve a substantial reduction in the cost of such products: they use cheaper and lower quality analogues of original materials, save on quality and production conditions, less monitor the lack of goods. So an inexperienced buyer can easily be given a substitute for natural leather. Given the low savings on the production of counterfeits, it is pretty challenging to distinguish counterfeit from the original. It is such counterfeits that cause the most damage to the owners of the original.

Previously, only material damage to buyers, shops or copyright holders was mentioned, but counterfeit goods cause more severe injury, endangering the safety and health of the buyer. And often, these threats are not obvious, and it is revealed when it is difficult to fix something. Original products are developed according to the relevant safety and quality standards and the applicable requirements set by law. When buying an actual product, the fee is for the product itself, and the fact that the product is made in compliance with all requirements, i.e. significantly reduces all potential health risks. What can not be said about counterfeits, and this must be remembered when you want to buy a cheaper version of the original product. Even thatKi, seemingly safe, goods, like clothing, can harm the human body. Fabric made of inferior quality materials in unsanitary conditions can be toxic to humans and cause various skin irritations. The most striking example of the toxicity of materials is shown to us by the cosmetics department. Researchers from Homeland Security test multiple skin products and strictly warn against buying fake options. In addition to the fact that such products are often toxic, researchers also report that they contain cyanide, lead, human and animal urine, faeces, arsenic and other hazardous substances. It applies not only to makeup but also to skin cleansers, sunscreens and others [6]. Another unobvious example of a health threat is poor quality sunglasses. According to a Brazilian Optical Industry Association survey, of the 24 million pairs of glasses produced by the country, 7 million are illegal. It is an incredible number and can hurt the eyes of millions of buyers.

Unlike the original products, which contain all the necessary sunscreens, counterfeit products have only tinted glass, which only creates the appearance of sun protection. As a result, the pupil of the eye dilates from tinted glass, but the amount of ultraviolet radiation remains the same, and the look is more damaged than without glasses at all [7].

When wearing low-quality shoes, cases of deformation of the wearer's foot are not uncommon, leading to further diseases of the human body's musculoskeletal system [7].

Counterfeit electrical goods are already a more obvious threat. Damage from them is often instantaneous and fatal. These include electric shocks or short circuits and fires.

For example, take charging for smartphones. More than half of the people on Earth use smartphones. They need to be assigned from time to time. Sometimes chargers fail, and then, when buying a replacement, smartphone owners often choose cheaper options.

Therefore, with the help of the company Apple, Electrical Safety First, has tested many counterfeit iPhone chargers purchased from Amazon, eBay and other stores. The study results showed that 98% of purchased devices have a potential risk of fatal electric shock or fire [8]. The most obvious and dangerous is the threat of counterfeit medical devices and pharmaceuticals. Defective equipment can quickly end a person's life. Such industries need the most control. When buying such goods, you should follow the minor deviations because the damage caused by them can be extensive. In developed countries, counterfeit pharmaceuticals have not become particularly widespread and are relatively successful, but the situation is deteriorating if developing countries are considered. The World Health Organization (WHO) claims that 10% of all medicines in developing countries are counterfeit. If you look at this number in terms of potential threat, it's a lot. The slightest deviation in the materials or manufacturing conditions of the pharmaceutical can lead to significant losses. Since 2013 The WHO has received more than 1,500 complaints about counterfeit goods from developing countries.

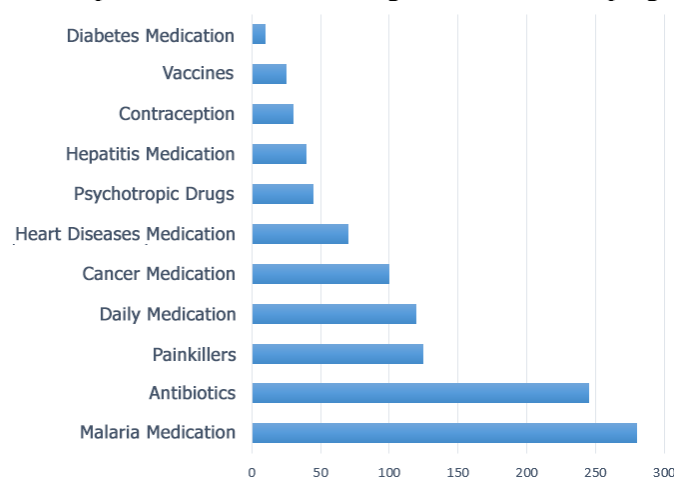


Figure 2: The most common counterfeit drugs

Malaria drugs and antibiotics have proven to be the most popular among counterfeiters. Most complaints about counterfeit goods come from Africa and account for 42% [9].

2.3. Analysis of known methods of identifying the originality of goods

The distribution of counterfeit products is both material and harmful to health. To stay afloat and minimize losses, companies must develop methods to identify genuine products to enable the buyer and various inspection services to distinguish the original from the counterfeit. There are several basic methods of product detection, from traditional to technological.

2.3.1. Legislative

This method is based on the buyer's confidence in the actions of the state to ban and seize counterfeit goods. Simply put, if the product is sold publicly and is still not banned, then it is the original. Many modern countries take the issue of counterfeiting seriously and severely punish the distribution of such products. The method mainly depends on the fact that the person who buys a counterfeit product will complain about it and, depending on the result, the source of distribution will be blocked. The Save Harbor approach works in Europe and America. It imposes certain restrictions on online stores and works as follows. The intermediary between the seller and the buyer is released from liability for copyright infringement if he complies with two requirements:

- It is possible to file a complaint;
- Reseller responds to complaints (removes or blocks infringing content).

With this approach, you can show confidence in those sellers who are in the market for a long time [10]. Fines for distributing counterfeit products are usually significant. Thus, in China, the fine for violating these rules is \$ 7,200 - \$ 72,000, and if counterfeit products threaten the buyer's health, the fine can reach \$ 288 thousand [11].

Figure 3: eBay complaint form

Pros of the approach:

- Regulated by law;
- It does not require significant material investments for implementation.

Cons of the approach:

- Based on trust;
- Accuracy is low;
- Unable to authenticate for time-tested stores.

2.3.2. Development of instructions for finding the difference between the original and the counterfeit

For each of its products, the manufacturer creates instructions to distinguish the original on specific grounds when buying the product. The method is based on physical and visual inspection. However, this method has already failed. In 2018, Baby Foot created an official website with all the necessary items to identify the original. Ironically, this instruction was used by counterfeiters to confuse the buyer [12].

Pros of the approach:

- It does not require significant material investments for implementation.

Cons of the approach:

- Based on customer awareness;
- Accuracy is low;
- The instructions work for both buyers and manufacturers of counterfeits.

2.3.3. Monitoring and blocking of suspicious resources

The approach is that the manufacturer monitors Internet resources for infringing goods and their subsequent blocking. Monitoring such resources in automatic or semi-automatic modes requires developing specialized software, respectively, and resources for this development. Identification of the original is based on the buyer's confidence that all counterfeit resources are blocked. The manufacturer of the product can both monitor and entrust this process to another company. For example, Group-IB has been providing similar services for a long time and has a wide range of functionality.

The company monitors the following areas: domain names, aggregators, bulletin boards, search engines, deep web, social networks, mobile application stores, messengers, contextual advertising.

When a counterfeit is found, the company takes all necessary pre-trial measures to block resources and provides full legal support [13].

Pros of the approach:

- Reliability is above average;
- You can entrust to another company;
- Allows you to block many counterfeit resources effectively.

Cons of the approach:

- Requires considerable material investment;
- Based on the trust of the buyer;
- Works are only for online shopping.

2.3.4. Use of artificial intelligence to detect counterfeit

Aspects of machine learning are deeply rooted in our daily lives. It is a compelling technology that can automate the work of many people. The prospects of this technology are especially tracked in recognition of something, and the detection of counterfeit goods is no exception. Different companies have already started using artificial intelligence for similar purposes and in different ways.

Alibaba group uses the latest developments in the field of machine learning to detect ads for counterfeit products. The developed system allows to detect of counterfeits on the following grounds:

- **Price:** the system detects unrealistic changes from the expected price and considers possible seasonal discounts, currency conversion, and the fact that the product can be used.
- **Image:** The system can detect the slightest deviations in the photos of potentially fake ads. When a critical number of variations is reached, the program notifies of product suspicion.
- **Description:** the system analyzes the description of the algorithm to search for suspicious phrases. The original goods do not need to impose their authenticity or novelty on the buyer. Respectively, the algorithm considers similar details.

Because the system is based on learning rather than a specific algorithm, the accuracy cannot be 100%. Potential flaws are possible, and the more goods are analyzed, the more legal interests can be counterfeited. That is why such a system can work only in a semi-automatic mode because the ban on legal goods can cause problems with the law [14]. According to a 2018 report, Alibaba confiscated \$ 536 million worth of counterfeit goods from third-party vendors. IBM has demonstrated another way to use artificial intelligence to detect counterfeits in its Crypto Anchor Verifier product. This latest technology is a micro-scanner that can be built into the cameras of mobile phones and perform a detailed analysis of the product's material and on this basis to conclude about its authenticity. The system contains data about the original product, and based on them, artificial intelligence notices differences with counterfeit products and draws conclusions based on the results of comparisons. Initially, the system was developed to recognize the authenticity of diamonds and, after successful use, the company quickly realized the potential of such technology [15].

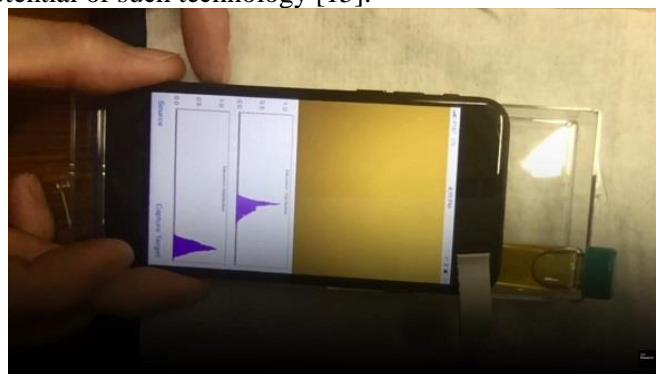


Figure 4: Crypto Anchor Verifier in action

After developing and refining training algorithms, the system successfully analysed the materials of such goods like wine, clothing, medicine, precious stones, etc. The system can distinguish expensive wine from cheap, recognize genetically modified products, analyse water quality and even find bacteria

by type of Escherichia coli [16]. In general, the technology is up-and-coming shortly we can expect the fruits of its widespread use.

Pros of the approach:

- Reliability is high
- Cheap to use
- Works in a semi-automatic mode
- The popularity of machine learning technology makes it easier to gain the trust of the buyer

Cons of the approach:

- The technology remains inaccurate
- The development of such systems can take a long time
- Requires additional implementation efforts

2.3.5. Use of unique markings to complicate counterfeiting

Another way to protect goods is to apply special identification marks on the packaging or product itself. This solution allows users to identify the original product to expect counterfeiters not to copy the technology. The application of this technology involves a pitfall, which is a constant race between the manufacturer of goods and attackers in creating the above signs. Accordingly, the manufacturer's mission is to develop and use labels with the highest possible complexity of its forgery. The attacker's mission is to copy the tags as accurately as possible to complicate the difference between the buyer [17]. Throughout its history, manufacturers have used a variety of markings, both tactile and visual. To date, the most popular way is to use holograms [18].



Figure 5: Using holograms to mark packaging

This method is a good option for the protection of goods of medium and low price category. When a balance between complexity and price is achieved, it can be transferred to mass use without severe losses in the cost of the final product.

Pros of the approach:

- Inexpensive with reasonable use;
- Average reliability.

Cons of the approach:

- Labels can be copied;
- Markings may be distorted during transport.

2.3.6. Using serialization to identify goods

The technology provides for the serialization of each unit of goods during production and their unique identifier assignment. The buyer can determine the authenticity of this unit.

The implementation of the approach is divided into two stages: entering each unit of goods in the database and referencing the database object in a unique identifier. The database must contain many products, use indexing to find the desired effect quickly, and be protected from unwanted attempts to obtain data by attackers. The unique identifier generation algorithm should avoid any possible

correlations. The identification mark itself must be machine-readable. Such types of labels as QR-Code, Data Matrix or NFC-label can be used [19].

One implementation of this approach is a joint development of Louis Vuitton SE and Microsoft. The result is called Aura Ledger and is based on blockchain technology.

The product uses NFC tags to identify goods and the blockchain as a database. The blockchain stores data on the entire product life cycle, starting with materials and production and ending with the end customer. However, the storage of such a volume of information is often redundant [20].

The use of blockchain has many advantages for this approach. The technology is characterized by high security, reliability and transparency.

Pros of the approach:

- High reliability;
- Ability to track each unit of product;
- Causes trust in customers.

Cons of the approach:

- The external label can be copied;
- Required costs are above average.

2.3.7. The results of the analysis of approaches

After considering the available approaches, we collect the primary accumulated data in one comparative table. Comparisons of methods are given in Table 1.

Table 1

Approach comparison table

Approach	Pros	Cons
Legislative	<ul style="list-style-type: none"> • Regulated by law • It does not require significant material investments for implementation 	<ul style="list-style-type: none"> • Based on trust • Accuracy is low • Unable to authenticate for unverified stores
Development of instructions for finding the difference between the original and the counterfeit	<ul style="list-style-type: none"> • It does not require significant material investments for implementation 	<ul style="list-style-type: none"> • Based on customer awareness • Accuracy is low • Counterfeit manufacturers also use instructions
Monitoring and blocking of suspicious resources	<ul style="list-style-type: none"> • Reliability is above average • You can entrust to another company • Allows you to block counterfeit resources effectively 	<ul style="list-style-type: none"> • Requires considerable material investment • Based on the trust of the buyer • Works only for online shopping
Use of artificial intelligence to detect counterfeit	<ul style="list-style-type: none"> • Reliability is high • Cheap to use • Works in a semi-automatic mode • The popularity of machine technology inspires the confidence of the buyer 	<ul style="list-style-type: none"> • The technology remains inaccurate • The development of such systems can take a long time • Requires additional implementation efforts

Use of special markings to complicate counterfeiting	<ul style="list-style-type: none"> • Inexpensive with reasonable use • Average reliability • High reliability 	<ul style="list-style-type: none"> • Labels can be copied • Markings may be distorted during transport
Using serialization to identify goods	<ul style="list-style-type: none"> • Ability to track each unit of product • Causes trust in customers 	<ul style="list-style-type: none"> • The external label can be copied • Required costs are above average

After analyzing the approaches presented in this paper, the most promising was implementing serialization of goods using blockchain technology in the project Aura Ledger from Microsoft and LVMH. This approach potential is explained by the high accuracy and security of implementation and the high trust of customers. In addition to the above advantages, the technology has room to improve and correct existing shortcomings, mainly using expensive NFC labels for each product and the possibility of multiple uses of tags by contract manufacturers. This work aims to improve the serialization system using blockchain technology, using cheaper QR-codes and the introduction of 2-stage product identification, which provides labels on the outside and inside the package to prevent the possibility of re-use of labels. The external title serves to identify the product, and the internal label contains the private data of each product and serves as a label invalid. As a result of the analysis and processing of literature sources, conclusions were drawn about the counterfeit goods problem's relevance today. Unable to compete, some companies choose to follow a crooked path and profit from the results of others by supplying counterfeit goods to the market. The world economy loses hundreds of billions of dollars due to counterfeiting every year. Moreover, low-quality counterfeit threatens the health of buyers. These two factors explain the urgency of this problem. There are many ways to identify original products, from legislative to technical. The system implemented in the course of this work is the result of improving existing blockchain-based technology. Two-stage identification and cheaper analogues of labels are introduced. As a result, we get more accuracy and less cost. It should be noted,

3. Material and methods

3.1. System analysis of the object of study

Systems analysis is a discipline that studies systems to solve problems of various origins. The subject develops and applies methods of representation and analysis of systems in various final manifestations. The developed models represent the systems in the form of relationships between the components that make up the system's structure and consider the solution of fundamental problems in the developed model. System analysis is based on a system approach [21]. A systems approach is a research methodology that considers objects as separate systems. The general statement of the process is that the whole world is a system and the objects that make up the world are subsystems. They, in turn, also consist of smaller designs. Thus, the world is a hierarchical structure of systems and subsystems.

According to the system approach of IP authentication of goods based on blockchain technology - a system that forms the structure of subsystems that interact for the general-purpose - to keep secure records of each unit of goods to identify the correct product by the buyer. The principles of the system approach consider IP in terms of their specification in the application of this system:

- According to the principle of the ultimate goal, all components of the system function is to ensure the goods authentication functionality.
- According to the principles of unity, connectivity and modularity, the system is considered as a whole from the outside. It contains an internal structure, which is generally represented in inputs (product key, manufacturer reference) and outputs (product opinion, additional product information).
- According to the hierarchy principle, the system is a hierarchical structure, on top of which are the software application modules of the manufacturer and customer application of the buyer, specified by subroutines and components at lower levels.

- According to the principle of functionality, the structure of the system is built around the functionality of secure accounting of goods and finding specific units, and when adding new functionality, the previous frame will be changed
- According to the principle of development, the system is flexible and ready to expand its functionality, and blockchain technology ideally provides a secure, uninterrupted accumulation of information.
- According to the principle of decentralization, the system goes to the minimum centralization, using the server only to establish communication between network nodes
- According to the uncertainty principle, the system is developed considering unforeseen situations and provides processing of such in the processes of creation, management and definition of goods.

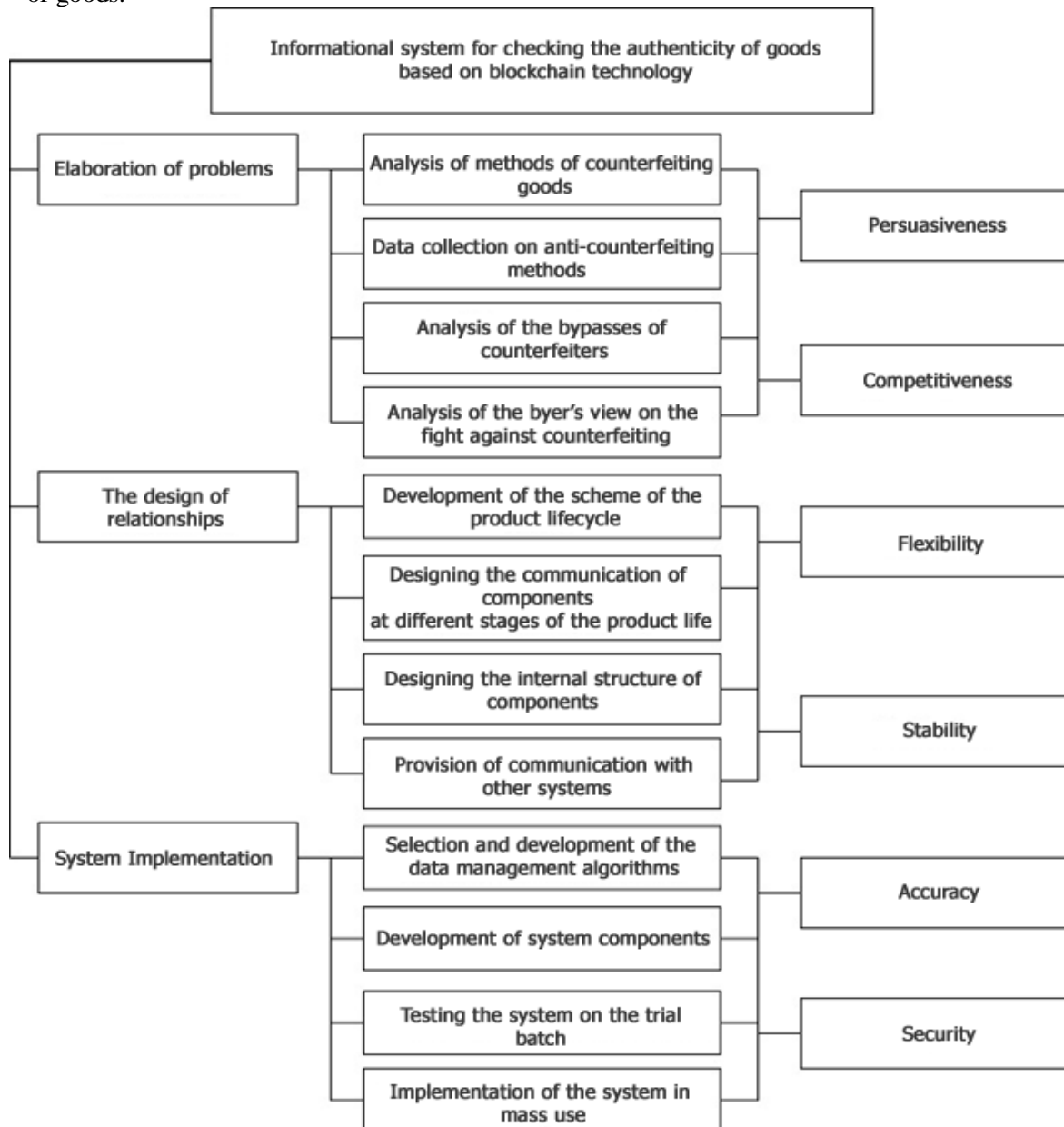


Figure 6: Goal tree

A target tree was built for further systematic analysis of the product authentication system based on blockchain technology. One of the models of systems analysis is the goal tree. This model allowed you to present the system in the form of a hierarchy of goals at different levels and identified the most priority criteria for the success of system development for a detailed description of the overall plan.

The developed goal tree consists of 4 levels: general goal, aspects, sub-aspects and criteria.

According to the developed model, the goal contains the following aspects:

- Elaboration of problems
- Designing relationships
- System implementation

These are the main aspects necessary to achieve the overall goal. In the future, each element will be specified using sub-aspects and criteria.

Elaboration of the problem is a theoretical and methodological foundation on which the following aspects will be based. Aspect is a set of analysis of issues, possible solutions and workarounds at different levels of system implementation. It consists of the following sub-aspects:

- Analysis of methods of counterfeiting goods: search for available modes of counterfeiting goods, studying production methods and copying various interests to form an initial information base, based on which methods of control will be developed identification of the most common and successful strategies.
- Data collection on anti-counterfeiting methods is an analysis of existing forms of anti-counterfeiting to highlight the advantages and disadvantages of different approaches that will be used in the development of methods of the final system.
- Analysis of the bypasses of counterfeiters: finding ways for fraudsters to compromise security methods to identify potential threats and the formation of priority areas that require a higher level of protection.
- Analysis of the buyer's view on the fight against counterfeiting is studying the buyer's point of view on the current state of the market in the fight and the development of counterfeit to find ways to establish ways of trust between the buyer and the manufacturer.

The criteria for the success of the aspect are persuasiveness and competitiveness. Convincingness shows how much the implementation of the system inspires confidence in the final buyer of goods. Competitiveness offers the coefficient of security of the system to methods of counterfeiting and ways to circumvent protection against counterfeiting. The set of criteria determines the project's potential success, provided that the requirements of the following aspects are met.

The second aspect is the design of relationships. This aspect involves determining the architecture of the system and the relationships between them at different levels of implementation. Sets the behaviour of the product in its application of sub-aspects are:

- Development of the product life cycle scheme defines all stages of the product life cycle, including the steps of creation and receipt of the product by the buyer and all intermediate stages between them.
- Designing the communication of components at different stages of the product life cycle: the result of this sub-aspect are specific patterns of behaviour of all parts of the system following the product life cycle. Such a template allows you to efficiently distribute large-scale applications with the possibility of separate consideration at different levels.
- Designing the internal structure of components determines the behaviour of each of the elements and their implementation methodology.
- Provision of communication with other systems: placement of entry and exit points for integration with other applications. It is essential to plan the places of integration before the start of the program implementation.

The criteria for this aspect are flexibility and stability. Flexibility allows the system to expand with as little effort as possible and fine-tuning individual components without global intervention in the implementation. Strength determines how well the system can operate without increasing expectations with increasing load or scale of performance. The last aspect is the implementation of the system. The aspect is responsible for all stages of realization of previously prepared plans into reality. Performance is the most extended and most demanding. It would help if you started this stage after full implementation of the previous ones. Sub-aspects are:

- Selection and development of the necessary data management algorithms: a secure accounting system requires careful selection of the required algorithms. This stage involves the analysis of possible algorithms considering the latest and most promising achievements of cryptography and blockchain.

- Development of system components is the implementation of each element and connections between them according to the created specifications and the chosen algorithmic applications.
- Testing the system on a trial batch: checking the design on a test batch of goods. Establishing the fact of meeting the requirements of all previous success criteria.
- Implementation of the system in mass use: the last stage after testing. Requires a complete system that provides all phases of putting the plan into use by end customers.

The criteria of the latter aspect are accuracy and security. Accuracy shows the ratio of the number of successfully detected original goods to all attempts to identify interests using the developed system. Security offers the degree of protection of the system from possible attacks and ways to compromise data. The next step is to choose the type of information system to achieve the overall goal. For functioning and tasks, consider the following IS:

1. The information retrieval system searches for information without its semantic processing.
2. Information and reference system. Uses queries as input to apply mathematical functions and algorithms. The results of this application are the initial data of the system.
3. Information and management system. The purpose of these systems is to solve and automate management aspects.
4. Decision support system. Such a system aims to make the most optimal decision from possible alternatives based on the collected data.
5. Intelligent information system. Systems simulate the solution of complex human problems that can not be solved algorithmically or have a significant advantage over them. Such systems include:
 - Intelligent information retrieval system - a search engine that works with search queries close to natural. Interpretation of logical connections of the question is processed using smart algorithms.
 - A computational logic system is a system that allows users to use complex machine methods through interaction with a computer.
 - Expert system - a system that provides digitization of areas that are difficult to present using mathematical models. Intelligent algorithms guarantee an effective alternative to these models.

The choice of the type of information system will be made using the method of analytical hierarchy. For this purpose, the four most suitable types were chosen: decision support system (A1), intelligent information retrieval system (A2), computational logic system (A3), information reference system (A4). The choice of design will be based on the following criteria: persuasiveness (K1), competitiveness (K2), flexibility (K3), stability (K4), accuracy (K5), security (K6). The method involves constructing a series of matrices based on expert assessments, expressed using a scale of significance. As a result, final scores will be determined for each of the four types. The scale is shown in Table 2.

Table 2

The scale of the relative importance of alternatives

Significance level	Characteristic
1	Lack of significance
2	Weak significance
3	Mediocre significance
4	Significant significance
5	Strong significance
6	Very strong significance
7	The significance is obvious
8	Insane significance
9	Absolute significance

For each of the matrices, two final parameters are defined: eigenvalues and eigenvectors. The following formula performs the calculation of eigenvalues:

$$B\lambda = (\prod_{j=1}^n a_j)^{1/n}. \quad (1)$$

In turn, the calculation of eigenvectors is performed by the following formula:

$$BB = \frac{w_i}{\sum_{i=1}^n w_i}. \quad (2)$$

Therefore, the first step will be to create a matrix of comparisons of criteria. Each of the selected system quality criteria will be compared in pairs.

Table 3

A matrix of comparisons of criteria

	K1	K2	K3	K4	K5	K6	HF	BB
K1	1.00	0.50	1.00	0.25	0.14	0.20	0.39	0.05
K2	2.00	1.00	3.00	0.33	0.20	0.50	0.76	0.10
K3	1.00	0.33	1.00	0.50	0.33	0.50	0.55	0.07
K4	4.00	3.00	2.00	1.00	0.33	0.50	1.26	0.17
K5	7.00	5.00	3.00	3.00	1.00	2.00	2.93	0.39
K6	5.00	2.00	2.00	2.00	0.50	1.00	1.65	0.22

After constructing a matrix of criteria comparisons, it is necessary to build a matrix of comparisons of alternatives to the requirements on a similar principle. Such matrices will be created for each criterion, as well as for the primary goal.

Table 4

Matrices for each criterion

Persuasiveness	A1	A2	A3	A4	HF	BB
A1	1.00	0.50	0.33	0.50	0.54	0.12
A2	2.00	1.00	0.50	1.00	1.00	0.23
A3	3.00	2.00	1.00	2.00	1.86	0.42
A4	2.00	1.00	0.50	1.00	1.00	0.23
Competitiveness	A1	A2	A3	A4	HF	BB
A1	1.00	3.00	3.00	0.50	1.46	0.30
A2	0.33	1.00	0.50	0.25	0.45	0.09
A3	0.33	2.00	1.00	0.25	0.64	0.13
A4	2.00	4.00	4.00	1.00	2.38	0.48
Flexibility	A1	A2	A3	A4	HF	BB
A1	1.00	4.00	3.00	2.00	2.21	0.45
A2	0.25	1.00	2.00	0.33	0.64	0.13
A3	0.33	0.50	1.00	0.25	0.45	0.09
A4	0.50	3.00	4.00	1.00	1.57	0.32
Stability	A1	A2	A3	A4	HF	BB
A1	1.00	0.25	0.50	0.33	0.45	0.09
A2	4.00	1.00	0.50	0.20	0.80	0.17
A3	2.00	2.00	1.00	0.50	1.19	0.25
A4	3.00	5.00	2.00	1.00	2.34	0.49
Accuracy	A1	A2	A3	A4	HF	BB
A1	1.00	0.50	0.25	0.25	0.42	0.08
A2	2.00	1.00	0.25	0.25	0.59	0.12
A3	4.00	4.00	1.00	1.00	2.00	0.40
A4	4.00	4.00	1.00	1.00	2.00	0.40
Security	A1	A2	A3	A4	HF	BB
A1	1.00	2.00	0.33	0.17	0.58	0.10
A2	0.50	1.00	0.25	0.14	0.37	0.07
A3	3.00	4.00	1.00	0.50	1.57	0.28
A4	6.00	7.00	2.00	1.00	3.03	0.55

main goal	A1	A2	A3	A4	HF	BB
A1	1.00	1.00	0.50	0.50	0.71	0.17
A2	1.00	1.00	1.00	0.50	0.84	0.20
A3	2.00	1.00	1.00	1.00	1.19	0.29
A4	2.00	2.00	1.00	1.00	1.41	0.34

At this stage, all the necessary data for the formation of the final table are obtained. The last step is to create a matrix for comparing types of information systems.

Table 5

A matrix for comparing types of information systems

Alternatives	K1	K2	K3	K4	K5	K6	Significance factor
	0.05	0.10	0.07	0.17	0.39	0.22	
A1	0.12	0.30	0.45	0.09	0.08	0.10	0.14
A2	0.23	0.09	0.13	0.17	0.12	0.07	0.12
A3	0.42	0.13	0.09	0.25	0.40	0.28	0.30
A4	0.23	0.48	0.32	0.49	0.40	0.55	0.44

According to the results of this table, the following results were obtained:

- Decision support system. A1 = 0.14;
- Intelligent information retrieval system. A2 = 0.12;
- Calculation and logic system. A3 = 0.30;
- Information and reference system. A4 = 0.44.

According to the results of the comparison, the most beneficial system is the Information and Reference System.

3.2. Concretization of system functioning

The second part of the system analysis of this work is to build a context diagram to detail the system's structure. Among the alternatives, the IDEF0 functional diagram was chosen. The process of describing at this stage requires the diagram construction itself, and it is detailing to the first and second levels. The context diagram of IDEF0 is given in Fig. 7.

The primary process in this chart is "Check the product for authenticity". This process summarizes all the structures of the lower levels of the hierarchy. It determines system inputs (arrows on the left), system outputs (arrows on the right), factors of influence (arrows at the top), resources (arrows at the bottom). Consider each of the types in more detail. System inputs define the data or objects that a system needs to get started, and the system uses that to get a unique result. The information of the developed techniques are:

- *Manufacturer link.* Since the general system is a set of software implementations of different manufacturers, the client's correct functioning requires a reference to a specific manufacturer. Depending on this, the behaviour may change.
- *Serialized product key.* A unique identifier of a unit of goods that has been encoded for correct perception of characters by most data transmission media such as URL links and others.

The outputs of the system represent the result of the functioning of the system. The outcome can be one or several. The products can be divided into primary and additional. Additional exits are optional, so you should not rely on their presence. The outputs of the developed system are:

- Conclusion about the product. The main output shows the buyer's condition, namely whether the product has already been used or not. Based on these data, the buyer concludes the authenticity of the goods.
- Additional information about the product. The other parameter provides the buyer with general data about the product or private data of the buyer about a specific unit of goods. The output depends on the software implementation of the manufacturer.

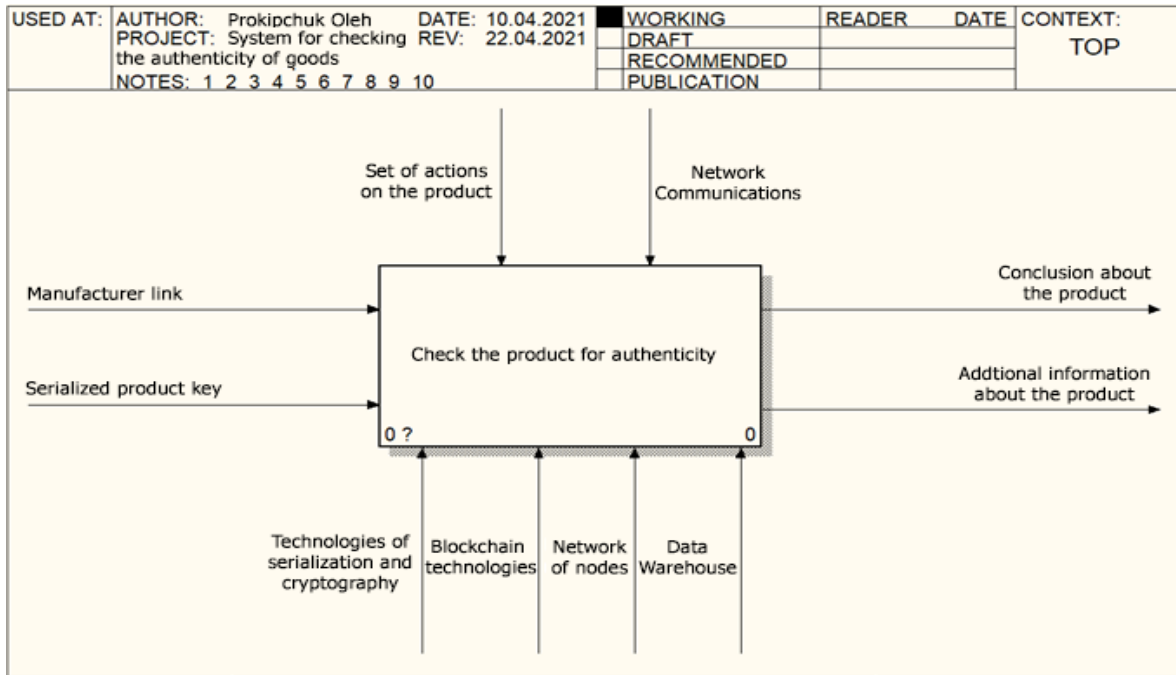


Figure 7: IDEF0 diagram

Factors influencing the system are the elements according to which the system performs its primary function and directly or indirectly impacts the outcome of the system. Factors influencing the developed system are:

- Set of actions on the product. This factor is the history of all transactions performed on the product, influencing the processing of goods according to input data.
- Network communications. Many standards and implementations of networks through which system nodes communicate with each other affect data transfer paths and algorithms in P2P networks.

System resources are a list of elements involved in the functioning of system processes. They can be both physical and intellectual. The resources of the developed system are:

- Technologies of serialization and cryptography. These technologies may differ for each manufacturer. Participate in the formation and management of keys and digital signatures. Also used for data encryption / decryption.
- Blockchain technologies. Provide basic and techniques for secure storage, transmission and supplementation of product data at the level of communication nodes in the P2P network.
- Network of nodes. Consolidation of all available manufacturers into one P2P network used for data exchange, updating and reliability.
- Data warehouse. Unlike a shared blockchain, the data warehouse is personal to each manufacturer. It is used to store information that is not written to the blockchain or is subject to change.

After that, the context diagram is detailed to the first level. Finally, the decomposition diagram is shown in Fig. 8. The result is a model consisting of 4 processes:

- Connect to the manufacturer
- Check the key for validity
- Update blockchain status
- Check the condition of the goods

In general, even such a high level of process hierarchy allows conveying in sufficient detail the structure and method of operation and use of the system

Since the diagram in Fig. 7 contained only one process, its specification has the same inputs, outputs, influencing factors and resources. Each process controls its elements, which may be different for each function. Between the processes, there are intermediate outputs that allow you to start a new approach and link them together in a single chain or organize the branching of processes.

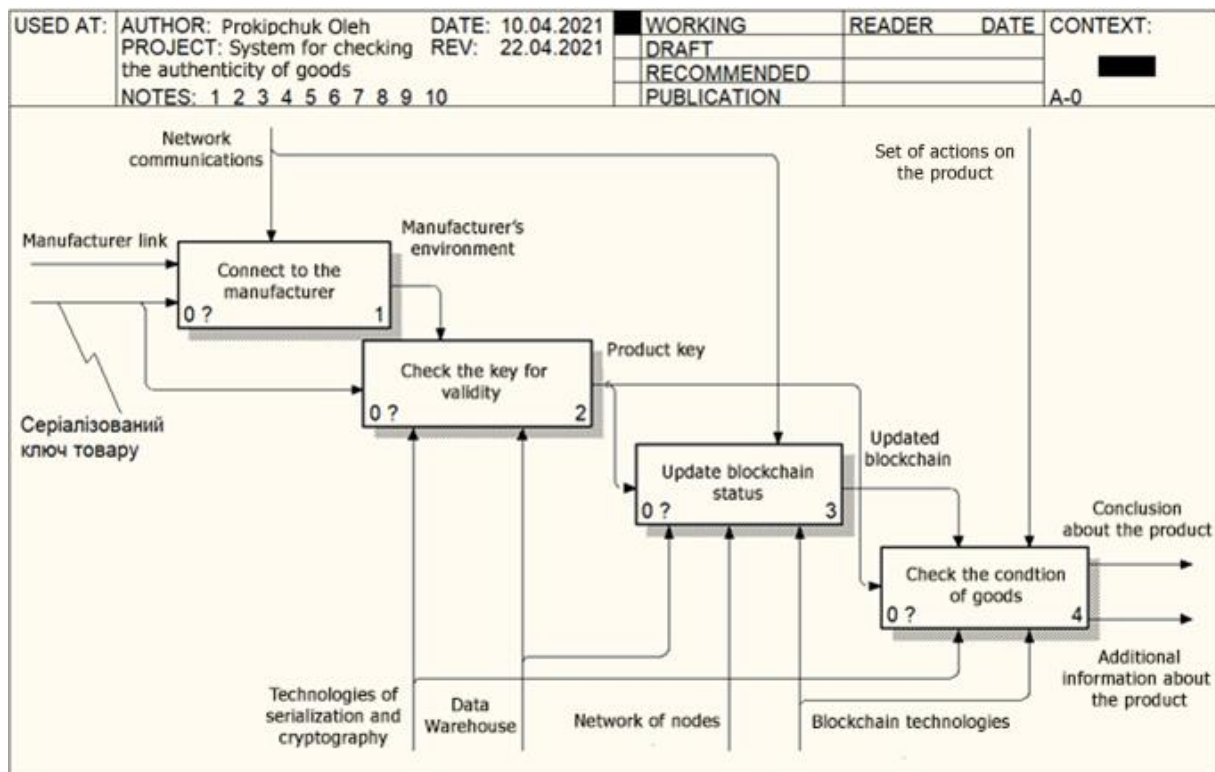


Figure 8: Decomposition of the primary process

The decomposition diagram also contains the branching of the elements. If necessary, any aspect can be applied simultaneously to all diagram processes by branching. We decompose and detail each of the methods listed above. The Connect to Manufacturer process is the first process of decomposing the primary process and acts as an entry point. Because the system consists of a plurality of vendor nodes integrated into a P2P network, the first step in the application will be to connect to a specific vendor. The link plays the central role in this process to the manufacturer, through the search and identification is carried out. In addition, the process takes a serialized key as input and outputs the manufacturer's environment, the influencing factor on which other behaviour depends. The decomposition of the process is shown in Fig. 9.

This decomposition diagram contains four processes:

- *Find a host.* The process of finding a host by link depends on the organization of network communications. The output is a specific host found on the network.
- *Establish a connection.* This process establishes a connection between the client and the host. The type of connection depends on the client, the received host, and the network communications. The result of the process is an established connection that can be used for data exchange.
- *Send data.* The process is data exchange between the client and the host. The data to be sent is the serialized key of the product, which is the process's input. Connections and network communications affect how data is transmitted. The output is the sent data.
- *Accept host data.* Receiving data is from the client based on network communications. The output of the process is the manufacturer's environment, which is then used to determine the implementation of subsequent operations.

The "Validate Key" process is a kind of filter that receives, converts and filters data. It is a second process because it requires a manufacturer's environment. The main goal is to get the key and convert it into an easy to perceive format by different algorithms and pass it on. All data that do not meet the requirements at any stage of the process are irrevocably eliminated. The product key, which is the output of the process, is necessary for the operation of the following two approaches.

The decomposition of the process is shown in Fig. 10.

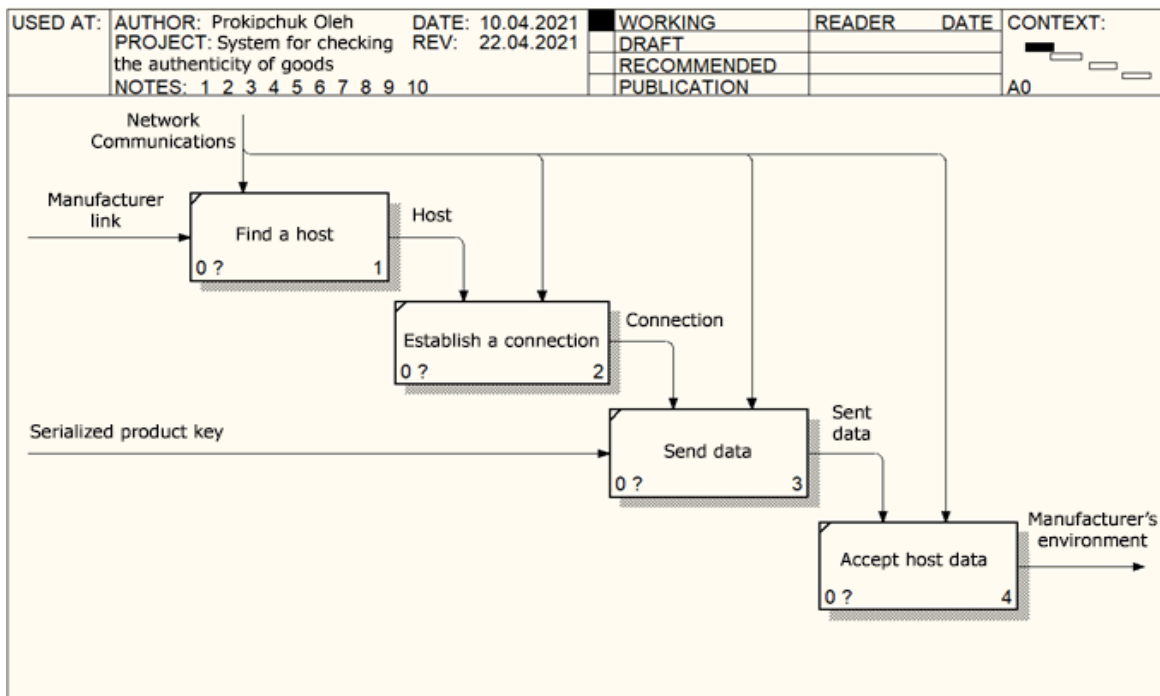


Figure 9: Decomposition process "Connect to manufacturer"

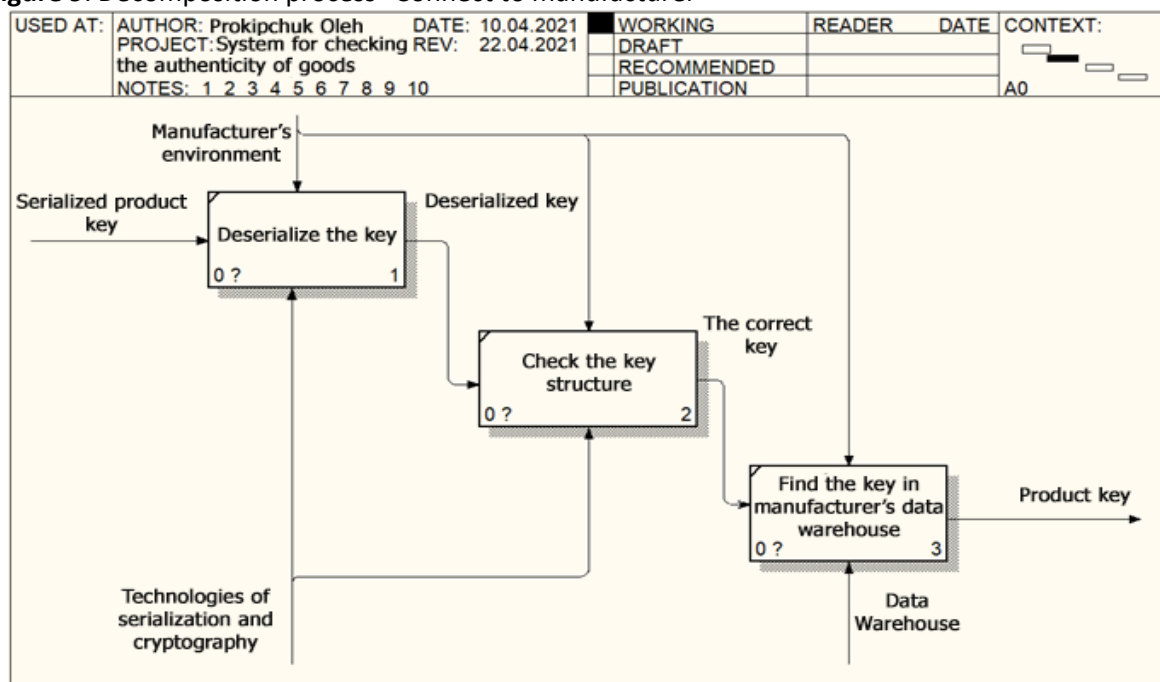


Figure 10: Decomposition process "Connect to manufacturer"

The resulting diagram contains the following processes:

- **Deserialize the key.** The function receives a serialized access and converts it from an easy-to-transmit format to an algorithm-friendly format. The implementation of the process depends on the environment of the manufacturer and uses serialization technologies. At the output, we get a deserialized product key.
- **Check the critical structure.** The process uses the obtained deserialized key and verifies that it is a valid cryptographic key built according to the necessary algorithms. The manufacturer's environment affects this process. The output is the correct key. All incorrect keys are eliminated.
- **Find the key in the manufacturer's data warehouse.** Searches the manufacturer's database for the access obtained as a result of the previous process. The manufacturer's environment affects the

organization of data and behaviour in cases of finding and not finding the product. The way out is the key to the found goods.

The Update Blockchain Status process is the third process among the IDEF0 chart decomposition processes. It can be performed simultaneously as the previous one, but it is performed synchronously to prevent unnecessary actions. The purpose of the process is to find the nodes of the manufacturers with the most up-to-date blockchain, request updates from them, and update your blockchain with the correct update. The result is an updated blockchain that the following processes can work.

The process decomposition diagram is shown in Fig. 11.

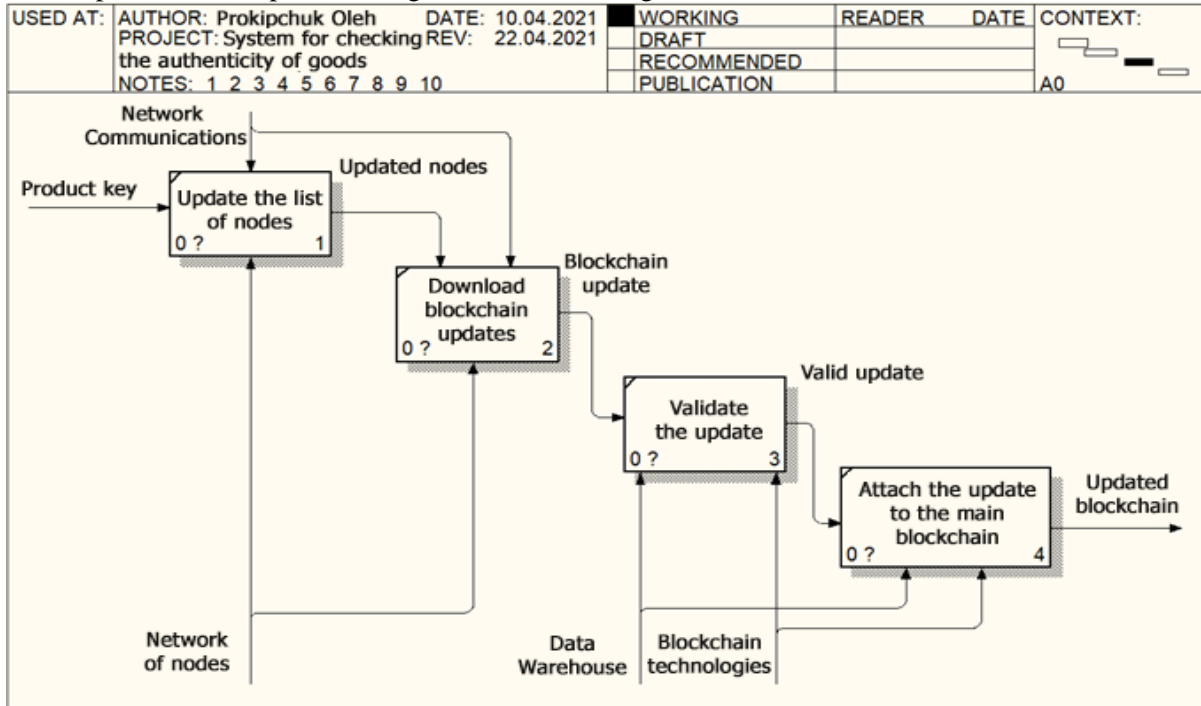


Figure 11: Decomposition of the "Update blockchain status" process

The developed diagram contains four processes:

- Update the list of nodes. The process does not use an empty product key as a trigger to start operations. Loads a list of P2P network nodes is by communicating with neighbouring nodes or contacting a dedicated server. Network communications affect the nodes obtained during the download process. The output is a list of updated nodes.
- Download blockchain updates. Monitors loaded P2P nodes in search of nodes with the most significant number of blocks in the blockchain. Gets the blockchain update from the node with the highest priority. The solution is to update the blockchain. Network communications affect the download process.
- Validate the update. Different end devices can be P2P network node. Among them may be attackers who seek to replace/distort the data, so each update received is checked for validity. Uses manufacturer data store and blockchain technology is for the verification process. A valid update is a way out of the process.
- Attach the update to the main blockchain. After receiving the correct update, you must add the update blocks to the manufacturer's main blockchain stored in its repository. The connection is performed using blockchain technologies. The output of the process is an updated blockchain that reflects the most current status of goods.

The process "Check the condition of the goods" is the final process, which results from the cycle "Check the goods for authenticity". One way or another, the process requires data obtained in previous methods or their derivatives. The process is responsible for collecting the necessary data in the blockchain according to the input parameters, processing this data to obtain the result, and performing additional actions to acquire other data about the product, which will be similarly sent to the customer.

The decomposition of the process is shown in Fig. 12.

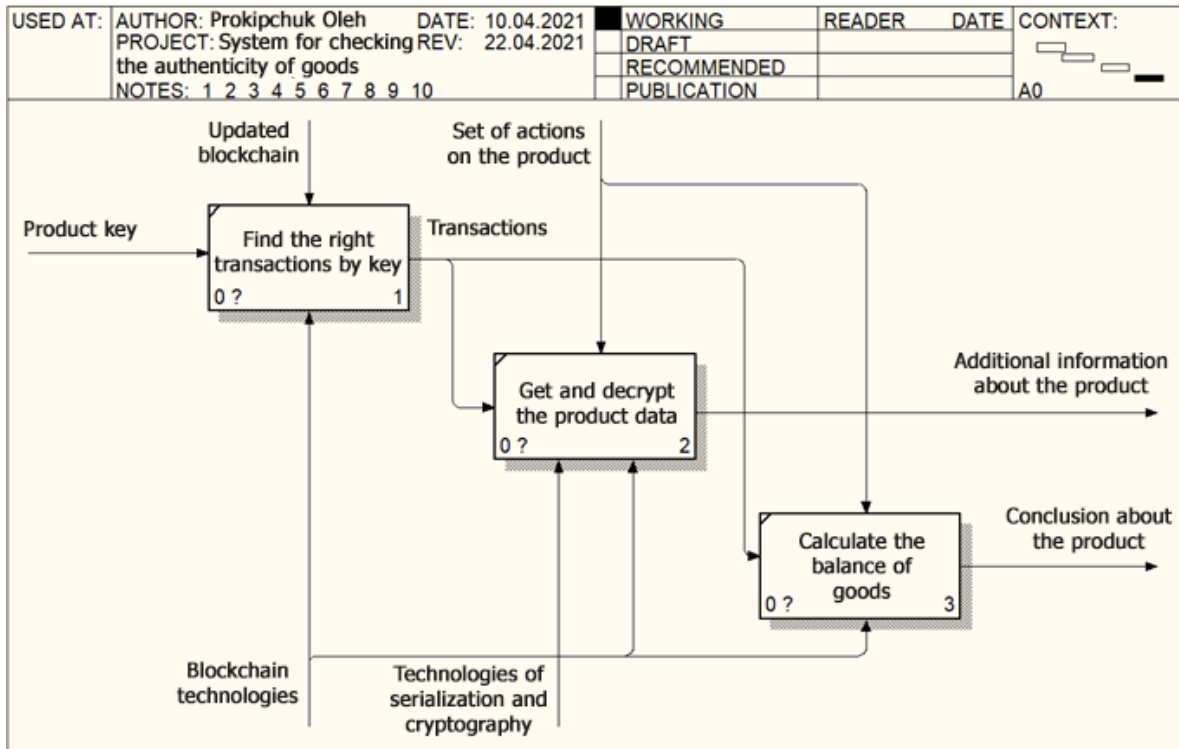


Figure 12: Decomposition of the process "Check the condition of the goods"

The developed diagram consists of the following processes:

- Find the correct transactions by key. Scans the blockchain and selects all transactions, the sender or recipient of a product with a permit received from the transaction input. Uses the updated blockchain for search is Blockchain technologies search. The way out is the transactions found.
- Get and decrypt product data. Analyzes the received transactions for encrypted data and translates them using the product key. Uses cryptography and blockchain technologies to manipulate data. At the output, we receive additional data about the product.
- Calculate the balance of goods. The most critical process is to calculate the condition of the product using the received transactions. With the blockchain technologies help and according to the executed transactions, the process gets the goods balance and sends the conclusion to the client.

3.3. Building a hierarchy of tasks

The last stage of system analysis is to build a diagram of the hierarchy of tasks, which reflects the system's structure in the form of an order. The diagram is given in Fig. 13.

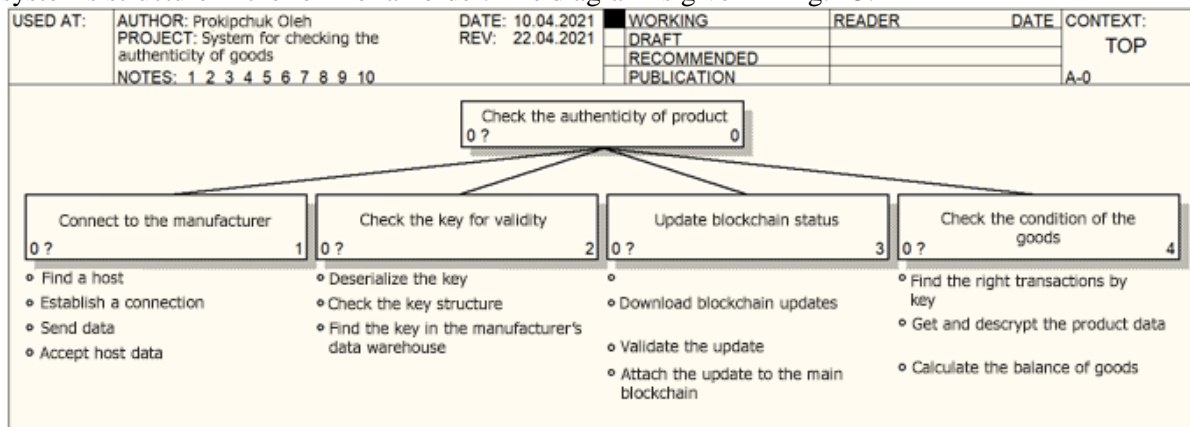


Figure 13: Task hierarchy diagram

There are several ways to represent such a chart. A tree structure was chosen to perform this work. The diagram is made based on the developed functional diagram. The main task is to "Check the authenticity of the product". It consists of the following subtasks:

Connect to the manufacturer. It consists of the following subtasks:

- Find a host;
- Establish a connection;
- Send data;
- Accept host data.

Check the key for validity. It consists of the following subtasks:

- Deserialize the key;
- Check the critical structure;
- Find the key in the manufacturer's data warehouse.

Update blockchain status. It consists of the following subtasks:

- Update the list of nodes;
- Download blockchain updates;
- Validate the update;
- Attach the update to the main blockchain.

Check the condition of the goods. It consists of the following subtasks:

- Find the correct transactions by key;
- Get and decrypt product data;
- Calculate the balance of interests.

During the second section of the work, a system analysis was performed for the topic "Information system for the authenticity of goods based on blockchain technology." As a result, a tree of goals of the system was constructed, which allowed to allocate them to a hierarchical structure to obtain aspects, subspects and criteria. Based on the criteria, the type of system was selected using the method of analytical hierarchy. The kind of system established is the Information and reference system.

After that, to detail the system's structure, a functional diagram IDEF0 was constructed, and this diagram was decomposed to the first and second levels. The result is a detailed structure of processes and system resources. The final stage of the section was to build a diagram of the processes hierarchy in the form of a tree structure, which allowed to consider the system in problems and subtasks.

3.4. Software tools for solving the problem

3.4.1. Selection and substantiation of means of problem-solving

At this stage, it is necessary to analyze the available tools for solving the problem and choose the best strategies for use in the implementation of the information system. To form a set of tools, you need to determine the issues that need to be solved. So the main tasks are the following:

- It would help if you created software that allows you to operate data and blockchain in a dialogue mode with the user interface.
- The developed software must have high performance of arithmetic and algorithmic operations, be flexible, and adjust and scale.
- The developed software must network connectivity to other applications, particularly the same software object on other end devices, for their integration into a P2P network.
- It would help if you created a P2P server to control the exchange of information between nodes.
- You need to create a client application in the form of a mobile application or website.

After the generated list of tasks, the selection of tools and technologies for their solution begins. First of all, we choose the technical means for the implementation of the leading software application. The technologies used in creating the program are as follows: Java, Spring Boot, JavaFX, GSON, Maven. The same set of technologies was used to implement the P2P server and implement the website's server part. The following technology stack was used to create the client part: HTML, CSS, JavaScript.

First of all, you need to choose the primary programming language. Possible options include Java, Python, C ++, C #. To achieve high performance, the language must be compiled. From the available options, the Java programming language was chosen.

Java is an object-oriented programming language developed by Sun Microsystems, which eventually became the property of Oracle. Programs created using this language are compiled into bytecode, which allows you to achieve high performance. Java is compiled into unique code that is recognized by a Java virtual machine (JVM). Thanks to which programs can be run on any operating system that supports JVM. This multi-platform is a big plus for the developed system, as it allows you to build clients and servers for different platforms [22]. Java is also object-oriented, which will let the developed method to be flexible and scalable. Developing software using a pure programming language is a very complex and costly task. That is why it is necessary to introduce support for development tools and aids [23]. The first essential tool should be a framework for centralized object management. Modern corporate programs are extensive and consist of hundreds of classes. With such an organization, the program can quickly become highly connected and unscaled. To avoid such consequences, the program must follow the following rules:

- Program classes should be weakly connected (**Low Coupling**);
- Each class should have only one duty (**Single Responsibility Principle**) [24].

When one object creates another object, it performs the duty of creating objects, i.e. according to the SRP, it should not be engaged in anything other than the construction of objects. Also, objects perform several functions at once without even considering the creation of other objects. To help in this situation come three design templates: strategy, control inversion and injection of dependencies.

Strategy is a design template that allows the class to expand its functionality by delegating additional work to ancillary objects. Thanks to this template, all classes will have one primary responsibility, and all others will be delegated.

Dependency Injection is a design template that changes the order of assignment of auxiliary objects to the main. The class does not create additional objects itself but only declares containers for them. The control program must inject other objects into the main. In this way, the object avoids the obligation to construct objects.

Inversion of control is a design template, which is a module that registers and constructs program objects. This template allows you to automate the process of injection of dependencies, thereby significantly reducing the connectivity of the program [25].

The program's high flexibility and scalability can be achieved Using these design templates. 2 possible frameworks were selected for the developed system: **Google, Guice** and **Spring Boot**. Of which we chose the latter because of its greater functionality [26].

The next step is to choose a tool to create the user interface. Such devices for the Java programming language are small and consist of **Swing** and **JavaFX**[27].

Both tools allow you to create a high-quality user interface, and both were developed under Oracle's guidance. As of version 9 of Java, JavaFX is no longer part of the main JDK development package and is being developed separately from Oracle. JavaFX was chosen for this work because of its convenience and higher functionality, particularly the ability to create designs in FXML files with XML mark-up when Swing can only generate and populate components programmatically.

One of the processes, the manual implementation of which requires a lot of extra time, is serialization and deserialization. Serialization is converting an object into a byte or character format convenient for transferring in non-Java environments. Deserialization is the reverse process, i.e., converting a character or byte format back into a program object. The symbolic data transmission format was chosen for this program, **JavaScript Object Notation (JSON)** [28], because it is easy to implement and easy to read. It is required to track and demonstrate intermediate program results.

Possible implementations of this technology are **Google GSON** [29] and **Jackson**. GSON was chosen because of its higher popularity, which provides more active work on the software product.

For convenient development and ensuring the correct interaction of all software components, it is advisable to use one assembly system. These systems automatically compile and assemble all modules into one software package and control all parts and the main product. There are three following systems for Java: **Ant**, **Maven** and **Gradle**. Today, Ant is an outdated system. Of the two modern systems, was chosen **Maven** [30] due to the high prevalence of its repositories. Although Gradle is a more modern and advanced system, not all components are compatible with it.

One of the critical points in the development of the main program is network communications. The required method is data exchange using the **TCP** protocol. Unlike **UDP**, **TCP**[31] monitors the data sent for possible losses, which is necessary for a secure and accurate system. One way to implement

such communication is to use **HTTP** application layer protocol because it is based on **TCP**. In addition, there are many frameworks for Java that allow you to implement such functionality quickly.

Nevertheless, for executing all P2P communications, the standard Java functionality was chosen to create Socket / Server connections. This technique allows you to establish a channel between the client and the server, which can both listen and edit. This approach is necessary for implementing a P2P network because higher-level protocols do not regulate the communication of such a connection to be arbitrary for the developed product.

Although the HTTP protocol was implemented to implement the P2P network, it is an integral part of the webserver. For the client application, an option was chosen on an Internet site that communicates with the server using the HTTP protocol. HTTP is an application layer protocol based on the TCP / IP protocol stack. Communication between the server and the client is performed through a request-response template. The client generates a request, fills it with data and sends it to the server. In turn, the server processes the request and sends a response to the client. Communication takes place using one of the following methods: GET, POST, PUT, DELETE, OPTIONS, HEAD, PATCH, TRACE, CONNECT [32]. In most cases, the first four methods are used.

There are several basic ways to implement such a server, but they are all based on a Servlet container. **Servlets** is a software unit (object) that processes a client request in a separate thread. The most popular implementation of the Servlet container is **Tomcat** [33]. But **Spring Boot** already used in the developed program contains an extension over the standard behaviour of Tomcat, which increases its functionality and simplifies interaction with other components of Spring Boot, which is why this tool is chosen for the implementation of the webserver.

In turn, the client part is a web application. Any website is based on three leading technologies: HTML, CSS, JavaScript. This stack of technologies has no relevant analogues today, so the choice is quite simple. At the beginning of the development of the Internet, various tools could be used to create a site, but they were all pushed out of the market over time.

HTML is a hypertext markup language. Allows you to build the framework of the future site using a set of predefined markup tags. At this stage, the site is not yet stylized and contains only the content. Although each tag can be given properties manually, it is not recommended. There is a specialized CSS language for such purposes.

CSS is a language of cascading tables and styles. Based on the name, this language performs the function of stylizing the content of the site. This process is as follows:

- The required tags are selected using selectors. Many selectors are defined for this language, such as selectors by tag name, class, attributes, content, etc.
- Styles are already defined for a specific selector. As a result, it is possible to stylize many blocks at once, avoiding the manual stylization of each tag.

JavaScript is a scripting language that is executed every time the site is launched. Used to perform all the site's logic and ensure user interaction with the website and for asynchronous communication with the server. A language execution environment, such as a browser, usually restricts a language's access to the owner's device resources, so visiting websites is utterly secure because dangerous scripts have been prevented from running. Since the language is a script, the user can call new commands directly while using the site [34]. Once the technology stack has been formed, it is time to choose software development and testing environments. The first such environment should be developing, compiling and running code in the Java programming language. Today's main competitors in the market are Eclipse and JetBrains IntelliJ Idea. NetBeans used to be actively involved in this fight, but now its market share is minimal. A general overview of the previous tools shows that these are compelling and functional tools. A more detailed comparison is given in Table 6.

Table 6
Comparison of IntelliJ Idea and Eclipse

Parameters	IntelliJ Idea	Eclipse
System requirements	At least 2 GB of RAM	At least 0.5 GB of RAM
Distribution method	Free with the paid version	Free
Debugging	An advanced set of debugging tools	Standard debugging tools

Plugins	750+ plugins	1250+ plugins
Autocomplete	Automatic	Using the keyboard shortcut Ctrl + Space
Productivity	Optimized for indexed operations	Faster at high loads
Refactoring Design	An advanced set of tools Modern design, easy to use	The standard set of tools Outdated and overloaded design
Orientation	Small and medium projects	Big projects

A student license for the software product IntelliJ Idea Ultimate Edition was obtained to perform this work [35]. A second necessary tool is a tool for debugging and testing a client web application. Most browsers today have built-in tools designed for this. The most popular of these are Google Chrome, Firefox and Opera. Among the tools listed above, the choice fell on Google Chrome because of its functionality and ease of use.

3.4.2. Technical characteristics of selected software development tools

All hardware and software in paragraph 3.1. are used to perform this work in a specific configuration. Each of them contains unique parameters that characterize a particular application. Most tools include features such as product version, developer, system requirements, etc. For selected software products, we give their detailed characteristics. Technical characteristics of the IntelliJ Idea are shown in Table 7. Google Chrome specifications are listed in Table 8.

Table 7

Technical characteristics of IntelliJ Idea

Attribute	Value
Date of issue	03/27/2019
Product version	2019.1
Subscription version	Ultimate Edition
Owner	JetBrains
Development language	Java
Operating Systems	Microsoft Windows 10/8/7 / Vista / 2003 / XP, OS X, Linux
RAM	2 GB recommended
Hard disk space	At least 1.3 GB
The minimum version of the JDK	1.8
License	Apache 2.0

Table 8

Google Chrome specifications

Attribute	Value
Date of issue	11/02/2020
Product version	86.0.4240.185
Type	The current is stable
Owner	Google LLC
Development languages	C ++, Assembler, Python, JavaScript
Operating Systems	Microsoft Windows 10/8/7 / Vista / 2003 / XP, OS X, Linux, Android
RAM	512 MB
Hard disk space	350 MB
License	BSD

Technical characteristics of Java are given in Table 9.

Table 9

Technical characteristics of Java

Attribute	Value
Date of issue	07/16/2019
Product version	SE 11.0.4
Type	Open JDK
Owner	Oracle Corporation
Development languages	C, C ++, Java
Operating Systems	Multiplatform
RAM	128 MB
Hard disk space	124 MB
License	GNU General Public License

Technical characteristics of Maven are given in Table 10.

Table 10

Technical characteristics of Maven

Attribute	Value
Date of issue	11/22/2015
Product version	3.6.3
Type	The current is stable
Owner	Apache Software Foundation
Development languages	Java
Operating Systems	Microsoft Windows 10/8/7 / Vista / 2003 / XP, OS X, Linux
RAM	128 MB
Hard disk space	64 MB
The minimum version of the JDK	1.7
License	Apache 2.0

The list of versions of used libraries and frameworks is given below:

- Spring Boot - v.2.4.2. Developer: Pivotal Software;
- JavaFX - 13. Developer: Sun Microsystems;
- GSON - v.2.8.6. Developer: Google LLC.

3.4.3. Blockchain and means of its implementation

The most complex and large-scale part of this work is the development of blockchain functionality.

Blockchain is a distributed database, which implies a chain of interconnected blocks, as the name suggests. This technology was first introduced in 1991 by Scott Stornett and Stuart Haber. Their work described the possibility of creating secure documents, which were combined into a structure of cryptographically protected linked blocks. Today, blockchain has become very common. It is used in many areas of everyday life, especially in cryptocurrencies such as Bitcoin, Ethereum, Binance Coin, Tether, Bitcoin Cash, Litecoin and others [36]. The principle of operation of the blockchain is shown in Fig. 14. The main qualitative characteristic of this technology is the security and transparency of such a data structure because you can only add new blocks to the chain of blocks without the ability to edit them. This means that any data that enters the blockchain becomes visible to all its members and remains there in its original format throughout the blockchain existence. There are many blockchain implementations, so its behaviour and characteristics can often vary from one product to another. The basis for this work was the first successful and most famous implementation of the blockchain - Bitcoin.

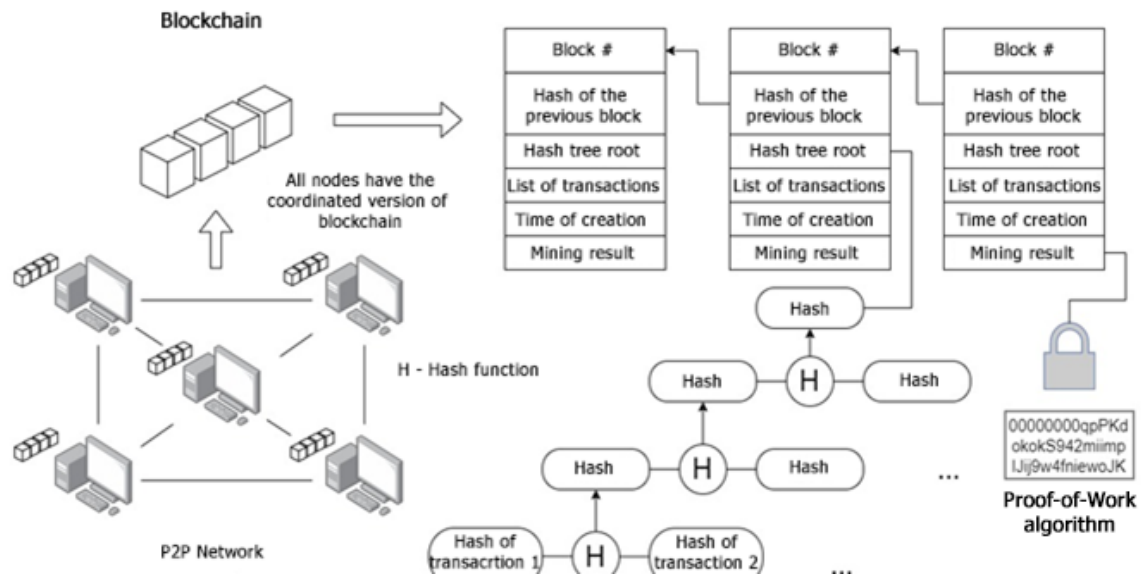


Figure 14: Principles of blockchain operation

Block is the central structural unit of the blockchain. The only operation to change a blockchain is to add a new block to an existing chain. Nodes, blockchain networks are engaged in the creation of blocks. In addition to being a structural unit of a chain, its most important function is to be a container for data that must be immutable.

The block consists of the **title** and **primary data**. The block header is also called its metadata. This division was created since prolonged use of the blockchain can reach substantial sizes. So the cryptocurrency Bitcoin today occupies more than 200 GB. It is not a problem for specialized fixed devices, but downloading 200 GB of data can be a problem for mobile devices. The block was divided into a header containing the minimum required data set and the central part of optimising. Due to this, there is an opportunity to develop **mobile clients** that work only with block headers. Such mobile clients have one disadvantage. If necessary, they have to download the required data, and when using such a client, it is essential to make sure that this download is carried out from a reliable source. The structure of the block is given in Table 11. Thus, the size of the block header is 112 bytes. Next, consider each of the components in more detail [37].

Table 11

Block structure

Field	Size	
Title		
Hash of the previous block		32 Bytes
Block hash		32 Bytes
Hash root		32 Bytes
Creation time		8 bytes
Appendix (nonsense)		8 bytes
Main part		
Hash tree	It depends on the number of transactions	
List of transactions	Limited to implementation	

Block Hash is a unique block identifier obtained by passing a block header through a hash function. The hashing algorithm that is used to perform this work is **SHA-256**. The algorithm was developed by the US National Security Agency and generated a 256-bit hash for any data.

To form a chain, the blocks must be interconnected in a certain way. Each block contains a reference to the previous block in the form of **Previous Block Hash**. As a result, having the last block, it is

possible to trace the initial block, moving by previous hashes. The hash of the last block is part of the data on the basis of which the hash of the current block is calculated. In this way, protection against data distortion is achieved. If you change the data of any blockchain block, all subsequent blocks become invalid.

Hash root (Merkle root) is the root of the hash tree. **A hash tree (Merkle tree)** is a tree built on the hash of transaction values. The tree is made according to the following algorithm:

- Construction begins with the leaves of the tree, which are the hashes of each of the submitted transactions;
- Transaction hashes form a queue;
- Two elements are selected from the queue, and on their basis, a parent node is created;
- The parent node is created by skipping the concatenation of children's hashes through the hash function;
- The resulting node is added to the queue of the next stage;
- The operation is repeated until the current queue ends, after which the queue of the next step becomes current;
- The general algorithm is repeated until there is one element left in the final queue. It is the root of the tree.

Thus, each parent node is a hash of the child node hashes [38-40].

This tree is used to save the computing resources of the device. When checking a block for validity or checking that a transaction belongs to a block in a mobile client, it is necessary to calculate the hashes of all transactions. The number of transactions in the unit can reach several thousand, so the calculation of several such functions negatively affects the program's performance. When using a hash tree, the complexity of the operation of checking the transaction belonging to the block is $O(\log(n))$, which with a large number of transactions gives a significant advantage over $O(n)$.

Typically, a hash tree is used even with a small number of block transactions.

Time of creation (Time) - time showing the exact date of creation of the block served in milliseconds. Creating a block and the time of its writing to the blockchain can differ significantly, which is why the blocks of the chain do not necessarily have to go in chronological order.

Application (Nonce) - a specific numerical value that is added to each block. This numeric value is the result of block mining. This number does not affect the block's content but is only an application that makes a difference in determining the hash of the block.

Blockchain nodes always accept only the longest chain of blocks. All others are considered irrelevant and are discarded. The blockchain is already protected from data distortion by hashing it, but what prevents an attacker from creating a long chain of blocks so that other nodes will take the fake blockchain for real? An algorithm was developed to avoid such a situation

Proof-of-Work. The algorithm is that the creation of each block must be accompanied by specific resource and time costs of the processor. The process that provides such costs is

Mining (Mining). Mining a block means selecting such a value **None** that the block hash starts with a certain number of zeros. The number of zeros that must be at the beginning of the hash is determined **complexity of mining.** Complexity is a configuration parameter that can be used to control the duration of mining. There are two main algorithms for selecting the value of Nonce: iteration and the use of random numbers. Random numbers allow you to run this algorithm in parallel in multithreading mode to increase mining productivity. Thus, to create a fake circuit longer than the current one, it is necessary to perform Proof-of-work for each new circuit block. If you take 10 minutes for the length of mining and try to compromise a blockchain with a size of 1000 blocks, it will take 10,000 minutes. It makes the possibility of such attacks virtually impossible.

The primary data of the block are its transactions. **Transaction** stores protected data that can interact with each other. The person who creates the block (miner) is not necessarily the owner of the transactions. The node receives transactions from other nodes, combines them into one block and adds them to the blockchain. The content of the transaction can be anything. In the Bitcoin system and this work program, the filling in the numerical balance transfer form from one purse to another is used. The transaction consists of transaction id, sender address, recipient address, transferred balance, inputs, outputs, additional data and electronic digital signature. Consider each of the elements in more detail:

Transaction Id is its unique identifier, which is assigned after its creation.

Sender and recipient addresses are the public keys of the sender's and recipient's wallets.

A **purse** is a set of **public** and **private cryptographic keys**. For the wallet, you can calculate its balance. The sender uses the wallet to create a money transfer transaction to another wallet. The wallet's public key is used as its address as well as to verify the digital signature. The private wallet key is used to create **EDS** transactions. Public and private keys are generated using an algorithm **RSA**. The generation of such keys depends on random numbers. In turn, the algorithm for generating such numbers must have no patterns.

RSA is an asymmetric cryptographic encryption algorithm based on the properties of large prime numbers. The purpose of the algorithm is to encrypt data and create EDS. The algorithm is widely used today and is used in many applications [39]. An example of the RSA algorithm is shown in Fig. 15.

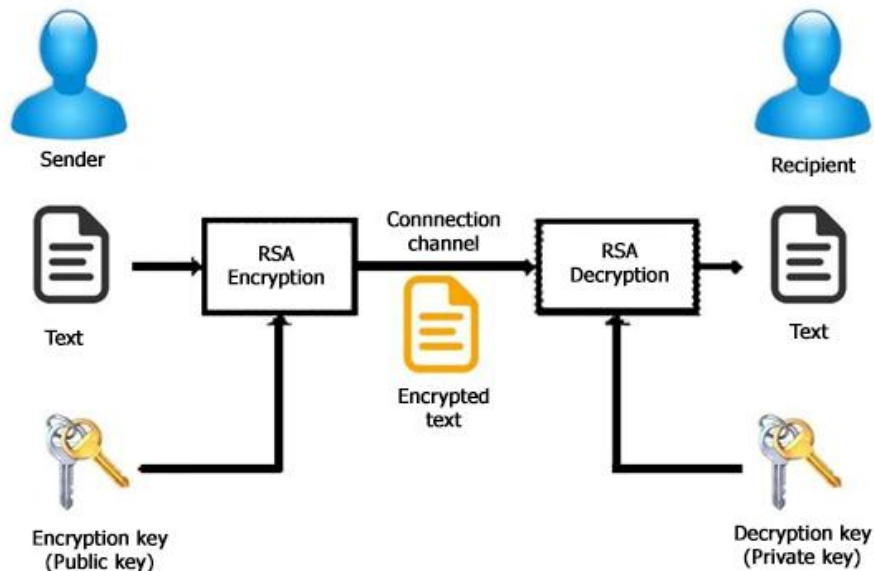


Figure 15: Example of encryption using the RSA algorithm

An **electronic digital signature (EDS)** is a type of signature applied to electronic documents. Such a signature allows you to confirm the signed data's immutability and establish the author of the signature. According to the laws of Ukraine, an electronic signature is equated to a non-electronic one. In a blockchain, such a signature is a guarantee of the immutability of transaction data. If you change any signature data, it becomes invalid. Any transaction must be signed with an EDS. It is necessary to establish that the transaction was made by the owner of the sender's wallet, not someone else. The signature is created using a private key that is known only to the wallet owner. And you can check the signature with public access that is known to all.

Additional data is arbitrary data that is written to the transaction. Such data may be unrestricted. In the developed system, this field is used for the storage of private data of the goods.

Transferred balance is the numerical value of the balance that the sender wants to transfer to the recipient in this transaction.

Each transaction must contain inputs and outputs.

Transaction input is a new transaction output that contains a specific numeric value that the wallet owner can operate. The wallet owner cannot send more than the sum of the transaction input values. Once input is applied, it becomes used and can no longer be used as a transaction input. Inputs are converted to unused outputs that can be used later.

The output of the transaction is a numeric value passed from one wallet to another. Such a result can be spent, i.e. used as the input of another transaction, or unused.

After the transaction, two outputs are created: the amount sent to the recipient and the input values balance after sending to the recipient. The balance is sent back to the sender so that he can use it in subsequent transactions. The amount of new exits is the balance of the wallet.

To get started, you need to combine nodes into a single network because the reliability of the blockchain directly depends on the number of users. When implementing the standard, it was decided to move away from the standard centralized client-server model favouring the decentralized P2P model.

P2P is a network topology in which each member can simultaneously act as a client, making requests to one group of nodes and being a server, responding to requests from another group of nodes. Such a network is not hierarchical, and all its end devices are on the same level. If one of the nodes fails, the network can still work correctly. The nodes of such a network are called peers.

There are several ways to organize such a network: decentralized, centralized and mixed. Decentralized P2P consists only of nodes located on 1 level. To find nodes, adjacent peers are used. In centralized P2P, there is a server that knows the addresses of the nodes. When communicating, the peers contact the server to obtain the addresses of other peers. The mixed-method uses both approaches.

The view of the P2P network is shown in Fig. 16.

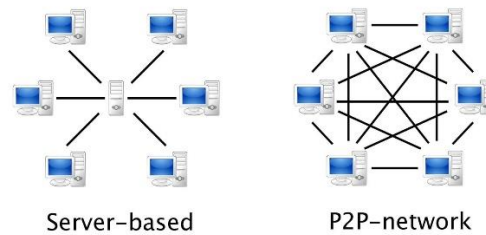


Figure 16: Comparison of P2P network and client-server network

The blockchain each has its local copy of the blockchain. When the node starts working, it scans the network for the longest chain of blocks. If it finds one, it updates the local copy. When a node creates a block, it sends that block to all available peers. Both processes are united by one thing - **validation**.

Network nodes can include both trusted feathers and attackers who promote fake data. To avoid blockchain distortion, each node checks all data received on the network for validity. The validation algorithm goes through each block sequentially and lists all the data, searching for inconsistencies such as transactions created by the non-wallet owner, distorted block data, etc. Such a check can be slow with a large number of blocks. When planning the implementation of the blockchain, you must carefully choose the means of its creation. It was decided to create all the logic in the programming language **Java**. To demonstrate the work and structure of the blockchain, it was decided not to use auxiliary libraries to develop it but to implement it **manually**. Also, the manual implementation provides more flexibility in development and controllability.

The next stage is the selection of auxiliary algorithms and the means of their implementation.

The hashing algorithm **SHA256** was chosen, as it is pretty standard and used in the Bitcoin product, which became the basis for the developed IP. The algorithm can be conveniently implemented with **java.security Java** package. **RSA** is selected as a cryptographic algorithm, primarily because of its timely implementation using the package **java.security**. Its primary analogue was the algorithm **ECDSA**, based on the properties of elliptic curves, but the possibility of its performance via Java is complicated. A tool was chosen to generate random numbers **SecureRandom** from java.security package. When selecting the type of network topology, the main criterion was managing nodes for commercialization. That is why the centralized type was specified.

As a result of this section of this work, a list of technological and software tools needed to create an information system for the goods authenticity based on blockchain technology was formed [41-54]. Each of the selected devices was described in detail, justified its choice on the given technical characteristics based on article analysis [55-67]. Java, GSON, Spring Boot, JavaFX, Maven, HTML, CSS, JavaScript. IntelliJ Idea and Google Chrome have become software tools for debugging and testing programs . A detailed principle of the blockchain operation and its related algorithms was given and a reasonable choice of means of its implementation based on article analysis [68-79].

4. Experiments, results and discussion

4.1. Description of the created software

The first stage of the fourth section is a description of the created software. The report is executed according to the GOST 19.402-78 standard "Description of the program".

4.1.1. General information about the program

The information system for authenticating goods based on blockchain technology consists of two programs: ProductChain and ProductChainServer.

The main program is ProductChain, which is run using the executable file "ProductChain.jar". The program performs the basic logic of the system and consists of three modules:

- **Blockchain:** the module that is the technological root of the whole system. Responsible for all operations related to blockchain logic.
- **GUI:** the module that contains the user interface of a typical manufacturer's workspace. This module is designed only to demonstrate the operation of the system. When transferring the product, each manufacturer implements this module at its discretion.
- **Webserver:** the module contains the logic responsible for the interaction of the client interface with the application. This module is designed only to demonstrate the operation of the system. When transferring the product, each manufacturer implements this module at its discretion.

An additional program for the main program is a server that combines the various processes of the main application. We started using the executable file "ProductChainServer.jar". Program interface language: English. The size of the executable file "ProductChain.jar": 33,640 KB. The size of the executable file "ProductChainServer.jar": 254 KB.

4.1.2. Functional purpose

The implementation of the system is pervasive and consists of various modules. In this regard, we present the essential functions of the program. Blockchain, GUI and Web server modules are organized with the help of unique components - managers, each of which is responsible for its area of functionality. In this way, you can conveniently separate work objects and their behaviour. If desired, you can change this behaviour without additional edits to work objects.

TransactionManager:

- *initiateTransaction*, *processTransaction*, *signTransaction*, and *commitTransaction*: Methods are responsible for the transaction's life and perform operations such as creating a transaction, filling in data, using EDS, and using a transaction.
- *validateTransaction*: checks the transaction for correctness verifies the balance and digital signature of the transaction.

TransactionOutputManager:

- *updateUnspentOutputs* and *removeUnspentOutputs*: methods are responsible for adding and removing selected outputs.
- *getUnspentOutputsByPublicKey*: allows you to get outputs by output recipient.

BlockManager - *createBlock*: is responsible for creating and mining a block according to the provided list of transactions.

WalletManager:

- *createWallet*: method that creates a wallet with a pair of keys.
- *sendFunds*: using the provided wallet, sends its funds to another wallet.
- *getBalance*: calculates the wallet balance by calculating unused outputs.

BlockchainManager:

- *init*: is responsible for the program initialization process. Performs all start-up actions such as creating a base block, base transaction, base wallet, etc.
- *requestFromCoinBase*: requests funds to the base wallet.
- *addTransaction*: Adds a transaction to the transaction pool.
- *flush*: uses available transactions from the pool to form a block and add it to the blockchain.

ProductManager:

- *createProductType* and *createProductUnit*: methods create product types and product units.
- *applyProductUnit*: adds a unit of goods to the blockchain by sending it one balance.
- *consumeProductUnit*: consumes a unit of goods by returning 1 balance to the product type.

- *isProductUnitConsumed*: checks whether a unit of goods is consumed according to the following algorithm. If the balance of a unit of goods = 0, then the unit is consumed.

Miner:

- *mineBlock*: carries out mining of the block according to the set complexity
- *checkBlock*: checks whether the block has been mined correctly

ChainValidator - *isChainValid*: rebuilds the blockchain from scratch and checks each block for inconsistencies.

Sha256HashGenerator - *doDataHashing*: hashes the received tape according to the SHA256 algorithm.

MerkleTreeCreator - *createMerkleTree*: builds a hash tree based on the received list of transaction hashes.

RSAKeyGenerator:

- *generateKeys*: generates random keys using the RSA cryptographic algorithm and the SecureRandom random number generation algorithm.
- *restoreKeys*: restores keys from received byte arrays.

RSASignatureManager:

- *applySignature*: creates an EDS based on the received data and a private key.
- *verifySignature*: checks whether the EDS was created by the owner of the received public key.

RSAEncryptionManager:

- *encrypt*: encrypts data using a public key.
- *decrypt*: decrypts data using a private key.

Util - class contains methods of conversion and comparison of keys

BlockchainToFXTreeConverter - *convert*: converts a blockchain as a JSON object into a tree interface.

JavaFXBlockchainInteractionController- this class contains methods that are listeners of blockchain events. The class serves as a bridge between the network part and the interface part.

PeerClient:

- *sendMethodRequest*: sends a request using a socket connection.
- *loadBlockchain*: downloads the largest blockchain from a P2P network.
- *sendBlock*: sends the block to all P2P nodes of the network.
- *registerPeer*: registers the node on the P2P server.
- *loadPeerList*: gets a list of nodes in the P2P server.

PeerServer – class that is responsible for the P2P server of the node. The server runs in the background and accepts all requests from other hosts.

PeerWorkerThread - class of the P2P server, responsible for receiving requests from other nodes.

Methods that implement the behaviour of a specific method of the P2P server node implement the interface **PeerServerMethodHandler** and abstract class **AbstractPeerServerMethodHandler**:

- *GetBlockchainMethodHandler* – class that sends a blockchain on request.
- *GetBlockchainSizeMethodHandler* – class that provides the size of the current version of the blockchain on request.
- *SendBlockMethodHandler* – class that sends a new block after its successful mining.

MainController:

- *showProduct*: The HTTP server method returns an HTML page of public product information.
- *consumeProduct*: The HTTP server method returns an HTML page of private product information and consumes the product in a blockchain. The application.properties file is responsible for configuring all processes and methods. This file contains the parameters of various modules, which allows you to change the program's behaviour quickly.

4.1.3. Description of the logical structure of the system

To demonstrate the operation of the blockchain, these software applications are run manually.

The ProductChain server program is organized as an application that is controlled in a dialogue mode with a user interface. When you start the program, a download window is displayed, which corresponds

to the initialization of the program in the P2P network. After successful download of the blockchain, the user gets to the main menu of the program. This menu consists of 4 options: Products, Transactions, Blockchain, Exit. The first three buttons lead to the launch of the corresponding usage options:

- **Products:** Opens a scene for viewing and creating product types. This scene is organized in the form of a table. Below the table, possible scene options: go back, create product type, available product units. Create product type opens a modal window containing fields to fill in a new product type. Functional units of goods lead to the opening of a similar tabular scene and units of goods.
- **Transactions:** Opens the transaction view scene. The scene is organized in the form of a table. Below the table is the scene options: go back, update transactions, create and perform block mining. The block creation option opens a modal window that requires confirmation of the start of the operation. At the time of mining, the window displays the operation screen. After successfully or unsuccessfully adding a block, the window shows the status of the operation and the key to close the window.
- **Blockchain:** Opens the blockchain viewing scene. The scene is organized in the form of a tree list that can be expanded and collapsed. By default, this list is collapsed, and the blocks in it are sorted in descending order.

The Back button in all scenes plays the previous location in the background opening history.

The "Exit" key is responsible for closing the program.

The server part of the application on HTTP requests returns HTML pages that cannot interact. The browser opens these pages and is divided into 2 main blocks: the title and the information part.

ProductChainServer is a command-line dialogue program. When the server starts, the command line displays the status of its execution. To close the server, use the key combination Ctrl + C or the command key to close the command line.

4.1.4. Used technical means

The program is designed to run on devices such as a personal computer. The program is controlled in the mode of dialogue with the operator. Devices are used for control: monitor, keyboard, manipulator type "Mouse". The program is stored on the hard disk. In working condition, the data is stored in RAM. A network adapter is used for communications. Client requests are executed on both stationary devices (PCs) and mobile devices. PC requirements:

- Windows 7/8/10 operating system;
- Java software version 11 and higher installed;
- free hard disk space 64 MB;
- free 4 GB of RAM;
- internet access with a bandwidth of 10 Mbps.

Mobile device requirements:

- Chrome / Opera / Firefox / Safari Internet browser or other browsers that supports HTTP communication and ECMAScript6 standard is installed;
- Internet access with a bandwidth of 1 Mbps.

4.1.5. Call and download

Both ProductChain and ProductChainServer come in a packaged archive. After unpacking, the programs are launched using the command line and the necessary parameters. The ProductChain program is called with the following command: `java [command line parameters in the form parameter_name=parameter value] -jar (relative path to the executable file) /ProductChain.jar`.

The ProductChainServer program is called with the following command: `java -cp (relative path to the executable file) /ProductChain.jar com.oprokipchuk.ProductChainServer.Main (command line parameters as parameter_name=parameter value)`

The browser page is opened by clicking on the appropriate links.

4.1.6. Incoming data

The input data for the developed software applications are the command line input parameters. ProductChain startup options are listed in Table 12.

Table 12
ProductChain command-line options

Parameter	Type	Default value
-DWEB_SERVER_PORT	Number within [0, 65535]	8080
-DSERVER_IP	Xxxx format string	127.0.0.1
-DSERVER_PORT	Number within [0, 65535]	8001
-DCLIENT_IP	Xxxx format string	127.0.0.1
-DCLIENT_PORT	Number within [0, 65535]	8101

Consider the purpose of each of the parameters in more detail:

- **WEB_SERVER_PORT.** The parameter is responsible for the port on which the application's HTTP server is deployed. The specified port must be free, i.e. not occupied by another program. To run multiple program processes on the same device, you must use different values for this parameter.
- **SERVER_IP.** The parameter must specify a valid IP address for the P2P server.
- **SERVER_PORT.** The parameter must specify a valid P2P server port.
- **CLIENT_IP.** The parameter is responsible for the IP on which the P2P node is deployed. The specified IP address is used when registering a host on a P2P server.
- **CLIENT_PORT.** The parameter is responsible for the port on which the P2P node is deployed. The port must be accessible. Different values are specified to start multiple processes.

ProductChainServer contains one parameter called port (default value 8001), responsible for the port on which the P2P server is deployed. The parameters port and SERVER_PORT must match for the system to work correctly. In the actual application, the standard values of IP parameters need to be changed because they indicate the local address of the device (localhost).

4.1.7. Output data

The output of the developed programs is made in the form of a visual display in the appropriate interfaces. For the ProductChain program, the results are the current status of the blockchain, which can be viewed in the blockchain scene of the user interface and information about the product and the product position sent in response to an HTTP request.

For ProductChainServer, the result is a query history that is displayed on the command line.

4.2. User manual

4.2.1. Introduction

The information system is designed to establish a relationship of trust between the manufacturer of the product and the buyer based on verifying this product authenticity.

Among the developed software are the manufacturer's tools with which he creates products. When creating a product, unique identifiers are generated for it in the form of public and private keys and the corresponding links to the manufacturer's web server, which processes HTTP requests for the purchased product. The manufacturer uses the received links to place two QR-codes.

The public QR-code is placed on the product packaging. When scanning such a code, the buyer receives a link through which he receives general information about the product, checks the manufacturer's resource and finds out whether this product has already been used or not.

The personal QR code is placed inside the product packaging. A link to such a code leads to the product activation page, allowing you to get private information about the product.

4.2.2. General information about the program

The developed software package consists of two applications: "ProductChain" and "ProductChainServer". The program consists of the leading blockchain module, which provides constant functionality and auxiliary modules, which are designed to demonstrate typical uses of the system. Implementations of additional modules differ depending on the manufacturer. The server part of the program is created using java, spring boot, maven, gson and JavaFX. The website was created using HTML, CSS, javascript. Any modern browser is enough to run a website correctly.

4.2.3. Classes of solved tasks

ProductChain is designed to provide product management tools through a dialogue with the manufacturer's user interface, which allows you to perform the following actions:

- Create / view products and product types.
- Add created products to the blockchain.
- Send the created blocks to P2P network nodes.
- Receive data from P2P network nodes.
- View current blockchain content.

The server part of the application allows accepting HTTP requests. It is responsible for two actions: access to public product information (response to a shared link request) and access to private product information and product activation (response to a personal link request).

ProductChainServer is a P2P server that integrates client applications into a P2P network. This program keeps a register of all system nodes and provides information about all registered nodes.

4.2.4. Description of the main characteristics and features of the program

To use the manufacturer's client application, you must have a personal computer running Windows and a stable connection and high bandwidth. Like any other interactive program, you must have a basic set of devices for interaction (keyboard, mouse, monitor). To download and send a large amount of data requires a good Internet, it directly affects the program's quality. It is also essential to have the mighty computing power of the PC because the blockchain involves many resource-intensive tasks. First of all, the mining process should take place as soon as possible because whoever creates the block first, the block will be valid. All other nodes must cancel mining and start again.

Each vendor has its implementation of non-core modules, so the availability of some features directly depends on the availability of the vendor's server.

To use the system on the buyer's part, access to any of the modern browsers is enough.

4.2.5. Information about functional limitations on the application

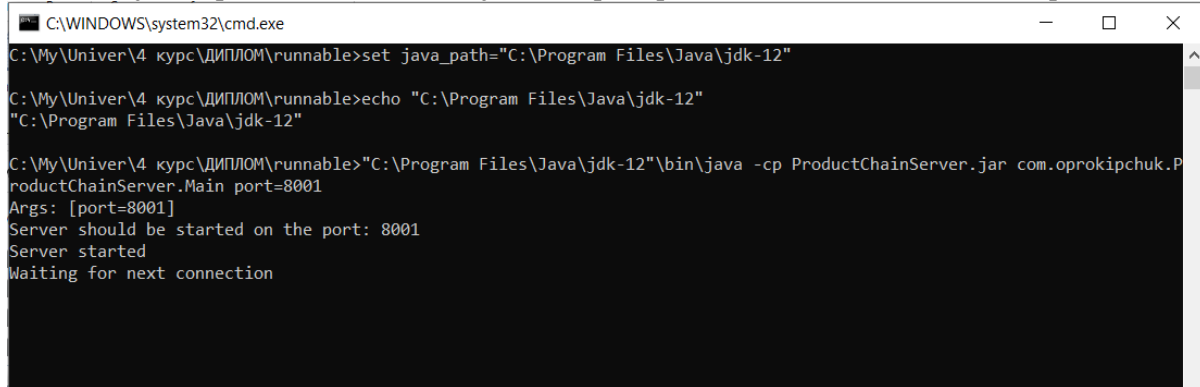
The main indirect limitation on the application is the bandwidth of the Internet and the device's computing resources. Although minimal resources are sufficient for local use, it will be impossible to successfully use the application without powerful hardware in the conditions of a real application with a thousand nodes. When using the program, there is a limit on creating private data for the product in 117 characters. It is due to the peculiarity of the encryption algorithm based on RSA keys. The limit is increased by using larger keys.

To maintain the manufacturer's server and the P2P server, you must have a dedicated IP address service from the provider. As you know, any Internet device can send requests, but only devices with a dedicated IP can receive requests. Another restriction on the maintenance of such servers is in the ports used. Such ports must be "open", i.e. the connection to them should not be blocked either by the PC and OS's internal security systems or by external Internet routing systems and the provider.

4.3. Analysis of the control example

To demonstrate the operation of a comprehensive system for the authenticity of goods based on blockchain technology, we give a control example of use. This example consists of two parts: the actions performed by the manufacturer when creating and adding goods to the blockchain and the actions performed by the buyer to verify the authenticity of the goods.

To start, you need to start a P2P server. Run the server on the command line with the following command: `java -cp ProductChainServer.jar com.oprokipchuk.ProductChainServer.Main port = 8001`.



```
C:\WINDOWS\system32\cmd.exe
C:\My\Univer\4 курс\ДИПЛОМ\runnable>set java_path="C:\Program Files\Java\jdk-12"
C:\My\Univer\4 курс\ДИПЛОМ\runnable>echo "C:\Program Files\Java\jdk-12"
"C:\Program Files\Java\jdk-12"
C:\My\Univer\4 курс\ДИПЛОМ\runnable>"C:\Program Files\Java\jdk-12"\bin\java -cp ProductChainServer.jar com.oprokipchuk.P
productChainServer.Main port=8001
Args: [port=8001]
Server should be started on the port: 8001
Server started
Waiting for next connection
```

Figure 17: P2P server running

Caption «Waiting for next connection "signals that the server is ready to start. You can then run the manufacturer's client application. For the first process, this was done with the following command: `java -DSERVER_PORT = 8001 -DCLIENT_PORT = 8101 -DWEB_SERVER_PORT = 8081 -jar ProductChain.jar`. When you start the application, the download screen appears, which can take a long time depending on the speed of the Internet and the device's capacity.

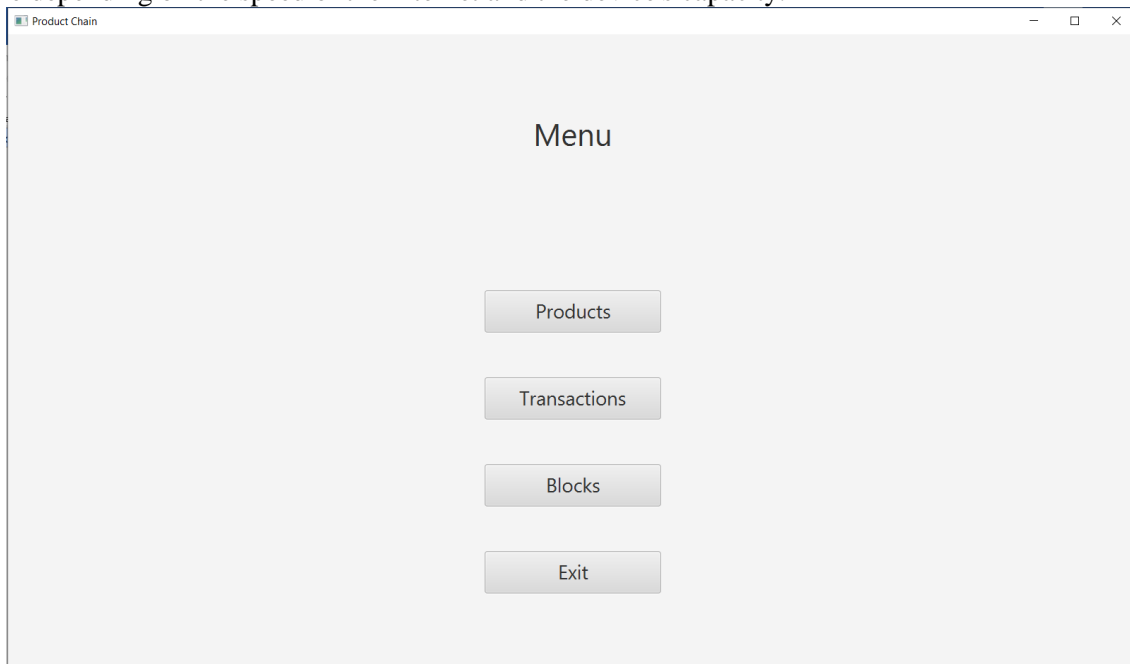


Figure 18: ProductChain main menu

The first step in using the program is to go to the products menu "Products". After the transition, you can see an empty table of product types and possible options.

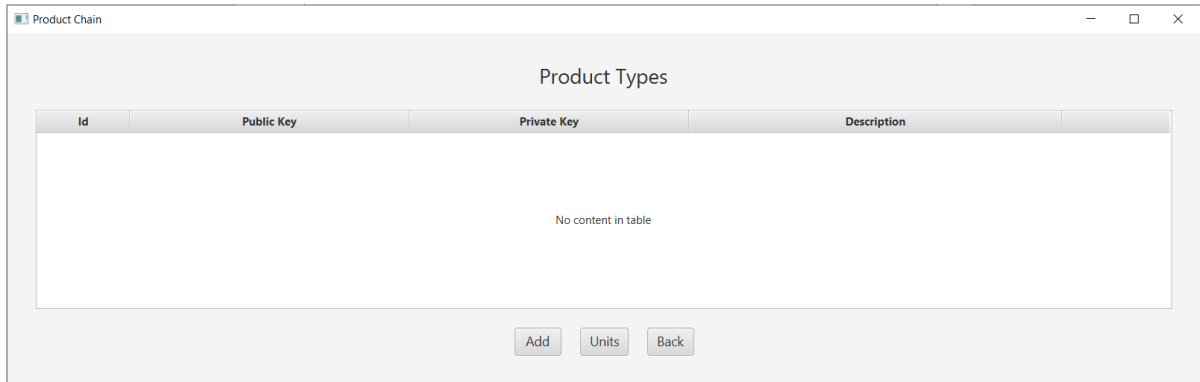


Figure 19: Scene of product types

To create a new product type, press the "Units" key. After clicking, a modal window opens with the ability to enter new product data. A feature of this window is the ability to record the product type's keys manually and automatic generation using the "Generate Keys".

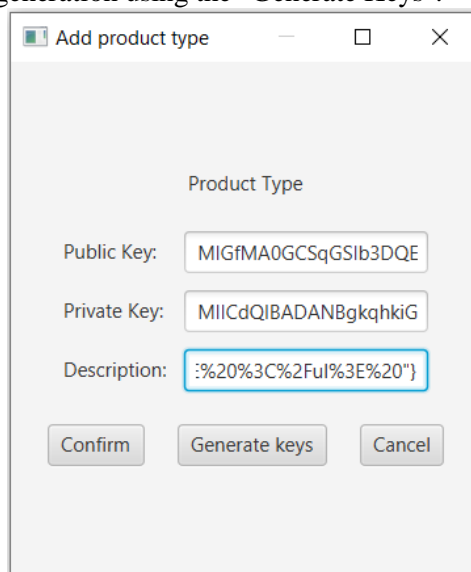


Figure 20: Product type creation window

As a result of creating several types of products, the window for viewing these types will look like this.

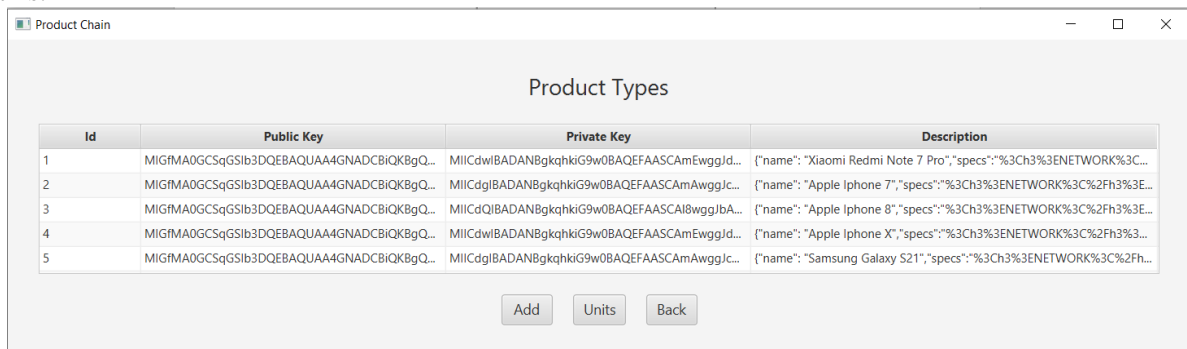


Figure 21: Scene of product types

At this stage, you need to select a specific product type and press the "Units "to go to particular units of goods.

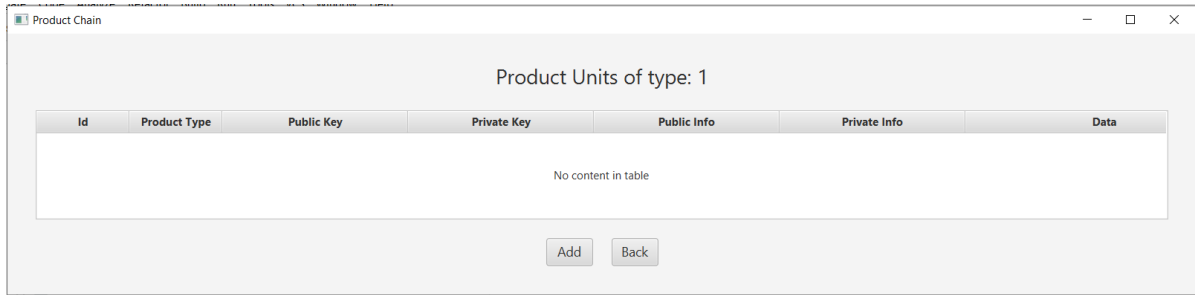


Figure 22: Scene of product units

Every physical commodity is a unit of product. Although different units may have one type of product, each product unit has unique identifiers to identify the required product. We create several units of goods using the "Add" key.

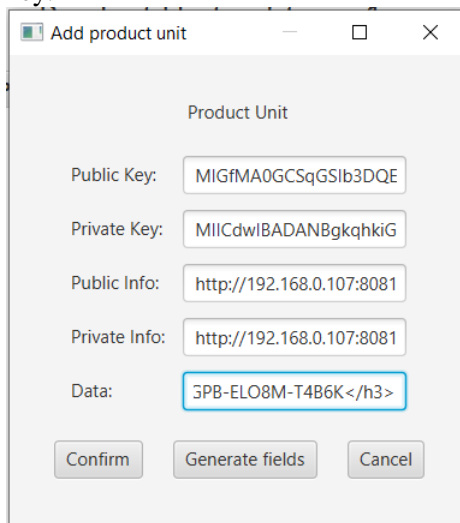


Figure 23: Product unit creation window

There is a specific key for automatic field generation for the product type creation window and for the product unit creation window. In this case, the field "Public Info and Private Info must be saved. They will be needed later.

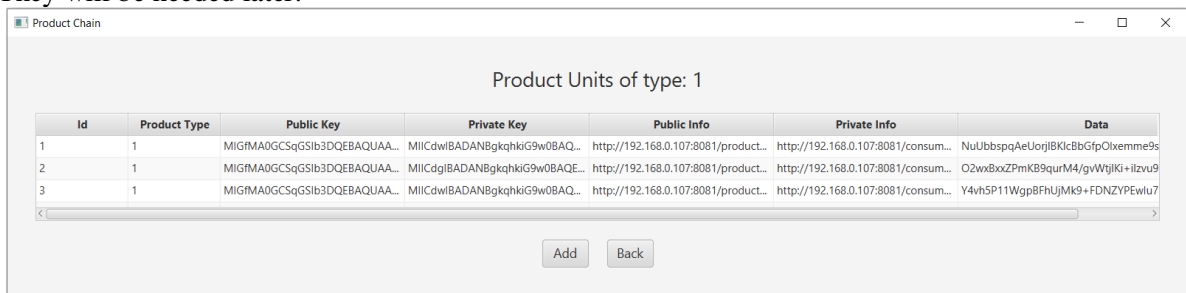


Figure 24: Scene of product units

It is what the product unit scene looks like after creating multiple instances. The next step is to add the created data to the blockchain. To do this, return to the main menu and open the transaction scene using the "Transactions". Only an empty transaction table will be displayed when you open this scene because transactions have not yet been created. To automatically make the necessary transactions, you need to press the "Refresh" key. In Fig. 25 you can see 4 automatically generated transactions. Three of them correspond to the three created units of goods and have a balance of 1, and the last transaction, with a balance of 1000, corresponds to the transfer of funds from the base wallet to a specific type of goods. It is necessary so that the type of product can be created as a unit. After such a transfer, it is possible to build 1000 units of goods of one type.

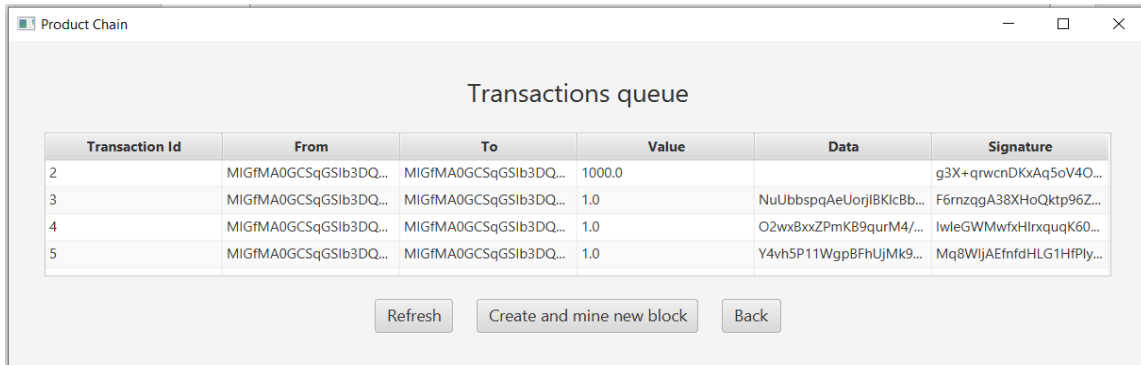


Figure 25: Transaction viewing scene

To create a block based on these transactions and add it to the blockchain, you need to press the "Create and our new block" key, which leads to the appearance of the modal window "Create Block".

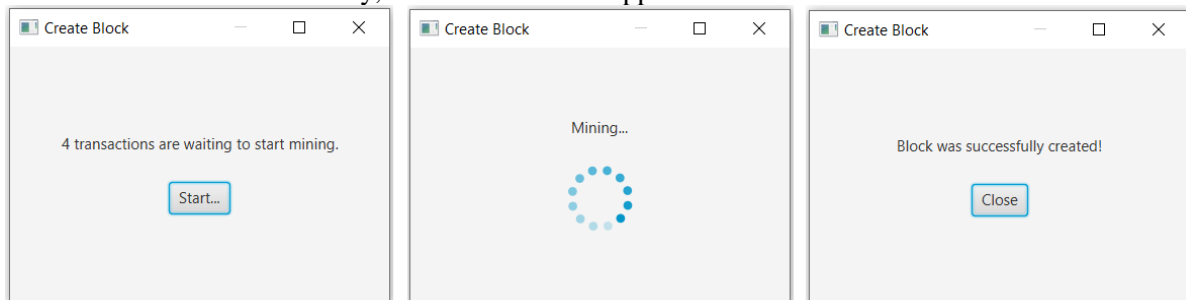


Figure 26: Stages of block creation

Creating a block is visually divided into 3 stages:

- Notification of how many transactions will be added to the created block. This value depends on the number of transactions made and the maximum possible number of transactions in the block. In this example, the maximum number of transactions is set to 4, so all transactions fall into the block. If there are more transactions, it is necessary to repeat the creation of the block until the full application of all transactions.
- The mining stage is the longest stage, depending on the established complexity.
- Block creation status can be both successful and unsuccessful.

After creating a block, you can go to the main menu and the blockchain scene by pressing the "Blocks" key.

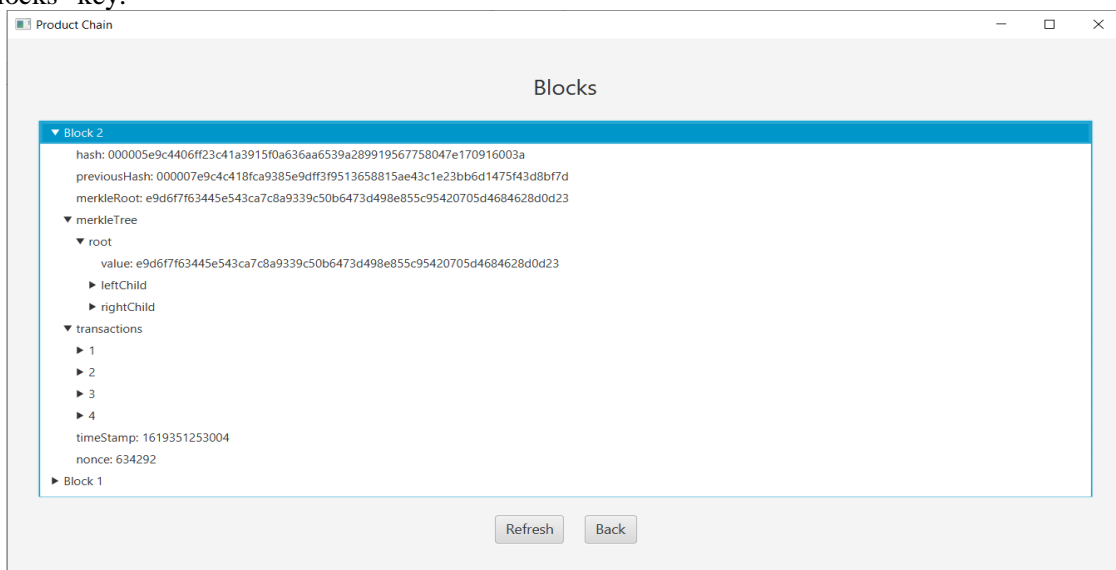


Figure 27: Block viewing scene

You can view all the blocks in this scene, and the values are written to the blockchain. By default, this list is organized into a tree structure, sorted by blocks in descending order, and all its values are collapsed. With the "Refresh" key, you can update the list of blocks to the current one.

Now the manufacturer needs to generate QR-codes using any generator based on the saved fields "Public info" and "Private Info" and place them on the product packaging.



Figure 28: Generated QR-codes

Then the control example is continued by the buyer of the goods, and the first stage of its application is the scanning of the public code. Upon receipt of the product before unpacking, the buyer can scan the external QR code and find out whether this product has already been used and whether this code leads to the correct manufacturer. In Figure 4.13. shows a view of such a page from a mobile device. From the received information, it is possible to establish the correct site of the manufacturer, characteristics of the actual goods coincide with the information on a site, and also the status of the goods - is not used. Therefore it is possible to conclude that the goods are authentic.

Otherwise, counterfeit goods are established, and the buyer can either refuse to buy the goods or apply to the relevant authorities to protect the rights of buyers.

Because this product is authentic, the buyer opens the package and scans the internal OR code.

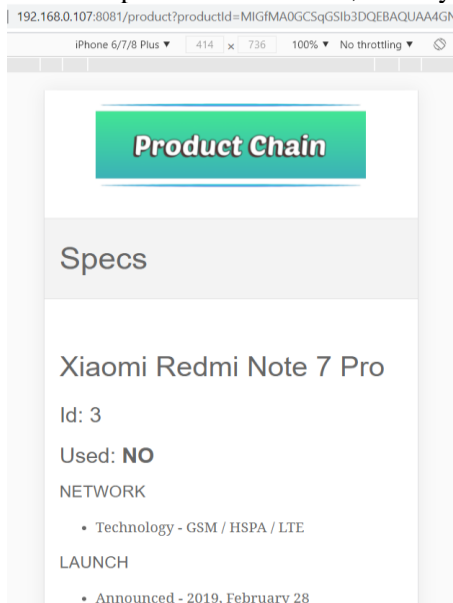


Figure 29: The transition to Public Info

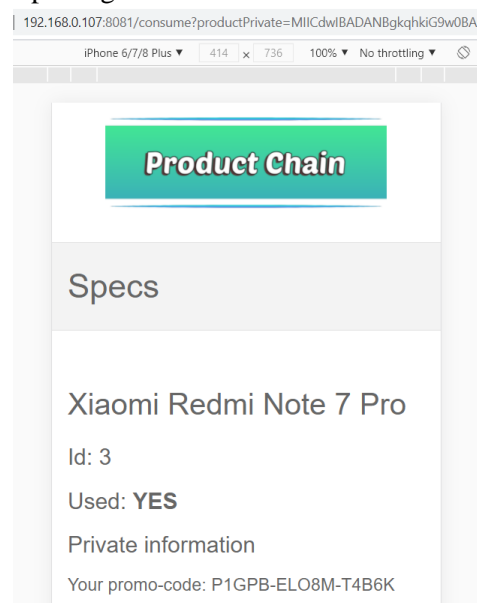


Figure 30: The transition to Private Info

As a result of scanning the internal QR-code, the buyer receives information related to a specific product, such as activation codes, promo codes, etc. Here comes the second factor of authentication: the buyer checks whether the product id from the public and private QR-codes match and scans the public QR-code again, resulting from which the status of the product changes to "used".

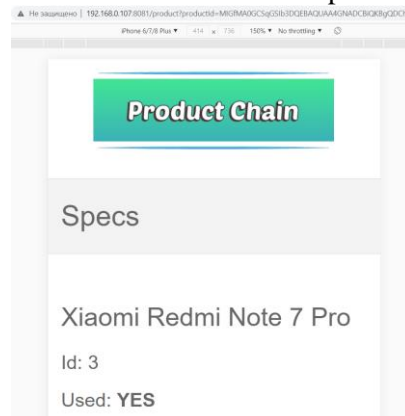


Figure 31: The site as a result of re-transition to Private Info

After re-scanning the external QR code, the buyer observes the fact that the product is now used. At the moment, the authentication is completed, and the buyer begins to use the received goods.

Next, the control example is continued by the manufacturer. The Product Chain application creates a consumption transaction at the private link, which can be seen in Fig. 13, which must also be added to the block and then to the blockchain.

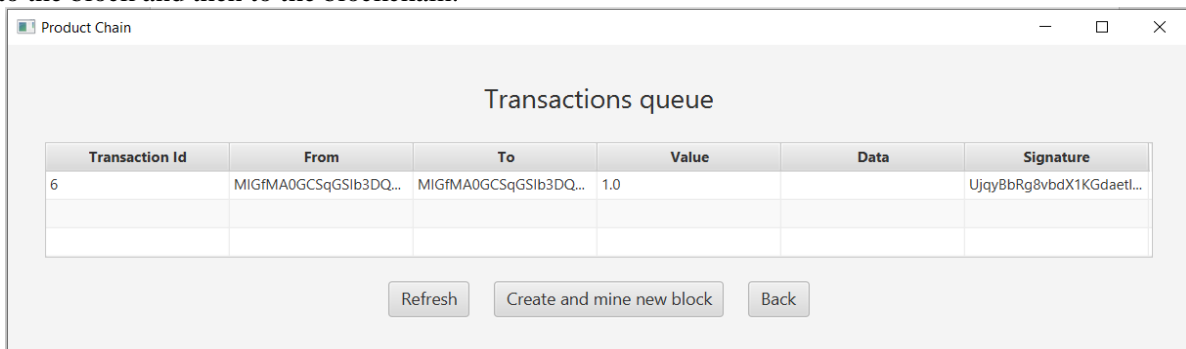


Figure 32: Transaction of consumption of goods

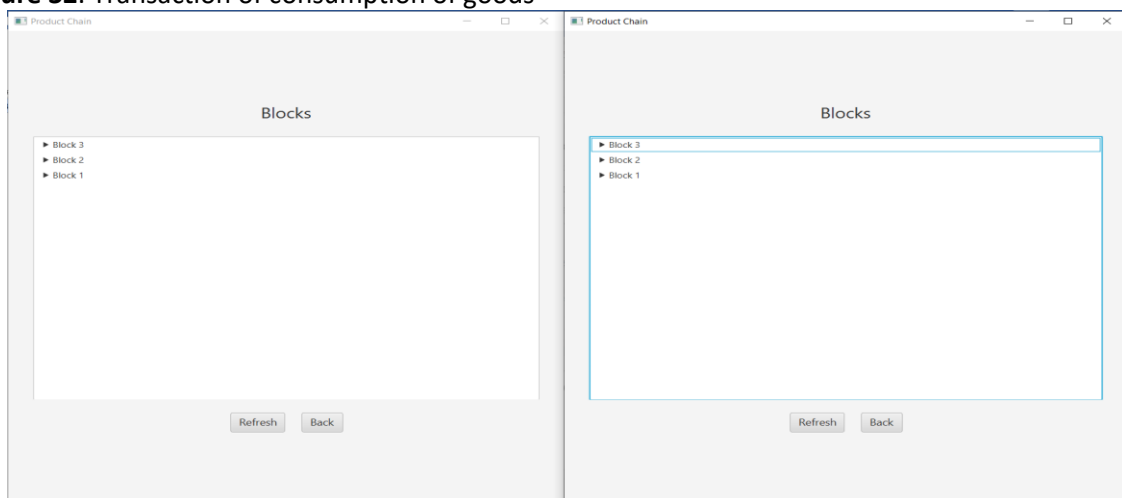


Figure 33: 2 Product Chain application processes are running simultaneously

Because the application is connected to a P2P server, it can send and receive data from other nodes.

Let's start another process of the client application of the manufacturer working on other ports using the command: `java -DSERVER_PORT = 8001 -DCLIENT_PORT = 8102 -DWEB_SERVER_PORT`

= 8082 -jar ProductChain.jar. The new application will no longer have an open blockchain but will load an existing blockchain from the first node. Both processes contain the same number of blocks. From this point on, after each new block mining, this block is distributed throughout the blockchain network so that all its nodes have the most current state. However, the same feature is problematic because if a node gets a new block in the mining process, it will have to cancel mining and start creating it again.

To demonstrate this phenomenon, run mining simultaneously on two processes, as shown in Fig. 34.

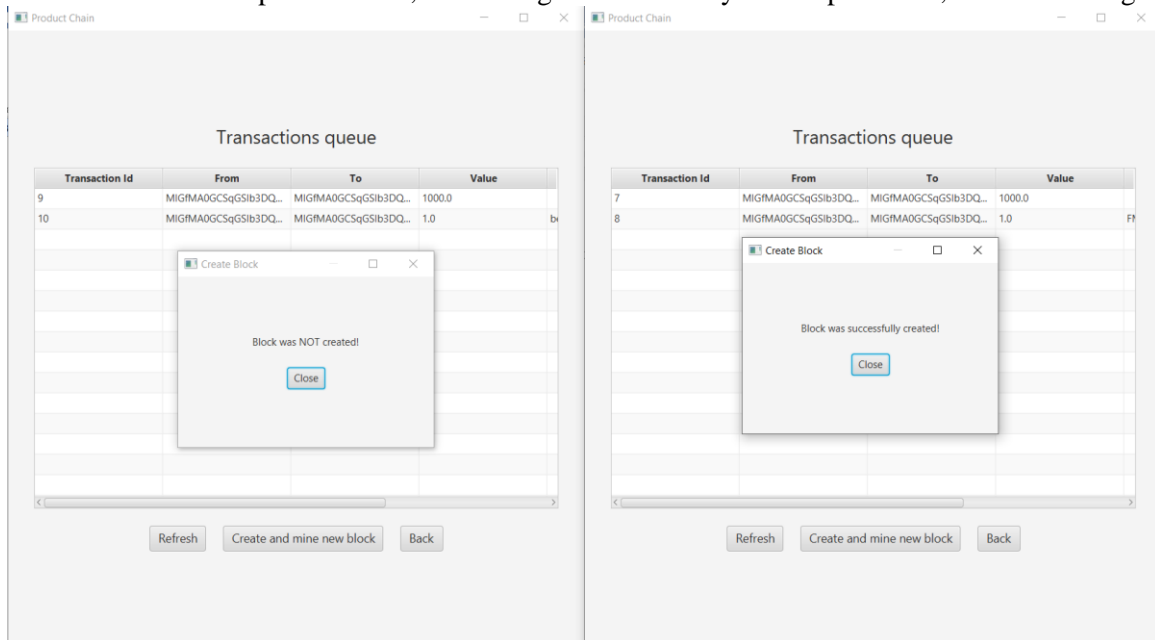


Figure 34: The result of simultaneous mining of blocks of two processes

According to the conclusions of each of the blocks, the mining of the first process was unsuccessful, and the second - successful. In other words, the first process received a new block during mining and stopped it to attach the block to the local copy of the blockchain and convert the transactions.

4.4. Research the behaviour of the program

To optimize further work with the program and the search for opportunities to increase returns by changing the working environment, we study the system's performance depending on the adjustment of specific parameters. The first such parameter is the complexity of mining. This parameter specifies the number of zeros at the beginning of the hash to assume that the job is confirmed. So we investigate the range of complexity from 1 to 7 (on complexity 8 and more mining on the working device takes too long). The results of the analysis are shown in Fig. 35.

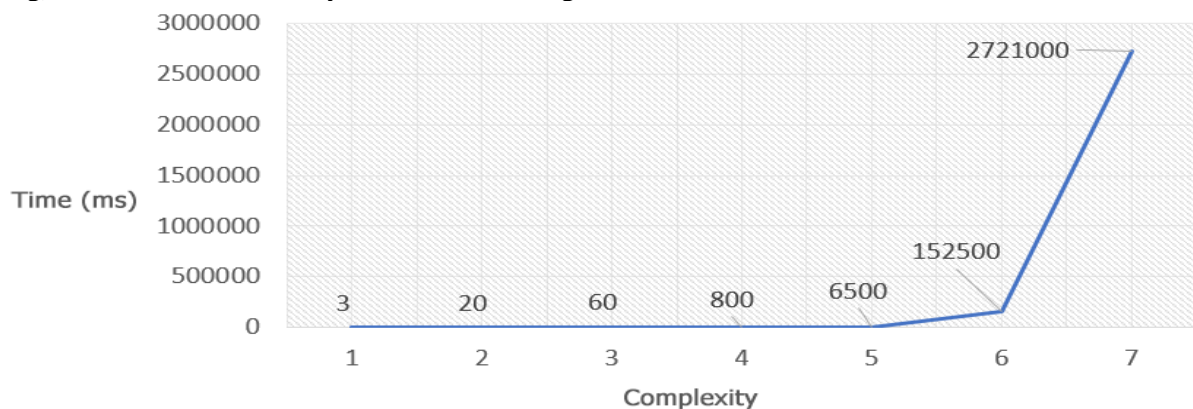


Figure 35: Graph of the dependence of mining time on its complexity

Time is given in milliseconds. From this graph, it is seen that with each subsequent complexity increases the total time several times. It is due to a significant increase in the number of operations that must be performed during mining. Of course, mining also depends on the capacity of the hardware. Real devices that are engaged in mining regularly are a set of advanced processors and video cards.

The dependence of mining time on the processor frequency is shown in Figure 4.19.

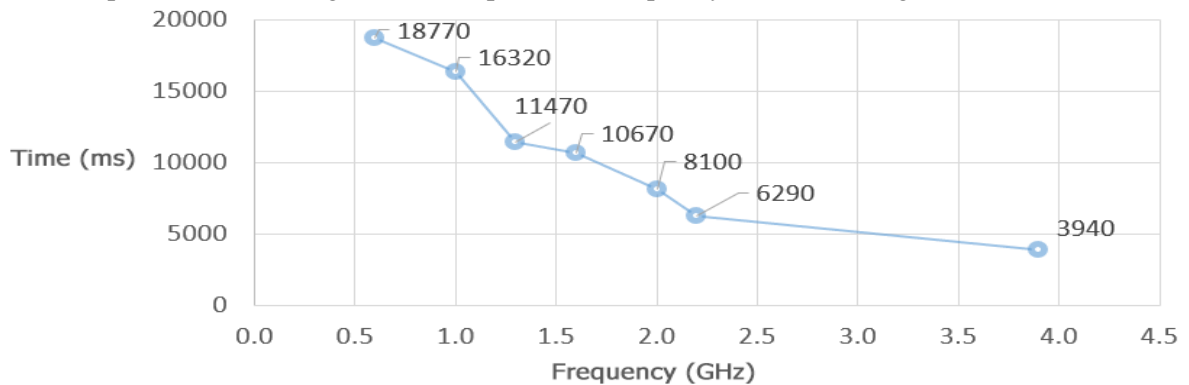


Figure 36: Graph of mining time dependence on CPU frequency

The analysis was performed for the complexity of mining - 5. From the obtained data, the additional power of the processor can significantly reduce mining time. The next subject of research is the creation of a block. The main resource load this stage is the calculation of hash functions for transactions, for the block, and the construction of a hash tree. The results of the study are shown in Fig. 37.

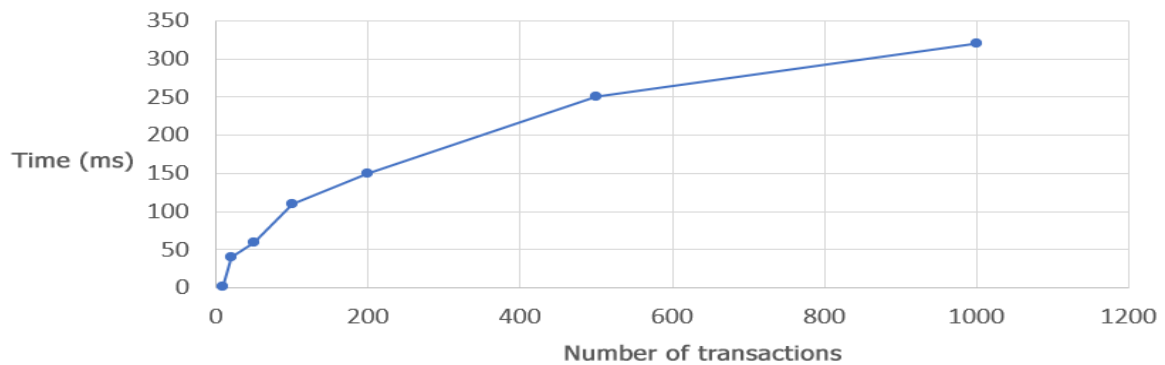


Figure 37: Graph of the dependence of the block creation time on the number of transactions

From the obtained data, the number of transactions of the unit is its primary resource load. It should be taken into account when choosing the maximum number of transactions per block.

The latest study is to identify the dependence of time validation of the blockchain on the number of blocks. The results of this study are presented in Fig. 38.

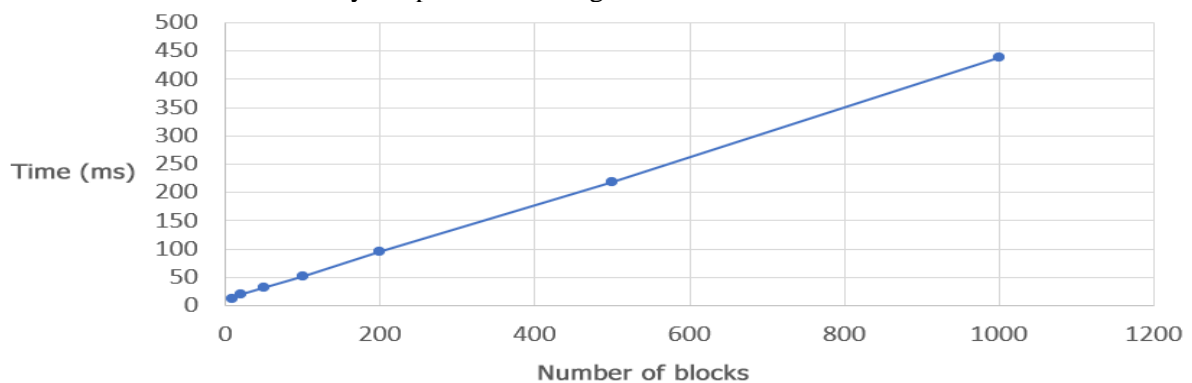


Figure 38: Graph of blockchain validation time depending on the number of blocks

On the received graph, it is possible to observe linear dependence of the time of validation of a blockchain on the quantity of blocks. It signals that the validation process is not overloaded with resource-dependent operations. When expanding the blockchain over time, you can count on the expected duration of the blockchain's validation according to the linear relationship.

As a result of the implementation of this section, the description of the created software according to the GOST 19.402-78 standard "Description of the program" was carried out. According to the standard, information about the software, functions, logical structure, inputs and outputs was prepared.

A user manual was also generated, which provides information on the use of the system, features and various limitations. An analysis of the control example was given, which analyzed the sample of use by both the manufacturer of goods and the buyer.

Finally, the behaviour of the developed software was analysed depending on the change of mining complexity parameters and the maximum number of transactions in the application.properties file block. In addition, the relationship between CPU frequency and mining time and the relationship between blockchain validation time and the number of blocks was determined. Based on the kept data, it is possible to make further optimization and system configuration decisions.

The topic of this work is an information system for the authenticity of goods based on blockchain technology. Given the current prevalence of commodity-market relations and the damage caused to them by the counterfeit market, the issue of protection against counterfeiting is becoming increasingly important daily. Every year, the share of counterfeit goods in the market only grows, as well as the damage it causes. And this is not only material damage but also damage to the environment and human health. In the worst cases, such carelessness can lead to death. That is why large companies today are beginning to invent a variety of solutions to combat counterfeiting. The developed information system is designed to help buyers of goods identify the original product by placing on the product two scanning identification codes that lead to information in the secure data structure of the blockchain. This system is economically feasible for companies producing goods because it can significantly reduce the material damage caused by manufacturers of counterfeit goods.

Positive perceptions of this software product by potential consumers of goods are also expected, as the system will allow users to have exactly what they want - not to come across counterfeits and save money. An additional advantage will be the use of today's popular technology - blockchain.

The fight against counterfeiting is actively going on every day, and many ways have already been developed to identify authentic goods. The most popular are: legislative, the method of developing recommendations, the method of complicating copying, the method of monitoring suspicious ads and the method of keeping records of each unit of goods. The developed system belongs to the last category. The market for the use of anti-counterfeiting methods is relatively developed, as most companies already use at least one of these methods. Among the analogues of the system, we can highlight the product Microsoft Aura Ledger, which also uses blockchain technology and comes in the form of software for manufacturers of goods. The advantage of the developed product is that it unites many manufacturers into a single network to ensure more excellent reliability and transparency of data and two-stage identification, which gives greater accuracy. The primary consumers are producers of goods that suffer material damage from counterfeiters. Such manufacturers want to minimize the impact of counterfeit products on their profits and maintain the reputation and trust of their customers.

Competitors are other vendors of anti-counterfeiting software at the same level of protection. Levels of security in a separate production can be combined. According to existing examples, states have a positive attitude to combating counterfeiting and are ready to cooperate.

Given the general state of counterfeiting, as well as the low prevalence of such methods in Ukraine, we can conclude that it is advisable to develop a strategy of counterfeiting for the Ukrainian market, which is an improvement of a similar software product (Microsoft Aura Ledger), which has already gained some popularity abroad. Consumers of the product will be manufacturers of goods, which, when purchased, will receive a set of necessary software and access to a standard blockchain network.

Based on the above data, it is developing a new product with related services was chosen for implementation. As a result of the market analysis, a rather painful problem of counterfeiting in the goods market was found, so the developed system of the authenticity of goods based on blockchain technology should receive a positive response from consumers. The use of the latest technologies, such as blockchain, allows to achieve a very high level of protection of information of manufacturers of goods and has a positive effect on the desire of buyers to use this system of protection. Many indirect

competitors offer protection against counterfeiting at other levels, but they are not a substitute for the developed product but complement a comprehensive protection system. The product's analogue is the Microsoft Aura Ledger counterfeit protection system, which is also based on a blockchain. The advantages of the presented product are that it is cheaper to implement and uses two-stage identification, which will allow you to compete for a large share of consumers. The product also provides software adaptation services for each consumer and consolidation consumers into a single P2P network.

As a result of this section, a detailed description of the information system for verifying the authenticity of goods based on blockchain technology in terms of its economic feasibility was complex. Potential consumers were identified as producers of goods that focus on the medium and low cost of implementation. Direct and indirect competitors were analyzed, the main one being Microsoft Aura Ledger. According to the listed advantages of the presented product, it has the potential to obtain the required market share. In addition, a choice was made as to the strategic product alternatives of both groups. The strategy of developing a new product with related services was chosen from the first group. The product development strategy was selected from the second group.

5. Conclusions

As a result, information system for verifying the authenticity of goods based on blockchain technology were developed. All the necessary modules have been developed to fully demonstrate the system's operation, including blockchain modules, the manufacturer's user interface and a web server. Also implemented P2P server, which allows you to interact with developed software applications. The issues were studied, system analysis and selection of technical means of implementation were carried out. After that, the system was created, and its behaviour was investigated. In addition, an analysis of the economic feasibility of the system was conducted. The article was carried out thanks to the current condition of counterfeit goods for today, and scales of losses exceeding 300 annually was received. Billions of dollars, not taking into account the damage to the health of buyers. At the end of the section, a detailed analysis of existing methods of combating counterfeiting from legislative to technological was made. The main advantages and disadvantages of each of these methods were identified. Thanks to this study, the main competitor of the developed system - Microsoft Aura Ledger - was identified. Based on the existing competitor, the advantages of the developed system in greater transparency, reliability and low cost of implementation have been established. This paper began with the construction of the system implementation goals, as a result of which the most critical development criteria were identified: persuasiveness, competitiveness, flexibility, stability, accuracy and security. Based on these criteria, the type of system was chosen: information and reference system. After that, the section continued with the construction of process diagrams and their decomposition in IDEF0 notation. The area ends with constructing a diagram of the hierarchy of tasks based on the obtained processes.

The result of the article was the choice of means of implementing the system. First of all, the analysis was given, and the choice of technological means of development was substantiated. They were the means of stationary development: java, maven, gson, spring boot and JavaFX. To implement network communications, it was decided to use the tools of socket-server development of the java programming language. It was decided to use HTML, CSS and JavaScript to develop the Web part. The development environment between IntelliJ Idea and Eclipse was then chosen in favour of the first tool. The last stage was the choice of blockchain implementation technologies and auxiliary algorithms. For this purpose, standard java programming tools were chosen and RSA, SHA256 algorithms, and a network type - a decentralized P2P network with a navigation server.

The paper focuses on the implementation of the system and what is associated with it. Therefore, this article began with a detailed description of the implemented system, covering aspects such as general information, logical and functional structures, technical requirements, inputs, and system outputs. The user's instruction with the description of principles and nuances of application was given. After that, the analysis of the control example was carried out both by buyers of goods and by manufacturers. At the end of the section, a statistical study of the program's behaviour was performed for further optimization based on dependencies such as time, mining complexity, block size, and processor frequency. The main conclusions of such studies are the need to improve mining algorithms further and increase the capacity of the device used.

6. References

- [1] O. Kuzmin, M. Bublyk, Economic evaluation and government regulation of technogenic (man-made) damage in the national economy, in: Computer sciences and information technologies (CSIT), 2016, pp. 37-39.
- [2] Y. Matseliukh, V. Vysotska, M. Bublyk, T. Kopach, O. Korolenko, Network Modelling of Resource Consumption Intensities in Human Capital Management in Digital Business Enterprises by the Critical Path Method, volume 2851 of CEUR Workshop Proceedings, 2021, pp. 366-380.
- [3] Allison. Behind the industry of counterfeit products in China and lawsuit success cases, 2021. URL: <https://daxueconsulting.com/counterfeit-products-in-china/>.
- [4] F. Richter, U.S. Companies Most Affected by Counterfeiting, 2019. URL: <https://www.statista.com/chart/17407/countries-most-affected-by-counterfeit-and-pirated-goods/>.
- [5] F. Richter, The Industries Most Affected by Counterfeit Products, 2019. URL: <https://www.statista.com/chart/17410/counterfeit-and-pirated-products-by-category/>.
- [6] C. Thompson, Buyer beware: That low-priced, high-end makeup could be counterfeit, and toxic, 2018. URL: <https://komonews.com/news/consumer/buyer-beware-that-low-priced-high-end-makeup-could-be-counterfeit-and-toxic>.
- [7] D. Law, Counterfeiting and consumer harm, 2019. URL: <https://www.lexology.com/library/detail.aspx?g=34dcef89-4614-4215-8111-95935d554513>.
- [8] S. Alipour, Ninety Eight Per Cent Of Fake Or Lookalike iPhone Chargers Put Consumers At Risk Of Lethal Electric Shock And Fire, 2017. URL: <https://www.electricalsafetyfirst.org.uk/media-centre/press-releases/2017/12/ninety-eight-per-cent-of-fake-or-lookalike-iphone-chargers-put-consumers-at-risk-of-lethal-electric-shock-and-fire/>.
- [9] B. Casassus, Health agency reveals scourge of fake drugs in developing world, 2017. URL: <https://www.nature.com/news/health-agency-reveals-scurge-of-fake-drugs-in-developing-world-1.23051>.
- [10] M. Yarovaya, Poddelki na marketpleysakh: kak s nimi boryutsya v YES i chto delat' ukrainskim pokupatelyam, 2019. URL: <https://ain.ua/2019/03/21/poddelki-na-marketpleysax>.
- [11] S. Rybachuk, Kontrafakt v e-commerce: pobedit', nel'zya smirit'sya, 2020. URL: <https://www.retail.ru/articles/kontrafakt-v-e-commerce-pobedit-nelzya-smiritsya/>.
- [12] Ye. Bashurina, Poddel'naya radost': kak brendy boryutsya s kontrafaktnoy kosmetikoy s pomoshch'yu blokcheyna, 2021. URL: <https://www.forbes.ru/forbes-woman/420249-poddelnaya-radost-kak-brendy-boryutsya-s-kontrafaktnoy-kosmetikoy-s-pomoshchyu>.
- [13] Group-IB Digital Risk Protection, 2019. URL: [https://www.tadviser.ru/index.php/%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82:Group-IB_Digital_Risk_Protection_\(%D1%80%D0%B0%D0%BD%D0%B5%D0%B5_Brand_Protection\)](https://www.tadviser.ru/index.php/%D0%9F%D1%80%D0%BE%D0%B4%D1%83%D0%BA%D1%82:Group-IB_Digital_Risk_Protection_(%D1%80%D0%B0%D0%BD%D0%B5%D0%B5_Brand_Protection)).
- [14] M.O. Medykovskyi, I.G. Tsmots, O.V. Skorokhoda, Spectrum neural network filtration technology for improving the forecast accuracy of dynamic processes in economics, volume 162(12) of Actual Problems of Economics, 2014, pp. 410-416.
- [15] IBM has found a way to defeat counterfeiting, 2018. URL: <https://getsiz.ru/ibm-nashla-sposob-pobedit-kontrafakt.html>.
- [16] I. Mozul, IBM sozdala portativnyy detektor kontrafaktnoy produktsii, 2018. URL: <https://itc.ua/news/ibm-sozdala-portativnyiy-detektor-kontrafaktnoy-produktsii/>.
- [17] Defeating Counterfeiting Begins With Smart Packaging, 2020. URL: <https://www.sepioproducts.com/blog/2020/06/23/defeating-counterfeiting-begins-with-smart-packaging/>.
- [18] How holograms can stop counterfeiting, 2014. URL: <https://www.packagingdigest.com/smart-packaging/how-holograms-can-stop-counterfeiting>.
- [19] How Smart Packaging In Pharmaceuticals Can Take The Fight To Counterfeit Medicines, 2019. URL: <https://qliktagsoftware.medium.com/how-smart-packaging-in-pharmaceuticals-can-take-the-fight-to-counterfeit-medicines-b9618fe17791>.
- [20] M. Venkataraman, How Blockchain Can Fight Counterfeiting and Fraud, 2019. URL: <https://www.globaltrademag.com/how-blockchain-can-fight-counterfeiting-and-fraud/>.

- [21] P. Bidyuk, A. Gozhyj, Y. Matsuki, N. Kuznetsova, I. Kalinina, Modeling and forecasting economic and financial processes using combined adaptive models, volume 1246 of *Advances in Intelligent Systems and Computing*, 2021, AISC, pp. 395-408.
- [22] V. Danylyk, V. Vysotska, V. Lytvyn, S. Vyshemyrska, I. Lurie, M. Luchkevych, Detecting Items with the Biggest Weight Based on Neural Network and Machine Learning Methods, volume 1158 of *Communications in Computer and Information Science*, 2020, pp. 383-396.
- [23] R. Martin, *Clean Architecture: A Craftsman's Guide To Software Structure And Design*. U.S.A.: Pearson, 2017.
- [24] V. Vysotska, A. Berko, V. Lytvyn, P. Kravets, L. Dzyubyk, Y. Bardachov, S. Vyshemyrska, Information Resource Management Technology Based on Fuzzy Logic, volume 1246 of *Advances in Intelligent Systems and Computing*, 2020, pp. 164-182. DOI: 10.1007/978-3-030-54215-3_11
- [25] C. Walls, *Spring in Action*. New York: Manning Publications, 2018.
- [26] C. Walls, *Spring Boot in Action*. New York: Manning Publications, 2018.
- [27] S. Grinev, *Mastering JavaFX 10: Build advanced and visually stunning Java applications*. New York: Packt Publishing, 2018.
- [28] S. Chaitanya, *JSON Tutorial: Learn JSON in 10 Minutes*, 2015. URL: <https://beginnersbook.com/2015/04/json-tutorial/>.
- [29] M. Aravind, *Gson Library*, 2017. URL: <https://medium.com/@manuaravindpta/gson-library-b7d4ef0381e2>.
- [30] *Maven: The Definitive Guide: The Definitive Guide* – New York: O'Reilly Media, 2008.
- [31] R. Schlager, *The OSI Model: simply explained*. New York: CreateSpace Independent Publishing Platform, 2013.
- [32] D. Gourley, B. Totty, M. Sayer, A. Aggarwal, *HTTP: The Definitive Guide: The Definitive Guide*. New York: O'Reilly Media, 2002.
- [33] B. Kurniawan, *How Tomcat Works: A Guide to Developing Your Own Java Servlet Container*. New York: Brainy Software, 2005.
- [34] C. Aquino, T. Gandee, *Front-End Web Development: The Big Nerd Ranch Guide*. New York: Big Nerd Ranch Guides, 2016.
- [35] B. Kommadi, *IntelliJ vs Eclipse Complete IDE Comparison*, 2019. URL: <https://medium.com/@bhagvankommadi/hi-team-6d2dee22d8b2>.
- [36] J. Reed, *Blockchain: The Essential Guide to Understanding the Blockchain Revolution*. New York: CreateSpace Independent Publishing Platform, 2016.
- [37] G. Walker, *Block Header. A summary of the data in the block*, 2016. URL: <https://learnmeabitcoin.com/technical/block-header>.
- [38] G. Walker, *Merkle Root. A fingerprint for all the transactions in a block*, 2016. URL: <https://learnmeabitcoin.com/technical/merkle-root>.
- [39] R. Canty, *Understanding Cryptography with RSA*, 2020. URL: <https://jryancanty.medium.com/understanding-cryptography-with-rsa-74721350331f>.
- [40] A. Nagpal, *How to create your own decentralized file sharing service using python*, 2018. URL: <https://medium.com/@amannagpal4/how-to-create-your-own-decentralized-file-sharing-service-using-python-2e00005bdc4a>.
- [41] I. Lurie, et al., Inductive technology of the target clusterization of enterprise's economic indicators of Ukraine, *CEUR Workshop Proceedings*, 2019, 2353, pp. 848-859.
- [42] V. Lytvynenko, D. Nikytenko, M. Voronenko, N. Savina, O. Naumov, Assessing the Possibility of a Country's Economic Growth Using Dynamic Bayesian Network Models, in: *IEEE 15th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT*, 2020, pp. 36-39.
- [43] V. Lytvynenko, et al., Comparative studies of self-organizing algorithms for forecasting economic parameters, *Int. Journal of Modern Education and Computer Science*, 2020, 12(6), pp. 1-15.
- [44] M. Voronenko, D. Nikytenko, J. Krejci, N. Savina, V. Lytvynenko, Assessing the possibility of a country's economic growth using static Bayesian network models, volume 2608 of *CEUR Workshop Proceedings*, 2020, pp. 462-473.
- [45] M. Grinchenko, O. Cherednichenko, I. Babych, Long-term forecasting technology of macroeconomic systems development: Regional aspect, volume 137 of *Lecture Notes in Business Information Processing*, 2013, pp. 49-60.

- [46] P. Pukach, K. Shakhovska, The mathematical method development of decisions supporting concerning products placement based on analysis of market basket content, in: 14th International Conference The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2017, pp. 347-350.
- [47] V. Ilkiv, Z., Nytrebych, P. Pukach, I. Kohut, B. Pakholok, Order relation on scalar products in real linear spaces, in: 15th International Conference on the Experience of Designing and Application of CAD Systems, CADSM, 2019, pp. 32-35.
- [48] A. Demchuk, V. Lytvyn, V. Vysotska, M. Dilai, Methods and Means of Web Content Personalization for Commercial Information Products Distribution, volume 1020 of Advances in Intelligent Systems and Computing, 2020, pp. 332-347. DOI: 10.1007/978-3-030-26474-1_24
- [49] V. Lytvyn, V. Vysotska, Y. Burov, O. Veres, I. Rishnyak, The Contextual Search Method Based on Domain Thesaurus, volume 689 of Advances in Intelligent Systems and Computing, 2018, pp. 310-319. () DOI: 10.1007/978-3-319-70581-1_22
- [50] V. Vysotska, L. Chyrun, L. Chyrun, Information Technology of Processing Information Resources in Electronic Content Commerce Systems, in: Proceedings of the International Conference on Computer Sciences and Information Technologies, CSIT, 2016, pp. 212-222. DOI: 10.1109/STC-CSIT.2016.7589909
- [51] V. Vysotska, I. Rishnyak, L. Chyrun, Analysis and evaluation of risks in electronic commerce, in: Proceedings of the 9th International Conference on CAD Systems in Microelectronics, 2007, pp. 332-333. DOI: 10.1109/CADSM.2007.4297570
- [52] K. Aliksieieva, A. Berko, V. Vysotska, Technology of commercial web-resource processing, in: Proceedings of 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2015.
- [53] V. Vysotska, L. Chyrun, Methods of information resources processing in electronic content commerce systems, in: Proceedings of 13th International Conference: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2015.
- [54] O. Cherednichenko, O. Yanholenko, O. Kanishcheva, Developing the Key Attributes for Product Matching Based on the Item's Image Tag Comparison, volume Vol-2631 of CEUR Workshop Proceedings, 2020, pp. 237-247.
- [55] O. Piatykop, O. Pronina, Model Selection of the Target Audience in Social Networks in Order to Promote the Product, volume Vol-2604 of CEUR workshop proceedings, 2020, pp. 396-406.
- [56] B. Rusyn, L. Pohreliuk, O. Kapshii, J. Varetsky, A. Demchuk, I. Karpov, A. Gozhyj, V. Gozhyj, I. Kalinina, An Intelligent System for Commercial of Information Products Distribution Based SEO and Sitecore CMS, CEUR workshop proceedings, 2020, Vol-2604, pp. 760-777.
- [57] T. Borovska, D. Grishin, I. Kolesnik, V. Severilov, I. Stanislavsky, T. Shestakevych, Research and Development of Models and Program for Optimal Product Line Control, Advances in Intelligent Systems and Computing IV, Springer Nature Switzerland AG, 2020, 1080, pp. 186-201.
- [58] V. Lytvyn, V. Vysotska, V. Shatskykh, I. Kohut, O. Petruchenko, L. Dzyubyk, V. Bobrivetc, V. Panasyuk, S. Sachenko, M. Komar, Design of a recommendation system based on Collaborative Filtering and machine learning considering personal needs of the user, volume 4(2-100) of Eastern-European Journal of Enterprise Technologies, 2019, pp. 6-28. DOI: 10.15587/1729-4061.2019.175507
- [59] I. Rishnyak, O. Veres, V. Lytvyn, M. Bublyk, I. Karpov, V. Vysotska, V. Panasyuk, Implementation models application for IT project risk management, volume Vol-2805 of CEUR Workshop Proceedings, 2020, pp. 102-117.
- [60] T. Shestakevych, V. Pasichnyk, M. Nazaruk, M. Medykovskiy, N. Antonyuk, Web-Products, Actual for Inclusive School Graduates: Evaluating the Accessibility, volume 871 of Advances in Intelligent Systems and Computing, 2019, pp. 350-363.
- [61] N. Shakhovska, Consolidated processing for differential information products, in: International Conference on Perspective Technologies and Methods in MEMS Design, 2011, pp. 176-177.
- [62] N. Shakhovska, O. Vovk, Y. Kryvenchuk, Uncertainty reduction in Big data catalogue for information product quality evaluation, volume 1(2-91) of Eastern-European Journal of Enterprise Technologies, 2018, pp. 12-20.
- [63] V. Vysotska, A. Berko, M. Bublyk, L. Chyrun, A. Vysotsky, K. Doroshkevych, Methods and tools for web resources processing in e-commercial content systems, in: IEEE 15th International

- Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2020, pp. 114-118. DOI: 10.1109/CSIT49958.2020.9321950
- [64] V. Kuchkovskiy, N. Shakhovska, Information technology of Blockchain: Database, smart contracts, architecture, in: IEEE 14th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2019, pp. 55-59.
- [65] A. Berko, M. Bublyk, L. Chyrun, Y. Matseliukh, R. Levus, V. Panasyuk, O. Brodyak, L. Dzyubyk, O. Garbich-Moshora, Models and Methods for E-Commerce Systems Designing in the Global Economy Development Conditions Based on Mealy and Moore Machines, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1574-1593.
- [66] M. Bublyk, Y. Matseliukh, Small-Batteries Utilization Analysis Based on Mathematical Statistics Methods in Challenges of Circular Economy, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1594-1603.
- [67] M. Bublyk, V. Mykhailov, Y. Matseliukh, T. Pihniak, A. Selskyi, I. Grybyk, Change Management in R&D-Quality Costs in Challenges of the Global Economy, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1139-1151.
- [68] A. Berko, I. Pelekh, L. Chyrun, M. Bublyk, I. Bobyk, Y. Matseliukh, L. Chyrun, Application of ontologies and meta-models for dynamic integration of weakly structured data, in: Proceedings of the IEEE 3rd International Conference on Data Stream Mining and Processing, DSMP, 2020, pp. 432-437. DOI: 10.1109/DSMP47368.2020.9204321
- [69] L. Chyrun, The E-Commerce Systems Modelling Based on Petri Networks, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1604-1631.
- [70] A. Berko, V. Andrunyk, L. Chyrun, M. Sorokovskyy, O. Oborska, O. Oryshchyn, M. Luchkevych, O. Brodovska, The Content Analysis Method for the Information Resources Formation in Electronic Content Commerce Systems, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1632-1651.
- [71] V. Kuchkovskiy, V. Andrunyk, M. Krylyshyn, L. Chyrun, A. Vysotskyi, S. Chyrun, N. Sokulska, I. Brodovska, Application of Online Marketing Methods and SEO Technologies for Web Resources Analysis within the Region, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1652-1693.
- [72] V. Vysotska, V.B. Fernandes, M. Emmerich, Web content support method in electronic business systems, volume Vol-2136 of CEUR Workshop Proceedings, 2018, pp. 20-41.
- [73] L. Chyrun, et al., Online Business Processes Support Methods, in: IEEE 15th International Scientific and Technical Conference on Computer Sciences and Information Technologies, CSIT, 2020, pp. 125-133.
- [74] A. Kopp, D. Orlovskiy, S. Orekhov, An Approach and Software Prototype for Translation of Natural Language Business Rules into Database Structure, volume Vol-2870 of CEUR Workshop Proceedings, 2021, pp. 1274-1291.
- [75] H. Lipyanina, A. Sachenko, T. Lendyuk, S. Nadvynychny, S. Grodskiy, Decision tree based targeting model of customer interaction with business page, CEUR Workshop Proceedings, 2020, pp. 1001-1012.
- [76] R. Allmendinger, M.T.M. Emmerich, J. Hakanen, Y. Jin, E. Rigoni, Surrogate-assisted multicriteria optimization: Complexities, prospective solutions, and business case, volume 24(1-2) of Journal of Multi-Criteria Decision Analysis, 2017, pp. 5-24.
- [77] I. Oksanych, I. Shevchenko, I. Shcherbak, S. Shcherbak, Development of specialized services for predicting the business activity indicators based on micro-service architecture, volume 2(2-86) of Eastern-European Journal of Enterprise Technologies, 2017, pp. 50-55.
- [78] A.Y. Berko, Methods and models of data integration in E-business systems, volume 10 of Actual Problems of Economics, 2008, pp. 17-24.
- [79] A. Berko, Consolidated data models for electronic business systems, in: The Experience of Designing and Application of CAD Systems in Microelectronics, CADSM, 2007, pp. 341-342.