Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(назва освітнього ступеня)

на тему: Система забезпечення справжності документів
(на основі технології Blockchain)

Виконав: студент IV курсу, групи СТс-42
спеціальності 126 Інформаційні системи та технології
(шифр і назва спеціальності)

_____ Ониськів П.П.
(підпис) (прізвище та ініціали)

Керівник _____ Гром'як Р.С.
(підпис) (прізвище та ініціали)

Нормоконтроль _____ Шимчук Г.В.
(підпис) (прізвище та ініціали)

Завідувач кафедри _____ Боднарчук І.О.
(підпис) (прізвище та ініціали)

Рецензент _____ Михайлишин М.С.
(підпис) (прізвище та ініціали)

Тернопіль - 2021

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Боднарчук І.О.
(підпис) (прізвище та ініціали)
«__» _____ 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 126 Інформаційні системи та технології
(шифр і назва спеціальності)

Студенту Ониськіву Петру Петровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Система забезпечення справжності документів (на основі технології Blockchain)

Керівник роботи Гром'як Роман Сільвестрович., к.ф.-м.н., доц. каф. КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «01» 02 2021 року № 4/7-63

2. Термін подання студентом завершеної роботи 26.06.2021р.

3. Вихідні дані до роботи наукові літературні джерела

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступ. 1. Аналіз досліджуваної області. 1.1 Зберігання документів в архіві в паперовому вигляді. 1.2. Електронний архів. 1.3. Способи забезпечення достовірності документів. 2. Проектна частина. 2.1. Існуючі реалізації електронного архіву на основі технології Blockchain. 2.2. Загальний опис розроблюваної системи. 2.3. Обґрунтування вибору Технологій. 2.4. Опис API системи. 3. Програмна реалізація системи. 3.1. API додатка основної бізнес-логіки. 3.2. API додатка для зберігання документів з підвищеною захищеністю. 3.3 API додатка для кожного вузла в мережі Blockchain. 4. Безпека життєдіяльності, основи хорони праці. Висновки. Перелік використаних джерел

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Титульний слайд. 2. Актуальність. 3. Мета, задачі дослідження. 4. Поняття електронного архіву, основні проблеми. 5. Способи забезпечення достовірності документів. 6. Особливості технології Blockchain. 7. Існуючі реалізації електронного архіву на основі технології Blockchain. 8. Загальний опис розроблюваної системи. 9. Схема загального представлення системи. 10. Схема зберігання документів в Blockchain. 11. Технології, які використовувалися в розробці. 12. Програмна реалізація системи. 13. Висновки.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи хорони праці	Гурик О.Я., доцент кафедри МТ		

7. Дата видачі завдання 2021 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	01.02 – 05.02	*Виконано*
2.	Підбір джерел про забезпечення справжності документів	05.02 – 15.02	*Виконано*
3.	Опрацювання джерел про способи забезпечення		
	достовірності документів	16.02 – 27.02	*Виконано*
4.	Виконання дослідження щодо система забезпечення		
	справжності документів (на основі технології		
	Blockchain)	28.02 – 15.03	*Виконано*
5	Розроблення програмного коду	16.03 – 10.04	*Виконано*
6.	Оформлення розділу «Аналіз досліджуваної області»	11.04 – 20.04	*Виконано*
7.	Оформлення розділу «Проектна частина»	21.04 – 29.04	*Виконано*
8.	Оформлення розділу «Програмна реалізація системи»	30.04 – 11.05	*Виконано*
9.	Виконання завдання до підрозділу «Безпека		
	життєдіяльності, основи хорони праці»	12.05 – 19.05	*Виконано*
10.	Оформлення кваліфікаційної роботи	20.05 – 28.05	*Виконано*
11.	Нормоконтроль	05.06 – 12.06	*Виконано*
12.	Перевірка на плагіат	12.06 – 15.06	*Виконано*
13.	Попередній захист кваліфікаційної роботи	16.06 – 19.06	*Виконано*
14.	Захист кваліфікаційної роботи	26.06	

Студент _____ Ониськів П.П.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Гром’як Р.С.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Система забезпечення справжності документів (на основі технології Blockchain) // Кваліфікаційна робота бакалавра // Ониськів Петро Петрович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, кафедра комп'ютерних наук, група СТс–42 // Тернопіль, 2021 // С. – 52, рис. – 27, табл. – 1, слайдів – 13 , бібліогр. – 19.

Ключові слова: API, BLOCKCHAIN, БАЗА ДАНИХ, ЕЛЕКТРОННИЙ АРХІВ, ДОСТОВІРНІСТЬ ДОКУМЕНТІВ

Кваліфікаційна робота присвячена створенню прототипу системи забезпечення достовірності документів із тривалим терміном зберігання в електронному архіві, який використовує технологію Blockchain.

Описані особливості електронного архіву, проведено аналіз способів зберігання документів в архіві та обґрунтовано технології, які використовуються для реалізації обраного способу. Також досліджено існуючі способи забезпечення достовірності документів та існуючі реалізації електронного архіву на основі технології Blockchain.

Обґрунтовано вибір технологій, наведено опис API системи. Реалізовано прототип системи електронного архіву, що використовує технологію Blockchain, для вирішення проблеми автентичності документів тривалого терміну зберігання.

ANNOTATION

Documents authenticity providing system (Blockchain-based) // Oniskiv Petro // Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Computer Science // Ternopil, 2021 // P. - 52, Fig. - 27, Table - 1, Slide - 13, References - 19.

Keywords: API, BLOCKCHAIN, DATABASE, ELECTRONIC ARCHIVE, AUTHENTICITY OF THE DOCUMENT

Thesis deals with the creation of a prototype of a system for ensuring the authenticity of documents with a long shelf life in an electronic archive, which uses Blockchain technology.

Features of the electronic archive are described, the analysis of ways of storage of documents in archive is carried out and the technologies which are used for realization of the chosen way are substantiated. Existing methods of ensuring the authenticity of documents and existing implementations of electronic archives based on Blockchain technology are also investigated.

The choice of technologies is substantiated, the description of API of the system is given. A prototype of an electronic archive system using Blockchain technology has been implemented to solve the problem of authenticity of long-term storage documents.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

API (Application Programming Interface) – програмний інтерфейс до датку; набір готових класів, процедур, функцій, структур і констант, наданих додатком для використання у зовнішніх програмних продуктах.

Blockchain (укр. Блокчейн) – публічний ланцюжок з формованих блоків транзакцій. Запис даних в Blockchain здійснюється шляхом додавання нових блоків. Ланцюжок зберігає дані про всі транзакції, які будь коли відбувалися, починаючи з вихідного блоку.

Nonce – одноразовий код, який використовується при обчисленні хешу заданої складності.

P2P-network (peer-to-peer network) – тимчасова мережа.

БД – база даних.

Блок – набір транзакцій та інших даних, котрий є ланкою ланцюга Blockchain.

ІС- інформаційна система.

ІТ – інформаційні технології.

ЕА – електронний архів.

ЕД – електронний документ.

ЕП – електронний підпис.

Майнінг – діяльність по обчисленню хешу заданої складності для створення нових блоків в Blockchain.

Нода – вузол мережі Blockchain.

ОС – операційна система.

ПК – персональний комп’ютер.

ПЗ – програмне забезпечення.

СУБД – система управління базами даних.

ЗМІСТ

ВСТУП

В даний час в світі досить широко використовуються повноцінні, юридично значущі ЕД, що підлягають зберіганню упродовж тривалих термінів, встановлених законами і іншими нормативними актами. Деякі з цих документів мають постійну цінність і повинні передаватися на державне архівне зберігання [1]. Наразі обов'язковою частиною ЕА є ЕП. ЕП володіє обмеженим терміном дії сертифіката, який в середньому становить 1 рік [2], отже для документів, термін зберігання яких є тривалим або постійним, необхідно регулярне оновлення ЕП. Оновлення ЕП для значного числа документів є витратною операцією.

Таким чином, виникає проблема відсутності такого ЕА, який забезпечить справжність документів з тривалим чи постійним терміном зберігання, не вимагаючи регулярного поновлення ЕП. На даний момент існує кілька розробок по цій тематиці, але поки не має відповідного ПЗ для конфіденційного державного архівного зберігання.

Мета даної роботи – створення прототипу системи забезпечення достовірності документів з тривалим терміном зберігання в ЕА, який використовує технологію Blockchain.

Для досягнення поставленої мети, потрібно забезпечити вирішення наступних задач:

– дослідити способи зберігання документів, які мають тривалий термін зберігання в архіві;

– дослідити поняття ЕА, проблеми ЕА в Україні;

–дослідити способи забезпечення достовірності документів з тривалим терміном зберігання в ЕА;

– розглянути існуючі реалізації системи підтвердження справжності документів ЕА на основі технології Blockchain;

– реалізувати прототип системи ЕА, що використовує технологію Blockchain для вирішення проблеми автентичності документів тривалого терміну зберігання.

1 АНАЛІЗ ДОСЛІДЖУВАНОЇ ОБЛАСТІ

1.1 Зберігання документів в архіві в паперовому вигляді

Цей спосіб зберігання інформації є найдавнішим [1]. Даний спосіб має ряд недоліків, в зв'язку з якими електронний вигляд архіву отримує все більшу поширеність.

Недоліки зберігання документів в друкованому вигляді:

– при великій кількості документів архів вимагає великої площі приміщення, де дані документи будуть зберігатися;

– для збереження фізичного стану документа необхідно дотримуватися режими зберігання документів, перерахованих в [1].

– фінансові витрати на охоронний режим [2 - 4];

– складний і довготривалий пошук по документам.

1.2 ЕА

Архів ЕД є не просто якимось матеріальним носієм і ІС, що дозволяє швидко виконувати пошук потрібної інформації, а й набором технологій і процесів, котрі забезпечують весь цикл операцій з документами від створення, експертизи цінності до їх використання, через облік, опис, забезпечення схоронності і розвиток науково-довідкового апарату [6].

Хоча ЕД з'явилися ще в СРСР 1970 рр. (1984 р ГОСТ 6.10.4-84 закріпив факт наявності в документальній середовищі документації на нових носіях), технічні і організаційні питання їх довготривалого зберігання дотепер не вирішені. Це пов'язано як з прискореними темпами розвитку ІТ, так і ІС.

ЕА призначений для того, щоб вирішити проблеми паперового архіву перераховані в пункті 1.1. Дану задачу ЕА виконує, але з'являються інші складнощі.

1.2.1 Види ЕА

Зберігання документів на матеріальних носіях. Традиційні ЕА для зберігання даних яких використовуються матеріальні носії (жорсткі диски, твердотільні накопичувачі). Перегляд і читання інформації, що зберігаються на таких носіях виконується при використанні технічних засобів, аналогічних або сумісних з тими системами, які були використані при створенні архіву.

Хоча даний спосіб зберігання документів вирішує такі проблеми паперового архіву, як: необхідність великого простору для зберігання, дотримання необхідних режимів зберігання документів, матеріальні носії не забезпечують швидкий і зручний пошук по документам. Також з'являється проблема довгострокового зберігання Дані про збереження оптичних дисків в архівах показують, що за кілька років близько 5% носіїв стають нечитабельним або «Проблемними» [5] (наявність відбитків пальців і подряпин на дисках). Також сучасні комп'ютери і ноутбуки не підтримують читання таких матеріальних носіїв, як оптичні диски і дискети.

Тому для зберігання ЕД на тривалий або постійний термін матеріальні носії не підходять.

Зберігання документів в віртуальних хмарних сховищах. Більш сучасний вид ЕА є БД, зберігання якої здійснюється в віртуальних хмарних сховищах - спеціальних віддалених серверах. Перегляд і читання інформації з таких архівів також вимагає наявності комп'ютера і монітора, але доступ до архіву можливий тільки при існуванні Інтернет- каналу.

Віртуальні хмарні сховища зручні для пошуку, підходять для достатньо тривалого і постійного терміну зберігання, але вимагають витрат на написання відповідного ПЗ та, відповідно, його підтримки. Також важливою постає проблема інформаційної безпеки системи, яка використовується [1].

1.2.2 Основні проблеми ЕА

Забезпечення збереження ЕД. Для виконання цієї процедури надзвичайно важливим є правильний вибір типу носія інформації і його довговічність.

Детальніше типи носіїв будуть розглянуті в наступних розділах. Також варто відзначити, що для дотримання правил безпечного зберігання різних документів потрібно їх зберігати щонайменше у двох примірниках. Це своєрідна страхівка, коли при втраті одного з носіїв вся необхідна інформація не буде загублена.

Забезпечення коректного зчитування інформації на довготривалих термінах зберігання. Читання і застосування ЕД залежить, перш за все, від ПЗ, яке застосовується: ОС, СУБД, браузерів, інших прикладних програм. Варто відмітити, що раптова зміна програмної платформи може спричинити абсолютну втрату документа через відсутність коректної можливості його прочитати. Рішенням може послужити регулярна міграція документів на більш нові формати, перехід на використання нових прикладних програм. Або ж використання найбільш поширених форматів.

Відсутність необхідної нормативно-правової бази. На даний момент відсутній визначений порядок побудови ЕА, який був би закріплений законодавчо.

Різні вимоги органів виконавчої влади. Дана проблема випливає з попереднього пункту. Відсутність єдиних нормативних документів призводить до того, що державні органи самостійно встановлювати правила зберігання ЕД для підвідомчих і підконтрольних їм установ. Різні вимоги можуть у найближчому майбутньому створити додаткові проблеми для всіх зацікавлених сторін [5].

Забезпечення достовірності та цілісності ЕД. При переході з паперового архіву на електронний стає важливим питання інформаційної безпеки. Дана проблема буде розглянута в підрозділі 1.3.

1.3 Способи забезпечення достовірності документів

1.3.1 ЕП

На даний момент у країнах Європи прийнято, що обов'язковою частиною ЕА є ЕП. Фактично ЕП є інформацією у визначеній електронній формі, котра

приєднується до якоїсь іншої електронної інформації (називається «підписувана інформація») чи якимось іншим чином перебуває у зв’язку з такою інформацією та котра застосовується для встановлення тієї особи, котра цю інформацію підписує. Іншими словами, ЕП забезпечує достовірність авторства і вмісту ЕД.

Види ЕП:

– з симетричною схемою - наявність тільки відкритого ключа;

– з асиметричною схемою - наявність відкритого і закритого ключа.

Симетричні схеми ЕП менш поширені в зв'язку з наступним недоліком: ключі, згенеровані для підпису, з метою безпеки є одноразовими.

Найбільш безпечною на даний момент вважається асиметрична схема ЕП.

На рисунку 1.1 зображені етапи підписання і перевірки даних документа.

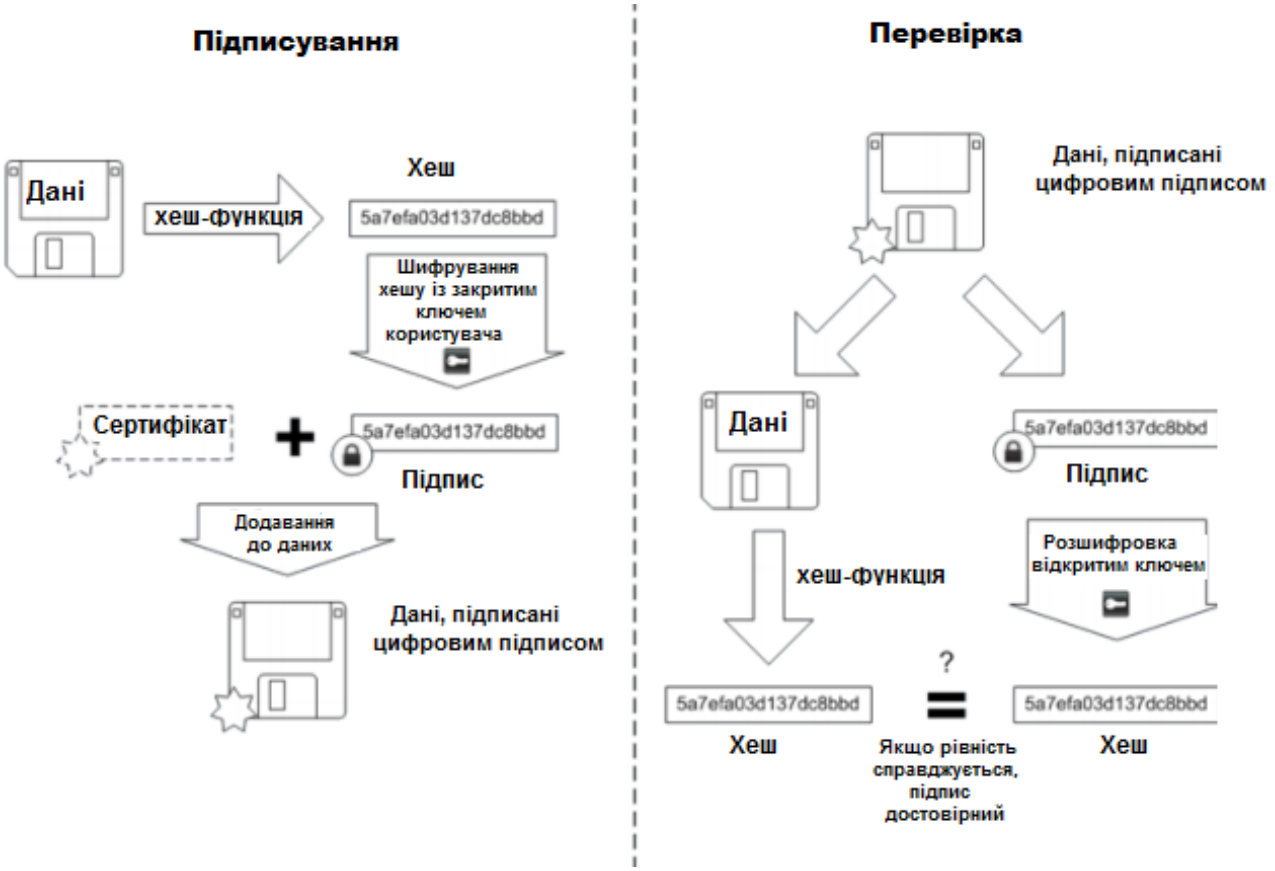

Рисунок 1.1 – Схема формування ЕП

Етапи підписання документа.

1. Дані переводяться в хеш.

2. Шифрування хешу закритим ключем користувача.

3. Прикріплення сертифіката (інформація про відкритий ключ та власника документа) [7].

4. Додавання отриманого ЕП до даних.

Необхідно навести етапи перевірки даних.

1. Дані переводяться в хеш.

2. Підпис розшифровується відкритим ключем.

3. Якщо хеш збігається з розшифрованим підписом, отже підпис вірний.

Проблема ЕП. Основною проблемою ЕП є обмежений термін дії сертифіката, який в середньому становить 1 рік [8].

Основні причини обмеженого терміну дії сертифіката:

– зміна власників і співробітників компанії, які володіють ЕП;

– розвиток стандартів безпеки. З постійним розвитком сучасних технологій ускладнюються і змінюються криптографічні алгоритми.

Для документів тривалого або постійного терміну зберігання операція поновлення ЕП є достатньо витратною.

1.3.2 Blockchain

Альтернативою використанню ЕП для забезпечення цілісності ЕД є технологія Blockchain, яка стрімко набирає популярність [10].

Дана модель була створена для роботи з криптовалютою Bitcoin. Основна ідея протоколу Blockchain – забезпечення прозорого виконання різних транзакцій між двома наперед незнайомими сторонами, які підтверджують свою достовірність, без залучення центральної складової. Незважаючи, властиво, на той факт, що Blockchain була розроблена фактично для підтримки Bitcoin, ця технологія може бути задіяна також і окремо від Bitcoin.

Варто згадати, що фактично, Blockchain є такою БД, в яку поміщаються певні факти, властиво повні копії цією БД знаходяться на всіх ПК, об’єднаних в P2P-мережу. Така БД може лінійно розширюватися у хронологічному порядку.

Ті факти, котрі попадають в БД підчас їх внесення, можуть бути змістовно різними, наприклад, грошові транзакції. Членами такої мережі є анонімні суб'єкти – вузли. Повинна забезпечуватися умова, при якій кожен свіжий вузол, котрий хоче стати членом мережі, обов'язково має завантажити повну копію Blockchain. Всередині мережі обов'язково використовуються криптографічні методи з метою виконати чітку і безпечну ідентифікацію як відправника, а також і одержувача. У тому випадку, коли новий вузол спробує виконати додавання факту в базу, в мережі повинен сформуватися консенсус, котрий власне і визначить, в котрому місці буде розміщений факт. Такий консенсус носить назву блоку. Схема децентралізації, на якій власне і базується протокол Bitcoin, переносить авторитет і довіру на мережу, яка є децентралізованою, і дозволяє її вузлам постійно і послідовно включати свої транзакції до складу загального блоку та вподальшому забезпечити створення Blockchain як унікального ланцюжка конфіденційних даних.

Можна визначити поняття блоку як спеціального способу організації фактів в мережі користувачів, які не мають атрибутів довірених. В основі блоку покладена достатньо проста ідея: необхідно згрупувати факти в блоки, скласти з таких окремих одиниць єдиний ланцюжок, котрий вподальшому буде реплікований по всіх решту вузлах мережі. Саме цей ланцюжок блоків буде видимий для всіх вузлів в мережі. В кожному наступному блоці міститься посилання на попередній, таким чином можна простежити походження кожного окремого факту. Хеш-код є засобом криптографії, який використовується безпечної аутентифікації самого джерела виконання транзакції і робить непотрібним необхідність існування центрального виконавця. Ніяка транзакція не може бути записаною повторно, що гарантується використанням комбінації засобів криптографії та власне Blockchain.

Blockchain - це послідовний безперервний ланцюжок блоків даних, копії якого зберігаються незалежно на декількох серверах [10]. Кожен блок містить в собі кілька транзакцій. В контексті ЕА транзакція - це інформація про передачу документа в архів і вміст документа (рисунок 1.2).

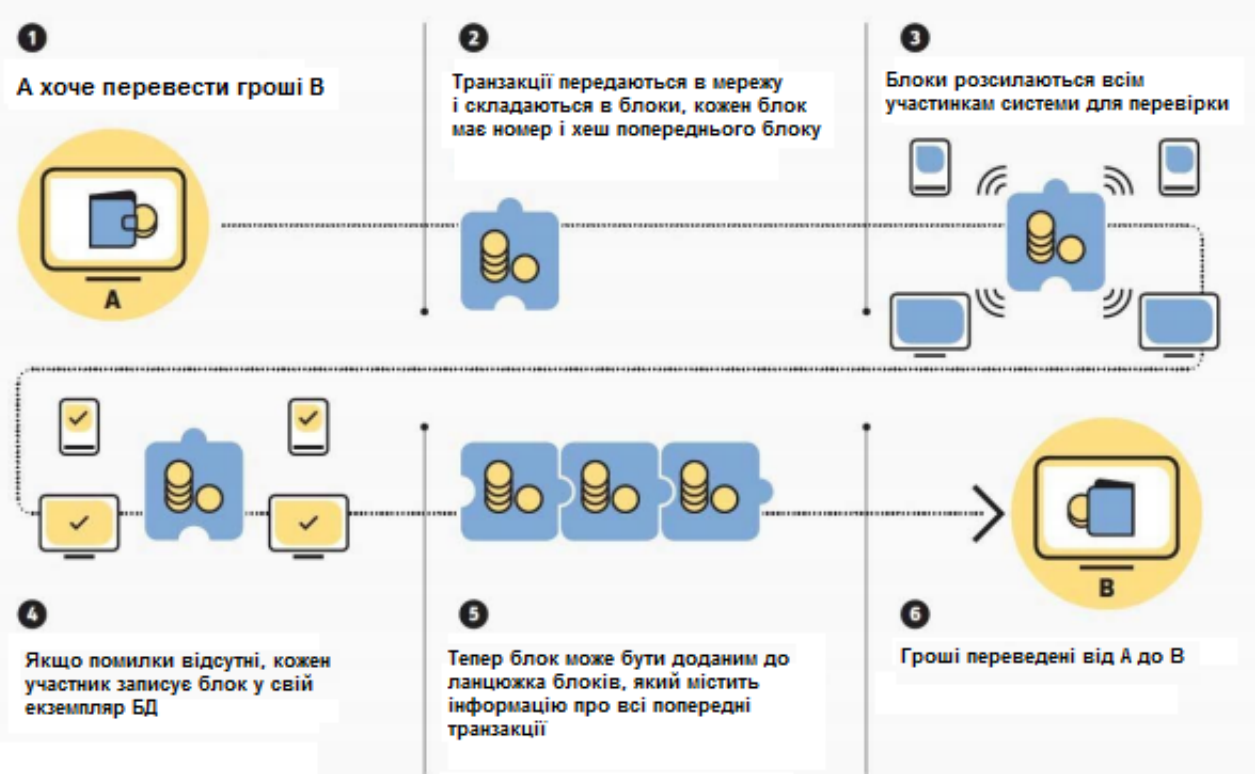

Рисунок 1.2 – Схема роботи Blockchain на прикладі криптовалюти

Необхідно згадати і про процес підтримки діяльності розподіленої платформи і самого «пошуку» блоків, який носить назву майнінгу. Жоден стандартний вузол не має можливості займатися майнінгом, оскільки лише отримує блоки від будь-яких інших вузлів. Власне процес, при якому звичайний вузол перетворюється у вузол-майнер, є абсолютно добровільним.

Для здійснення майнінгу комп’ютера-вузол повинен володіти достатньо великою потужністю в розумінні проведення обчислень. Сама ідея майнінгу – це здійснення вибору транзакцій (фактів) з їх очікуваного числа і наступному отриманні хеш-значення сформованого блоку транзакцій. Будь –який блок вважається успішно підтвердженим, коли його хеш-значення стартує з визначеного числа нулів, але, багато час отримані хеш-значення не є успішними, тому блок змінюється не значно і хеш-значення генерується знову. Вже той блок, який буде успішно підтвердженим, і включається до ланцюга та стане видимим для всіх інших вузлів цієї мережі.

Основна проблема ЕА, яку вирішує технологія Blockchain, - необхідність регулярного оновлення ЕП [14].

Основною характеристикою технології Blockchain є певна складність обчислення хешу блоку транзакцій. Ця складність вибирається свідомо високою для того, щоб зловмисник не зміг вирахувати весь ланцюжок хешів за короткий час [12].

Незважаючи на окремі недоліки самої технології Blockchain її використання в реалізації ЕА має наступні переваги [15]:

– децентралізація зберігання документів;

– можливість зберігання документів постійного терміну зберігання;

– забезпечення безпеки високого рівня.

2 ПРОЕКТНА ЧАСТИНА

2.1 Існуючі реалізації ЕА на основі технології Blockchain

2.1.1 BlockSign

Компанія Basno, Нью-Йорк США, створила веб-додаток BlockSign, доступний за посиланням blocksign.com , що дозволяє клієнтам ставити цифровий підпис під договором і зберігати його в громадському доступі. Платформа, запущена на основі Blockchain, надає можливість підписати, створити тимчасову мітку і пізніше перевірити документ на достовірність через Інтернет, таким чином довівши, що документ однозначно засвідчений певною людиною і не був змінений [10].

PDF-документ, підписаний за допомогою програми, містить в собі підпис. Цей підпис створюється за допомогою кодування з імені користувача, адреси електронної пошти та дати. Служба BlockSign потім створює криптографічний хеш всього документа, який він зберігає в 40-байтному слоті, що міститься в кожному біткоін-блоці, і називається OP-RETURN. Цей слот може бути використаний для зберігання повідомлень і інших довільних даних.

Якщо іншим людям необхідно підписати цей же документ, сайт попросить зробити те ж саме за допомогою свого підпису, але який містить інші ім'я користувача, адресу електронної пошти, що відповідають окремій людині.

Таким чином, хеші документа від кожного користувача реєструються в Blockchain. Перевірочні записи зберігаються на серверах у компанії від імені користувача і дозволяють перевірити, що конкретний документ підписаний певним користувачем і є дійсним.

Безпека системи спирається на електронну пошту. Система надсилає запит на підтвердження для користувача через електронну пошту, щоб підтвердити його особистість [10].

Недоліки BlockSign. Проаналізувавши роботу в веб-додатку BlockSign, можна виявити ряд недоліків.

1. Перевірочні записи, документи та інші дані зберігаються серверах, котрі є непідконтрольними користувачеві або його організації. Це є ризиком для підробки даних, так як невідомо хто може отримати доступ до серверів, якщо вони не будуть під контролем того, кому ці дані дійсно належать. Також це суперечить концепції Blockchain, коли безпека забезпечується децентралізацією серверів. У разі BlockSign все сервера в мережі Blockchain належать тільки цій компанії.

2. Відсутність відкритого API для реалізації приватного ЕА, використовуючи дану технологію [12].

Рішення недоліків BlockSign у власній реалізації. Для уникнення недоліків BlockSign можна знайти рішення. Розглянемо їх по черзі.

1. Зберігання перевірочних записів, документів та інших даних на непідконтрольних користувачеві або його організації серверах. Рішення - розмістити зберігання даних тільки на серверах організації, кому належать дані.

2. Відсутність відкритого API. Рішення - реалізація відкритого API для можливої подальшої інтеграції з іншими ресурсами.

2.1.2 Ethereum

Є платформою для побудови децентралізованих сервісів в он-лайн режимі на основі Blockchain, котрі працюють на базі смарт - контрактів. Власне сам смарт-контракт є комп'ютерним алгоритмом, котрий використовується з метою укладання та подальшої підтримки контрактів, які виконуються самостійно в середовищі - Blockchain.

Функціонально і програмно платформа реалізована у вигляді єдиної віртуальної машини без централізації [16].

Кожен-смарт контракт має свої атрибути [16]. Серед них обов'язковими є:

– застосування різних методів ЕП на базі відкритих і закритих ключів, котрими володіють дві або більше сторін угоди;

– існування приватного децентралізованого середовища (в т.ч., Ethereum);

– власне предмет договору та існування потрібних для його виконання засобів (рахунків у криптовалюті, програм-оракулів та інших);

– чітко прописані умови виконання, підтверджені підписами учасників договору, а також достовірність джерела надання цифрових даних.

Blockchain Ethereum за своєю суттю є системою стану транзакцій. В IT визначено поняття «система станів» чи по іншому «машина станів» - це така система, котра опрацьовує введену інформацію і на її підставі переходить в певний новий стан. Стан Ethereum має мільйони діючих транзакцій. Всі ці транзакції є згрупованими в певні блоки. Кожен блок містить визначену кількість транзакцій, в той же час кожен наступний блок є з'єднаним з попереднім. Саме тому і забезпечується своєрідний неперервний ланцюжок блоків.

Щоб відбувся перехід з одного стану блоку в інший, треба щоб будь-яка транзакція була коректною. Власне коректною транзакція може бути тільки в тому випадку, коли її перевірили спеціальним процесом - майнінгом. Властиво будь-який вузол в мережі, котрий оголосив себе майнером, має змогу створити і виконати перевірку блоку транзакцій.

Визначений вузол-майнер має подати свій математичний доказ коректності швидше за іншого конкурента з метою додавання його блоку до основного блокчейну. Тут використовується термін «доказ роботи» – це процес перевірки кожного блоку на надання свого математичного доказу.

Майнер, котрий створює новий блок, отримує за це певну винагороду. У Ethereum використовується вбудований цифровий токен, котрий називається «ефір» (від англ. Ether- «ефір»). Кожного разу, коли майнер створює свій блок транзакцій, відповідно створюється новий токен або новий ефір, який і буде винагородою за створення такого блоку.

Ethereum стрімко розвивається і набирає популярність серед платформ на основі технології Blockchain. Але дана платформа не підходить реалізації прототипу для вирішення поставленого завдання з наступних причин:

– наявність користувачів-майнерів в системі суперечить умові секретності державних документів;

– обов'язкове використання методів ЕП заважає вирішити проблему терміну дії сертифіката в ЕА;

– кожна транзакція (завантаження документа) повинна супроводжуватися грошовою винагородою для користувача-майнера.

2.2 Загальний опис розроблюваної системи

Розроблювана система містить в своєму складі чотири основні компоненти (частини):

– клієнтська сторона;

– додаток з основною бізнес-логікою;

– додаток для зберігання документів з підвищеною захищеністю;

– додаток для вузлів зберігання даних в мережі Blockchain.

Вони розділені на дві групи:

– клієнтська сторона, яка взаємодіє безпосередньо з користувачем. Працює в браузері користувача.

– серверна сторона, яка знаходиться на віддалених серверах і недоступна користувачеві.

На рисунку 2.1 представлено як користувач взаємодіє з системою. Архітектура була створена на основі патерну Модель-Представлення-Контролер. Користувач може завантажити і отримати документ з архіву.

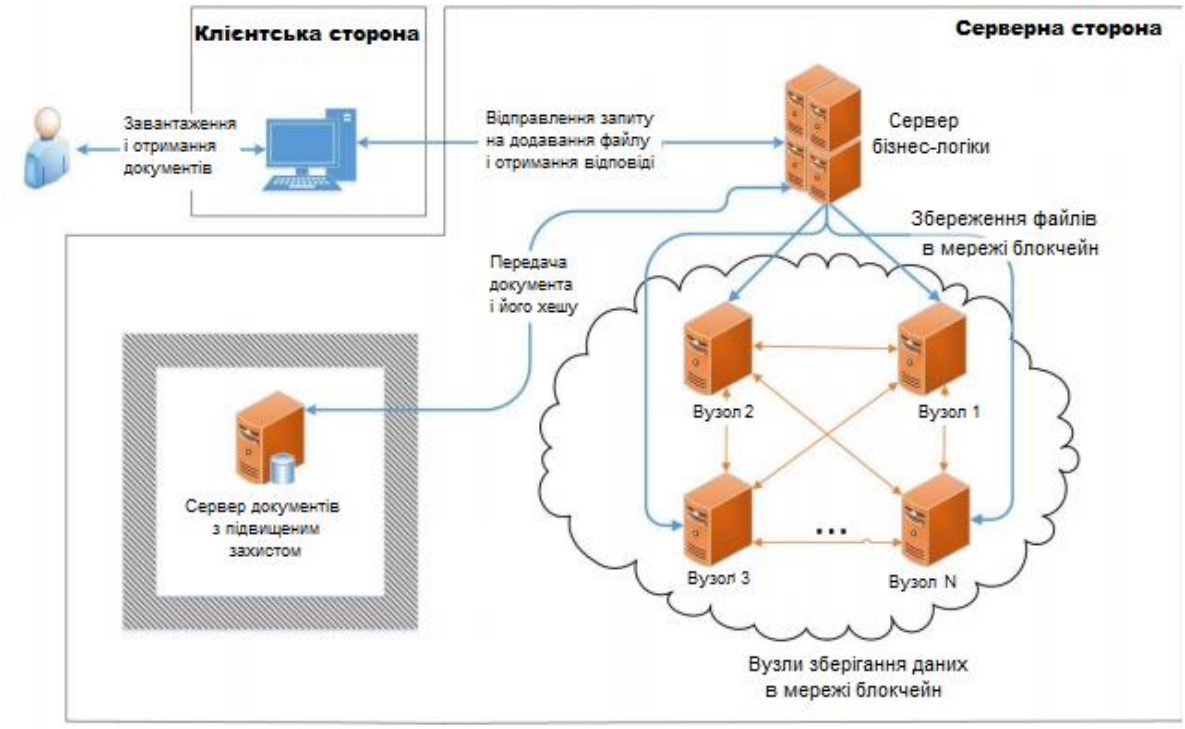

Рисунок 2.1 – Схема загального представлення системи

Завантаження документа. У разі завантаження документа через форму в браузері, дані відправляються на сервер бізнес-логіки, на якому перевіряються права користувача і вираховується хеш документа (рис. 2.2).

Рисунок 2.2 – Схема зберігання документів в Blockchain

Цей хеш даних файлу відправляються на захищений сервер даних.

Також сервер бізнес-логіки відправляє хеш документа у вигляді транзакції на всі вузли мережі Blockchain.

Кожен вузол додає отриману транзакцію в блок. Після накопичення певної кількості транзакцій в блоці, блок закривається. Далі вузли розподілено майнять блок, тобто вираховують його хеш на основі на всіх його транзакцій, хешу попереднього блоку в ланцюжку і одноразового коду nonce, с допомогою якого досягається задана складність хешу.

Перший вузол, який вирахував хеш із заданою складністю розсилає його і знайдений nonce решті вузлам. Кожен вузол звіряє отриманий хеш з тим, який він обчислює самостійно на основі свого ланцюжка блоків та отриманого nonce. У разі рівного розподілу вузол додає замайненний блок в ланцюжок і повідомляє про успішний результат.

Що стосується розбіжності хешів вузол посилає повідомлення про помилку, на даному вузлі проводиться перевірка всього ланцюжка блоків. Виявляється місце невідповідності в ланцюзі, надсилається запит на отримання вірних даних ланцюжка (рисунок 2.3).

Вузол, який перший вирахував хеш блоку і отримав підтвердження більшості вузлів, відправляє правильну ділянку ланцюга. Спотворена ділянка ланцюга перезаписується отриманими даними.

Якщо ж при перевірці ланцюжка невідповідностей виявлено не було, але стан ланцюга відрізняється від інших вузлів, то правильним вважається найдовший валідний ланцюжок таких блоків. При цьому виконується одна з вимог технології Blockchain, яка каже, що при будь-яких варіантах, ланцюжок блоків, який є найдовшим, приймається єдиним істинним для всієї мережі [13].

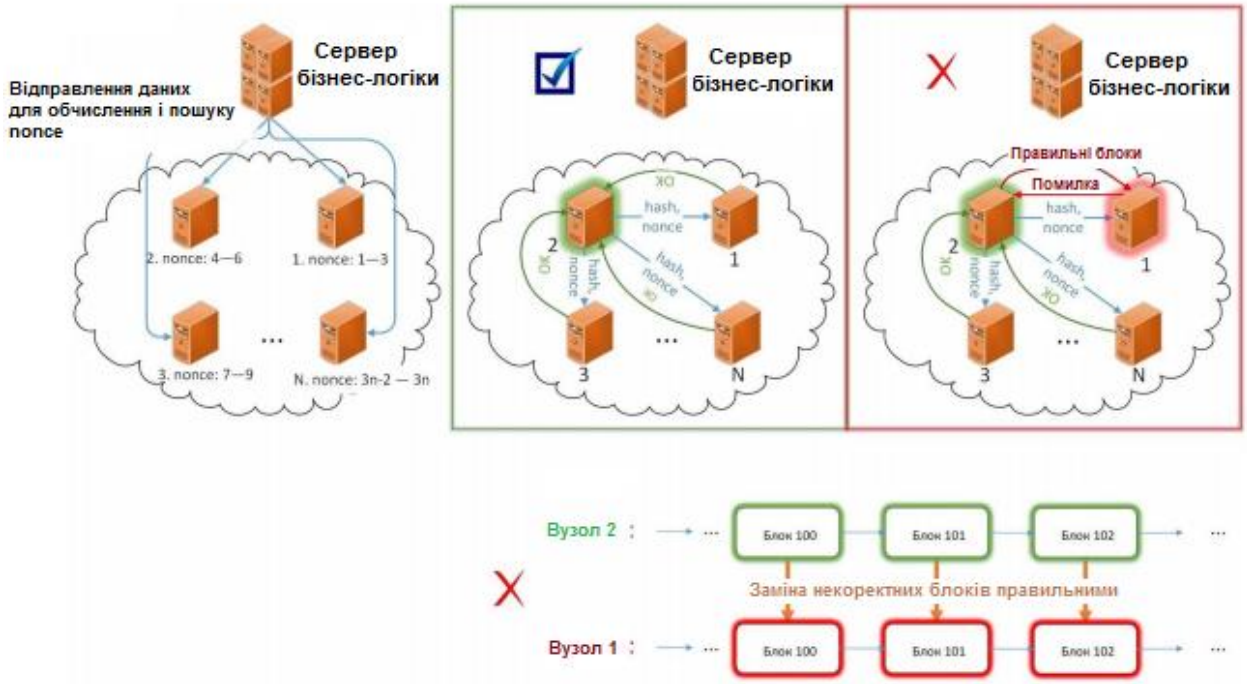

Рисунок 2.3 – Результати обчислення хешу блоку на вузлах

Вивантаження документа. Документ запитується за його ідентифікатором через браузер, запит відправляється на сервер бізнес-логіки. Далі запит з сервера бізнес-логіки потрапляє на захищений сервер даних, на якому відбувається пошук хешу цього документу. Знайдений хеш відправляється назад на сервер бізнес-логіки, а сервер бізнес-логіки віддає цей хеш всіх вузлів в мережі Blockchain. Після цього кожен вузол мережі Blockchain рахує свій ланцюжок з першого елемента до того блоку, в якому знаходиться хеш, який нас цікавить.

Після всіх успішних відповідей з усіх вузлів мережі Blockchain, сервер бізнес-логіки вважає, що файл надійно і конфіденційно збережений. Даний файл віддається користувачеві із захищеного сервера даних.

2.3 Обґрунтування вибору технологій

Наведемо основні технології, які використовуються в реалізації, та короткий опис їх можливостей. Вся інформація взята із офіційних сайтів розробників.

Spring Boot (https://spring.io/projects/spring-boot). ПЗ є універсальним фреймворком на відкритому вихідному коді для Java-платформи. У реалізації прототипу ЕА дана технологія дозволяє спростити конфігурацію проекту. За допомогою цього ПЗ можна досить легко створювати самостійні, виробничі додатки, які потім достатньо "просто запустити". Більшість програм Spring Boot потребують мінімальної конфігурації Spring.

Особливості:

– можливість створення окремих програм Spring;

– вбудовані Tomcat, Jetty або Undertow (не потрібно розгортати файли WAR);

- забезпечує можливості роботи для початківців, щоб спростити конфігурацію збірки;

– автоматичне налаштування Spring та сторонніх бібліотек;

– готові до використання функції, такі як метрики, перевірки працездатності та зовнішню конфігурацію;

– абсолютна відсутність генерації коду та відсутність вимог до конфігурації XML.

Spring Data JPA (https://spring.io/projects/spring-data-jpa). Технологія спрощує взаємодію з БД. В реалізації прототипу забезпечує ефективний доступ до даних. Впровадження рівня доступу до даних програми було досить тривалим часом. Це ПЗ забезпечує покращення реалізації рівнів доступу до даних, значно зменшивши затрачені зусилля. Властиво сам розробник програмує власні інтерфейси сховища, в тому числі і користувацькі методи пошуку, а Spring Data JPA автоматично забезпечить їх програмну реалізацію.

Особливості:

– сучасна підтримка процесу побудови сховищ даних на основі Spring та JPA;

– якісна підтримка предикатів Querydsl і, отже, безпечні для типу запити JPA;

– забезпечення прозорості аудиту класу домену;

– підтримка пагінації, реалізована можливість динамічного виконання запитів, функціональна інтеграція власного коду доступу до даних;

– підтримка відображення сутності з використанням мови XML;

– реалізація конфігурації сховища даних на основі JavaConfig.

PostgreSQL (https://www.postgresql.org/). Вільна СУБД, котра є за своєю суттю об'єктно-реляційною. В реалізації прототипу забезпечує ефективну організацію зберігання даних. ПЗ застосовує та розширює стандартну мову SQL разом із багатьма функціями, котрі дозволяють безпечне зберігання та масштабування найскладніших робочих навантажень даних. СУБД працює на всіх основних ОС, сумісна з ACID та має потужні доповнення.

PostgreSQL має безліч функцій, мета яких – допомога фахівцям у створенні програмних додатків, адміністраторам – у захисті цілісності даних та створенні відмовостійких середовищ, а також допомагати управляти даними безвідносно великим чи малим є їх набір. Окрім того, що PostgreSQL є безкоштовним та відкритим кодом, він дуже розширюваний. Наприклад, розробник може визначати персональні типи даних, кодувати власні функції, і навіть створювати код на різних мовах програмування, не перекомпілюючи БД.

Lombok (https://projectlombok.org/). Бібліотека, яка забезпечить читаність коду. Це плагін компілятора, який додає в Java нові «ключові слова» і перетворює анотації в Java-код, зменшуючи таким чином затрати на розробку і забезпечуючи деяку додаткову функціональність. Lombok перетворює анотації в вихідному коді в Java-оператори до того, як компілятор їх обробить: залежність lombok відсутня в run time, тому використання плагіна не збільшить розмір збірки.

ДСТУ 7564:2014 «Інформаційні технології. Криптографічний захист інформації. Функція хешування». Алгоритм хешування. Так як прототип ЕА може бути використаний в державних відомствах, для алгоритму хешування необхідно використати державний стандарт.

Swagger (https://swagger.io/tools/), StopLight (https://next.stoplight.io). Інструменти з відкритим кодом для зручного проектування та створення API

системи. Побудовані з урахуванням сучасного робочого процесу API, це ПЗ пропонує інтуїтивно зрозумілий інтерфейс та продумані функції, щоб забезпечити повний життєвий цикл розробки API. Swagger - це мова опису інтерфейсів, виражених за допомогою JSON. Swagger застосовується спільно із набором програмних інструментів для проектування, створення, документування та використання веб-служб. Swagger включає автоматизовану документацію, генерацію коду та тестування.

Серед інших сервісів Stoplight надає інструменти візуального моделювання для створення документації OpenAPI - без необхідності знати деталі специфікації OpenAPI або кодувати рядки по одному. Цей документ специфікації API може бути єдиним джерелом знань, який розширює можливості всього життєвого циклу API, від створення прототипу до тестування, розробки, документації, продажів і багато іншого. Інструменти візуального моделювання Stoplight усувають необхідність в ознайомленні з форматом специфікації OpenAPI. Не потрібно знати тип даних для кожної властивості, має властивість бути вкладеною чи визначеною безпосередньо і т. д.

Docker (https://www.docker.com/). ПЗ для автоматизації розгортання і управління додатками. Може бути інтегрований із різноманітними додатками. Docker має відкритий вихідний кодом, що значно спрощує створення контейнерів і додатків їх на основі. Кожен контейнер містить всі необхідні інструменти для якісної роботи програми, зокрема набір бібліотек, системні засоби, програмний код та середовище виконання.

При допомозі Docker можна швидко виконувати розгортання і масштабування додатків в будь-якому середовищі і бути впевненим в тому, що код буде працювати. Працює з різними ОС, в т.ч. Linux, Windows, MacOs.

2.4 Опис API системи

Розглянута концепція системи на Blockchain може застосовуватися двома способами:

– самостійний додаток без централізації;
– доповнення до звичайного централізованого ПЗ.

Розглянемо варіант самостійного додатку. У цьому випадку вся бізнес-логіка проходить безпосередню в розумному контракті, тобто всі актори звертаються напряму до контракту в Blockchain через клієнтський додаток (рис. 3.3).

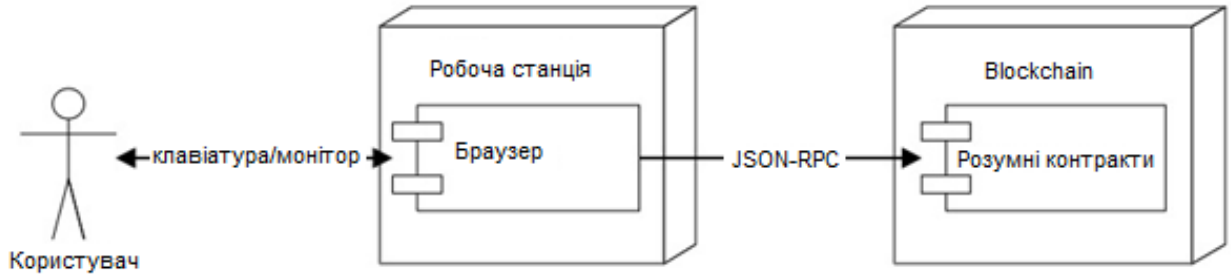

Рисунок 3.3 – Діаграма розгортання ПЗ системи з виконанням бізнес-логіки безпосередньо в розумних контрактах

Переваги такого підходу:

– функційна простота такої системи;
– достатньо легкий супровід технічного забезпечення;
– існування прозорої історії виконання всіх дій;
– можливість проводити роботу із системою без прив’язки до конкретного з місця.

Недоліки такого підходу є:

– необхідність плати за кожну операцію окремо;
– деяка складність додатків клієнтів;
– зростання часу відповіді від додатку.

У розроблюваній системі є три програмних інтерфейси, які відповідають наступним трьом компонентам:

– додаток з основною бізнес-логікою;
– додаток для зберігання документів з підвищеною захищеністю;

– додаток для вузлів зберігання даних в мережі Blockchain.

В наступному розділі буде наведено докладний опис та реалізацію кожного програмного інтерфейсу.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ

3.1 API додатка основної бізнес-логіки

На рисунку 3.1 перераховано список всіх методів розглянутого API.

Рисунок 3.1 – Список методів API додатка основної бізнес-логіки

Розглянемо кожен метод даного API:

1. Завантаження файлу - метод POST "/ uploadFile.

Програмний опис цього методу показано на рисунку 3.2.

Реалізація методу:

– за допомогою POST-методу "/ uploadFile" приймається завантажений файл по HTTP;

– обчислюється хеш отриманого файлу;

– хеш і дані файла відправляються на захищений сервер даних за допомогою HTTP POST методу "/ uploadFile". Його опис показано в п.3.2 на рисунку 3.9;

– id файлу збереженого в сховищі приходить як відповідь методу;

– хеш і отриманий зі сховища даних id файлу відправляються на всі вузли мережі Blockchain з використанням HTTP POST методу "/ Receive-file-

info". Його опис представлено в API для кожного вузла в мережі Blockchain (п. 3.3) на рисунку 3.16;

– метод повертає статус успішного завантаження файлу для відображення в браузері стану завантаження файлу.

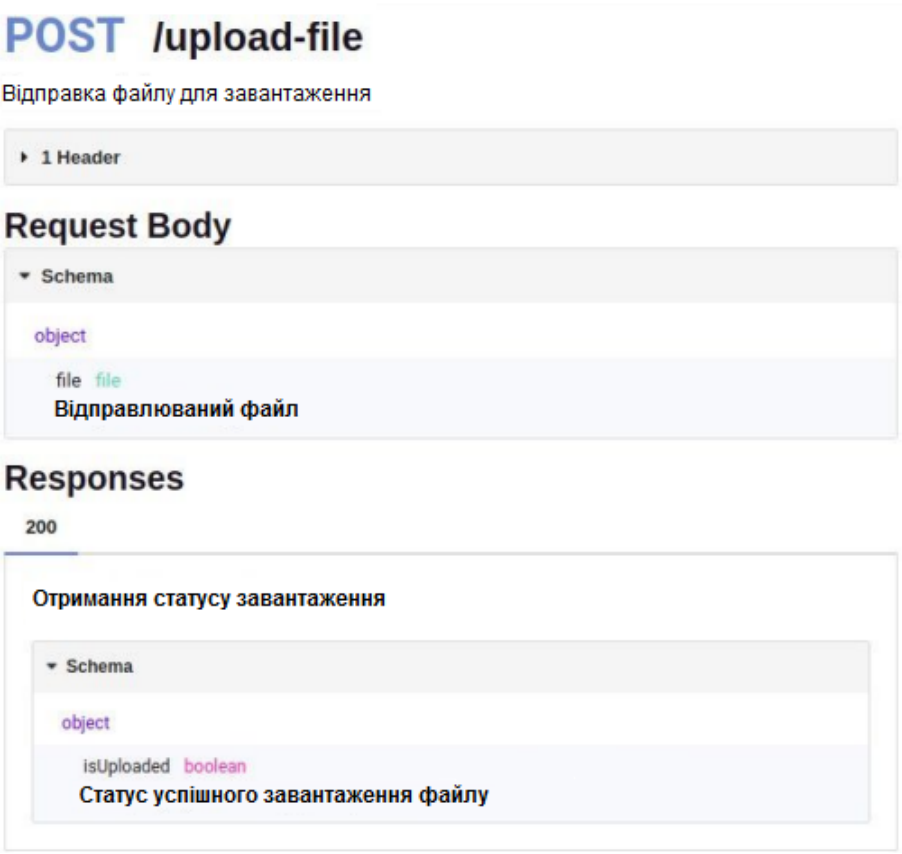

Рисунок 3.2 – Опис методу завантаження файлу

2. Отримання nonce - метод GET "/ nonces":

Опис даного методу представлено на рисунку 3.3.

Реалізація методу:

– при відправленні запиту за допомогою HTTP GET методу "/ nonces" вузол мережі Blockchain вказує свій ідентифікатор і ідентифікатор блоку, для якого необхідно отримати діапазон nonce для майнінга;

– для отриманих ідентифікаторів вузла і блоку послідовно віддаються діапазони nonce. Таким чином вузол, який закінчив майнінг для діапазону nonce може звернутися за наступним діапазоном;

– метод повертає початок і кінець діапазону nonce.

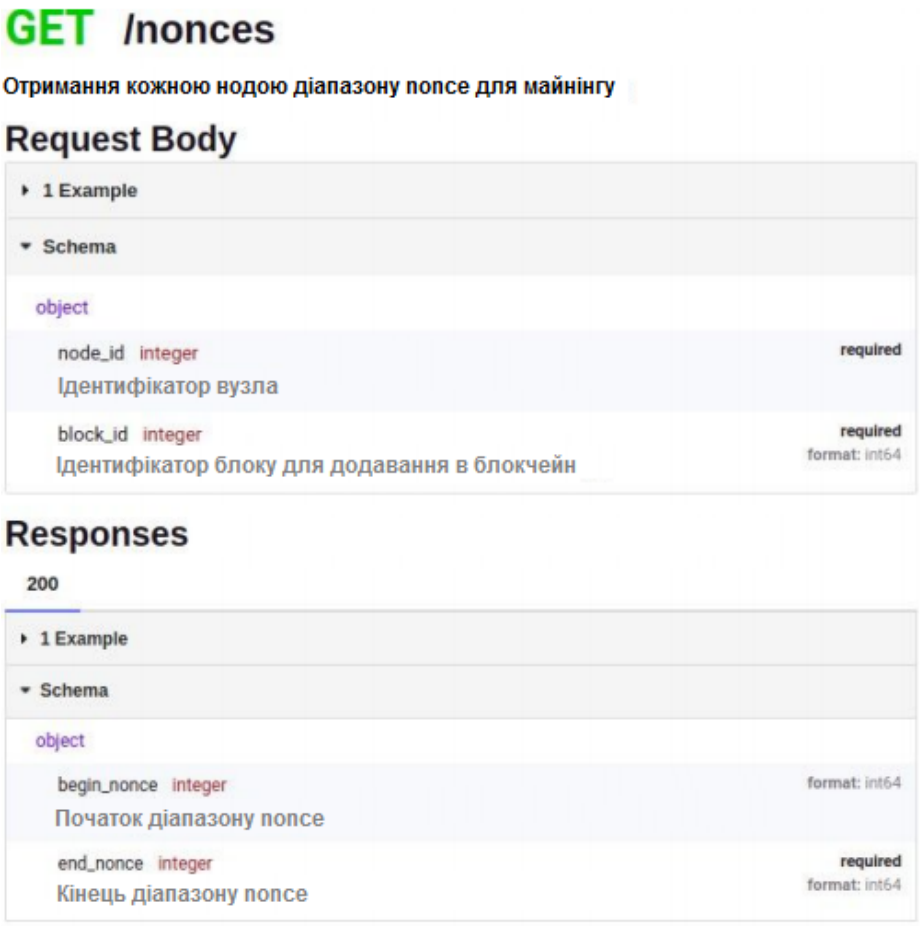

Рисунок 3.3 – Опис методу отримання nonce

3. Отримання результатів майнінгу - метод POST "/ receive-mining-result": Метод описано на рисунку 3.4.

Реалізація методу:

– з допомогою HTTP POST методу "/ receive-mining-result" приймаються номер блоку доданого в Blockchain і список ідентифікаторів файлів в ньому.

– отримані дані відправляються на сховище документів з допомогою HTTP POST методу "/ set-blocks". Його опис представлено в API для зберігання документів з підвищеною захищеністю (п. 3.2) на рисунку 3.11.

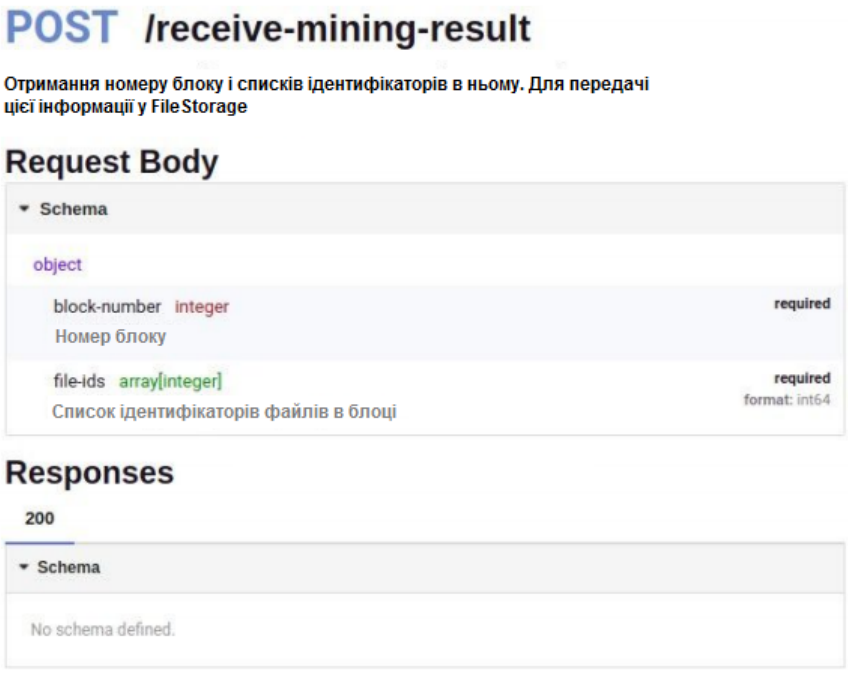

Рисунок 3.4 – Опис методу отримання результатів майнінгу

4. Прийняття результатів перевірки вузлів - метод POST "/ self-check-result":

Опис даного методу представлено на рисунку 3.5.

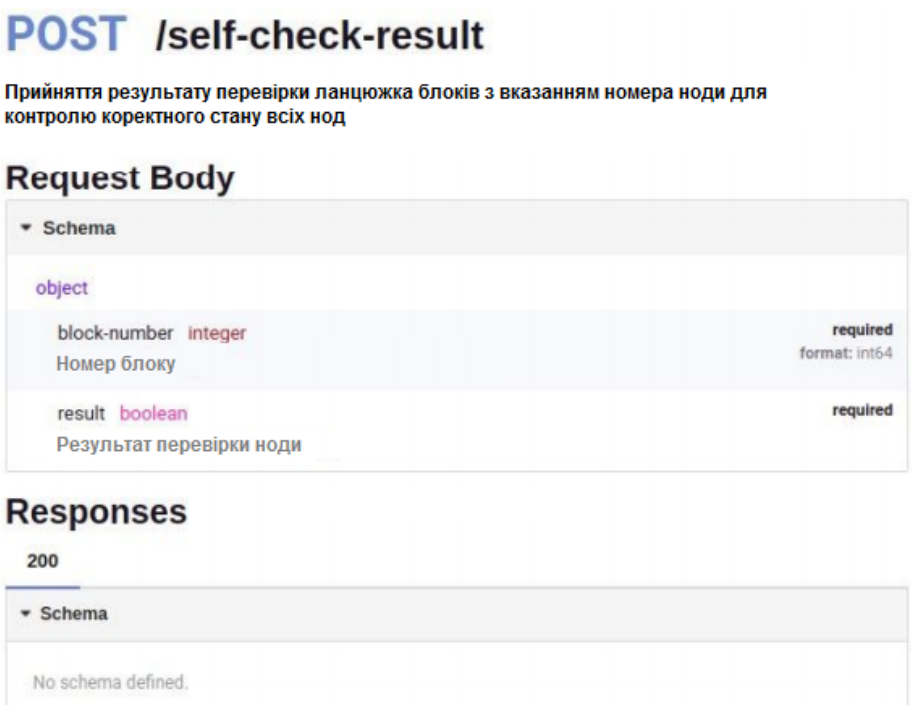

Рисунок 3.5 – Опис методу прийняття результатів перевірки вузлів

Реалізація методу:

– після прийняття результату перевірки вузла відбувається журналювання результатів перевірки на коректність всього ланцюжка блоків. Якщо помилка перевірки сталася на більшості вузлів, відбувається журналювання помилки критичного рівня.

5. Отримання всіх файлів - метод GET "/ files":

На рисунку 3.6 показано опис цього методу.

Реалізація методу:

– під час відправлення запиту із застосуванням HTTP GET методу "/ files" додаток основної бізнес-логіки перенаправляє запит на сховище даних з використанням HTTP GET методу "/ files". Він описаний в п.3.2 на рисунку 3.12;

– метод повертає список, в якому містяться ідентифікатори і імена файлів.

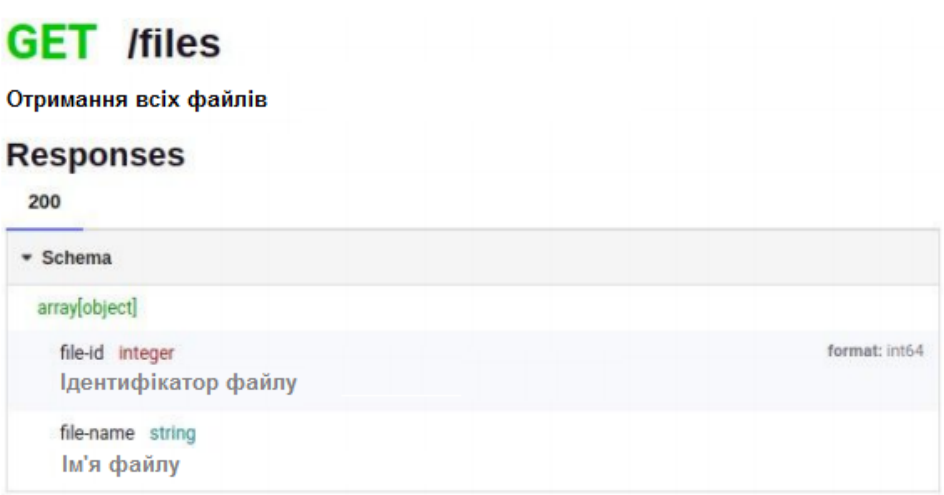

Рисунок 3.6 – Опис методу отримання всіх файлів

6. Отримання файлу по його id - метод GET "/ files / {file-id}":

Опис методу наведено на рисунку 3.7.

Рисунок 3.7 – Метод отримання файлу з id

Реалізація методу:

– при відправленні запиту за допомогою HTTP GET методу з браузера приходить ідентифікатор запитуваного файлу;

– отриманий ідентифікатор файлу відправляється на сховище документів за допомогою HTTP GET методу "/ files / {file-id}". Його опис наведено у п.3.2 на рисунку 3.13. Даний метод повертає номер блоку, в якому знаходиться запитуваний файл;

– отриманий номер блоку відправляється на кожен вузол мережі Blockchain з допомогою HTTP POST методу "/ Self-check / {block-number}", де відбувається перевірка до заданого номера блоку. Опис даного методу показано у п.3.3 на рисунку 3.21;

– після успішної перевірки на кожному вузлі мережі Blockchain файл віддається користувачеві для скачування.

3.2 API додатка для зберігання документів з підвищеною захищеністю

На рисунку 3.8 перераховано список всіх методів розглянутого API.

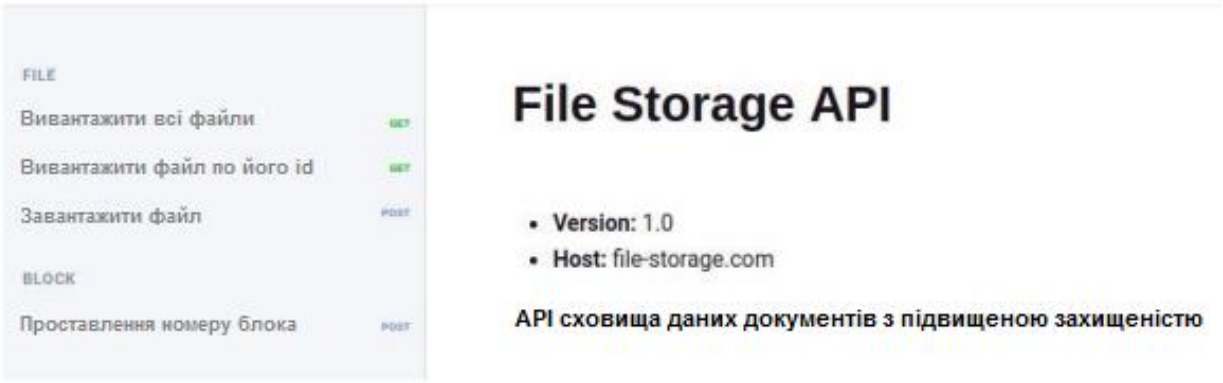

Рисунок 3.8 – Список методів API додатка для зберігання документів з підвищеної захищеністю

Розглянемо окремо кожен метод даного API для пояснення реалізації та роботи:

1. Завантаження файлу - метод POST "/ uploadFile":

Програмна складова для цього методу показана на рисунку 3.9.

Рисунок 3.9 – Опис методу завантаження файлу

Метод реалізовано так:

– з використанням HTTP POST методу "/ uploadFile" приймаються дані файлу;

– отриманий файл зберігається в БД (див. Рис. 3.10);

– метод повертає ідентифікатор файлу після його збереження.

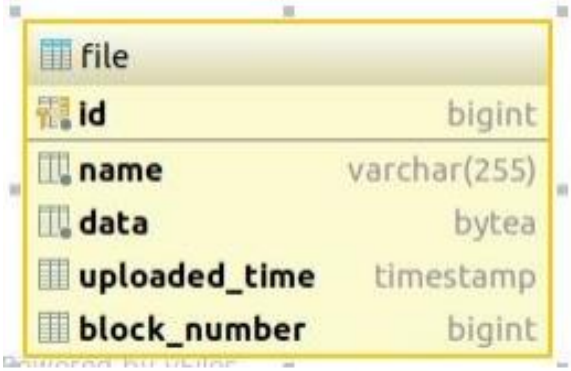

Рисунок 3.10 – Схема БД сховища документів

2. Проставлення номера блоку - метод POST "/ set-blocks". На рисунку 3.11 наведено опис методу.

Реалізація методу:

– даний метод викликається після успішного додавання блоку в Blockchain;

– при застосуванні HTTP POST методу "/ set-blocks" приймаються номер блоку і список ідентифікаторів файлів в ньому;

– в БД відбувається оновлення записів з відповідними ідентифікаторами, даним записам проставляється номер блоку, котрий прийшов.

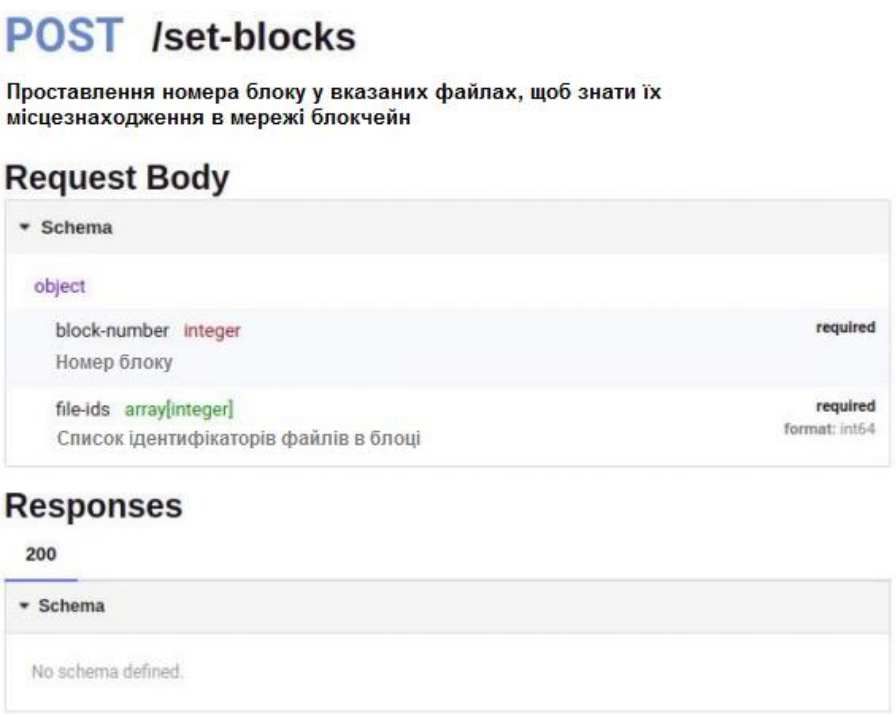

Рисунок 3.11 – Опис методу проставлення номера блоку

3. Отримання всіх файлів - метод GET "/ files":

Цей метод описано на рисунку 3.12.

Варто навести особливості реалізації методу:

– при відправленні запиту при допомозі HTTP GET методу "/ files" сховище документів витягує з БД всі файли;

– метод повертає список, в якому містяться ідентифікатори і імена файлів.

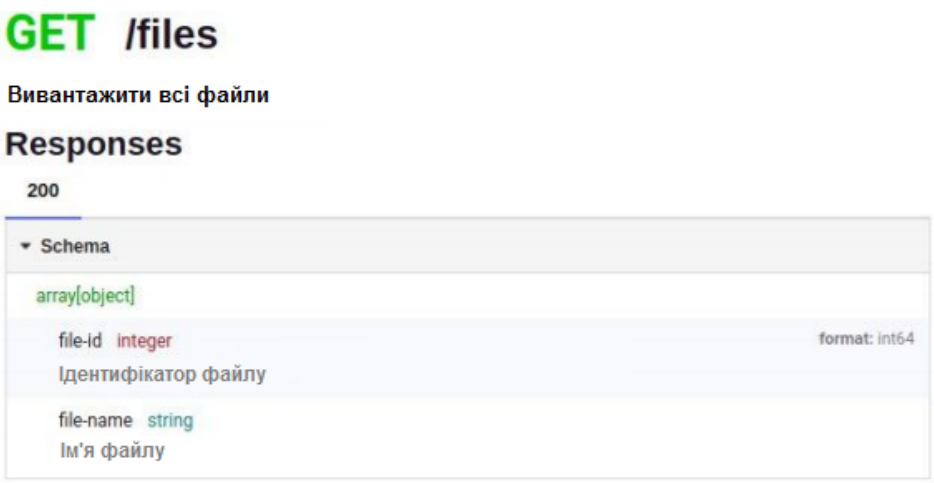

Рисунок 3.12 – Опис методу отримання всіх файлів

4. Отримання файлу по його id - метод GET "/ files / {file-id}":

Описано цей метод на рисунку 3.13.

Основні етапи реалізації методу:

– під час надсилання запиту з застосуванням HTTP GET методу з додатку основної бізнес-логіки приходить ідентифікатор запитуваного файлу;

– відбувається пошук за ідентифікатором в БД;

– метод повертає дані файлу і номер блоку, в якому він знаходиться.

Рисунок 3.13 – Опис методу отримання файлу з id

3.3 API додатка для кожного вузла в мережі Blockchain

На рисунку 3.14 показано API Blockchain.

```
/**
 * Сервис для роботи з даними в Blockchain
 */
public interface BlockchainDataService {

    /**
     * Поміщаємо файл в транзакцію, дальше в блок, потім в Blockchain
     * @param fileData данні файла
     * @param userId ідентифікатор користувача
     * @param uploadDateTime час завантаження файлу в мсек
     * @return boolean чи завантажений файл в Blockchain успішно
     */
    boolean placeToBlockchain(byte[] fileData, String userId, long uploadDateTime);
}
```

Рисунок 3.14 – API Blockchain

На рисунку 3.15 наведено список всіх методів розглянутого API.

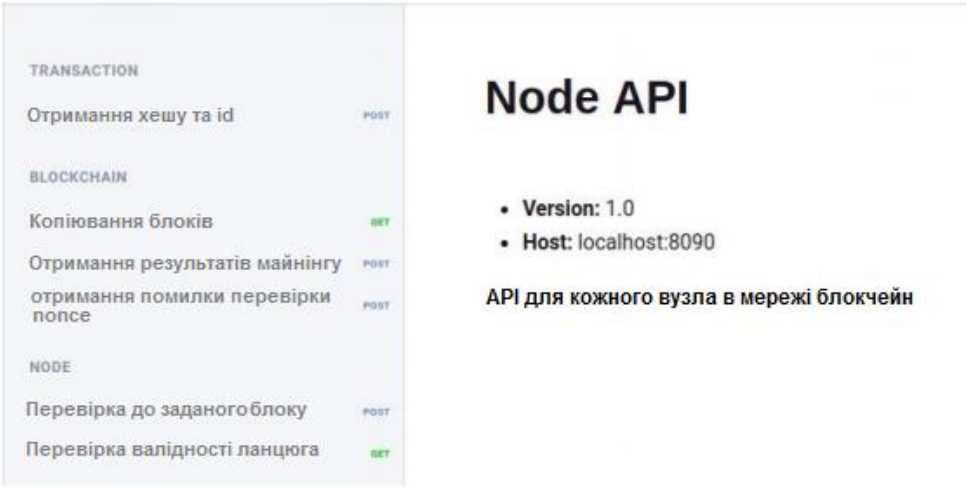

Рисунок 3.15 – Список методів API для кожного вузла мережі Blockchain

Розглянемо кожен метод даного API:

1. Отримання хешу і id файлу - метод POST "/ receive-file-info":

Опис даного методу представлено на рисунку 3.16.

Рисунок 3.16 – Опис методу отримання хешу і id файлу

Реалізація методу:

– за допомогою HTTP POST методу "/ receive-file-info" приймаються ідентифікатор, хеш і час завантаження файлу;

– додавання файлу в блок;

– створюється транзакція на основі отриманої інформації про файли. Транзакція додається в блок з черги блоків на майнінг;

– Перевіряється, чи потрібно закрити блок в разі його переповнення або в разі тривалої часової паузи;

– якщо потрібно закрити блок, то спочатку поточний блок додається в Blockchain, а потім створюється новий блок для подальшого додавання транзакцій в уже новий поточний блок.

– додавання блоку в Blockchain;

– відбувається майнінг блоку - обчислюємо його хеш із заданою складністю;

– вузол запитує сервер основної бізнес логіки діапазон nonce для майнінгу за допомогою HTTP GET методу "/ nonces" у п. 31. на рис. 3.3;

– якщо приходить інформація про вже замайнений блок за допомогою HTTP POST методу "/ Receive-mined-block-info" (див. рис. 3.18), то перевіряється валідність інформації, яка прийшла;

– якщо вузол перевірив весь діапазон nonce, то він знову запитує його за методом "/ nonces" поки не знайде шуканий хеш із заданою складністю;

– якщо вузол знайшов шуканий хеш із заданою складністю, то він відправляє його решту вузлам мережі Blockchain за допомогою HTTP POST методу "/ receive-mined-block-info" (див. рис. 3.18);

– замайненний блок додається до ланцюжку існуючих блоків;

– блок зберігається в БД (див. Рис. 3.17);

– усі транзакції блоку зберігаються в БД.

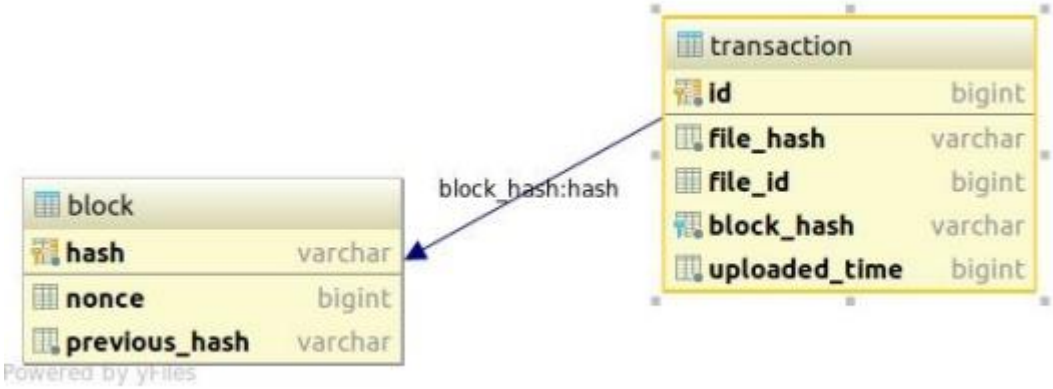

Рисунок 3.17 – Схема БД вузлів мережі Blockchain

2. Отримання результату майнінгу – метод POST "/ Receive-mined-block-info".

Опис даного методу наведено на рисунку 3.18.

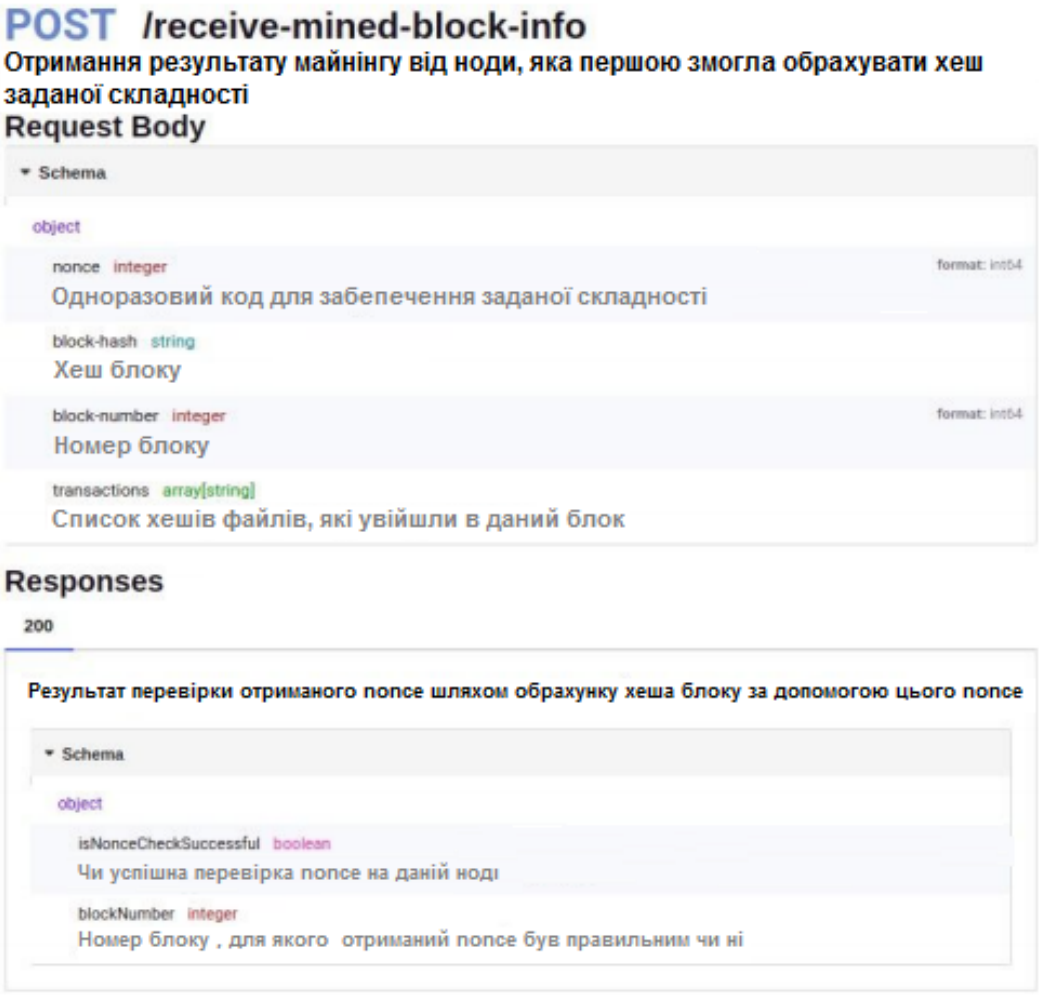

Рисунок 3.18 – Опис методу отримання результату Майнінг

Реалізація методу:

– за допомогою HTTP POST методу "/ receive-mined-block-info" приймаються: nonce, хеш блоку, номер блоку, список транзакцій;

– отримані дані перевіряються - вираховується хеш блоку на основі отриманого nonce і історії ланцюга, яка зберігається на вузлі, на який прийшла інформація про майнінг;

– якщо отриманий хеш збігається з прийшли хешем блоку, то перевірка пройшла успішно;

– блок зберігається в БД;

– якщо на вузлі, який прийняв інформацію про майнінг від іншого вузла, паралельно відбувався майнінг, то він його припиняє;

– якщо вирахуваний хеш відрізняється від отриманого хешу блоку, то на всіх вузлах запускається перевірка за допомогою методу HTTP GET методу "/ self-check" (див. рис. 3.20).

3. Отримання помилки перевірки nonce - метод POST "/ receive-nonce-error":

Даний метод описано на рисунку 3.19.

Реалізація методу аналогічна методу отримання результату майнінгу - метод POST "/ receive-mined-block-info". Метод викликається в разі закінчення часу відповіді методу "/ receive-mined-block-info".

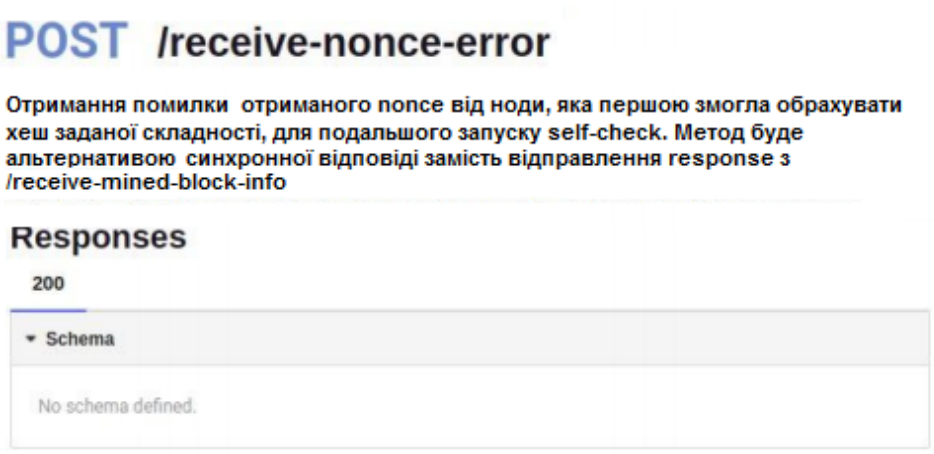

Рисунок 3.19 – Опис методу отримання помилки перевірки nonce

4. Перевірка валідності ланцюга - метод GET "/ self-check". Опис даного методу представлено на рисунку 3.20.

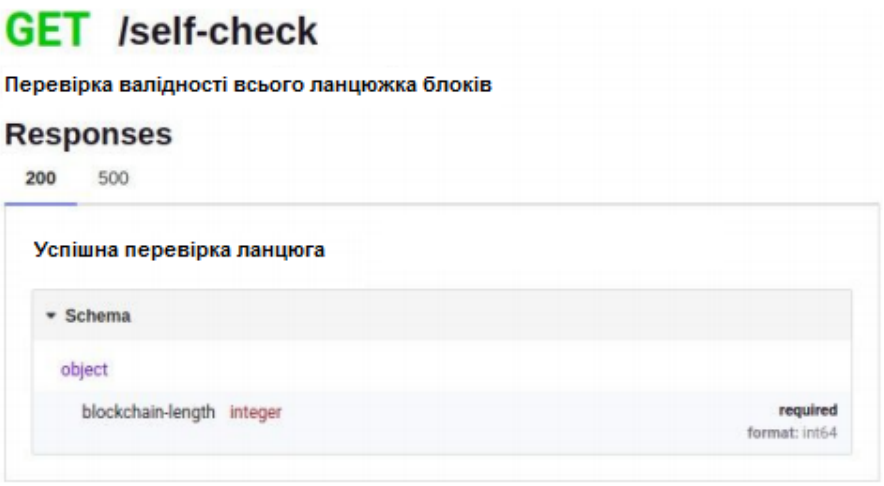

Рисунок 3.20 – Метод перевірки валідності ланцюжка

Реалізація методу:

– за допомогою HTTP GET методу "/ self-check" приходить запит на перевірку валідності всьому ланцюгу Blockchain;

– для всього ланцюжка заново послідовно відбувається обрахунок хешв всіхі блоків із використанням раніше збереженого nonce;

– якщо перевіркою виявлено невідповідність, тоді вузол очікує на відповідь методу "/ self-check" (див. рис. 3.20) від усіх інших вузлів і у вузла з макисмальною довжиною ланцюжка запрошує коректні блоки за допомогою HTTP GET методу "/ Copy-blocks" (див. рис. 3.22);

– метод повертає результат успішності перевірки і довжину свого ланцюжка.

5. Перевірка валідності до заданого блоку - метод POST "/ Self-check / {block-number}".

Програмний код цього методу показано на рисунку 3.21.

Реалізація методу:

– за допомогою HTTP POST методу "/ self-check / {block-number}" відбувається перевірка ланцюжка блоків до зазначеного номера блоку: послідовно вираховуються хеші блоків за допомогою вже збереженого nonce;

– якщо під час перевірки було виявлено невідповідність, то вузол чекає відповіді методу "/ self-check" (див. рис. 3.20) від інших вузлів і у вузла з найбільшою довжиною ланцюжка запрошувати коректні блоки за допомогою HTTP GET методу "/ Copy-blocks" (див. рис. 3.22);

– перевіряється наявність хешу файлу, який прийшов, в заданому блоці;

– метод повертає результат перевірки.

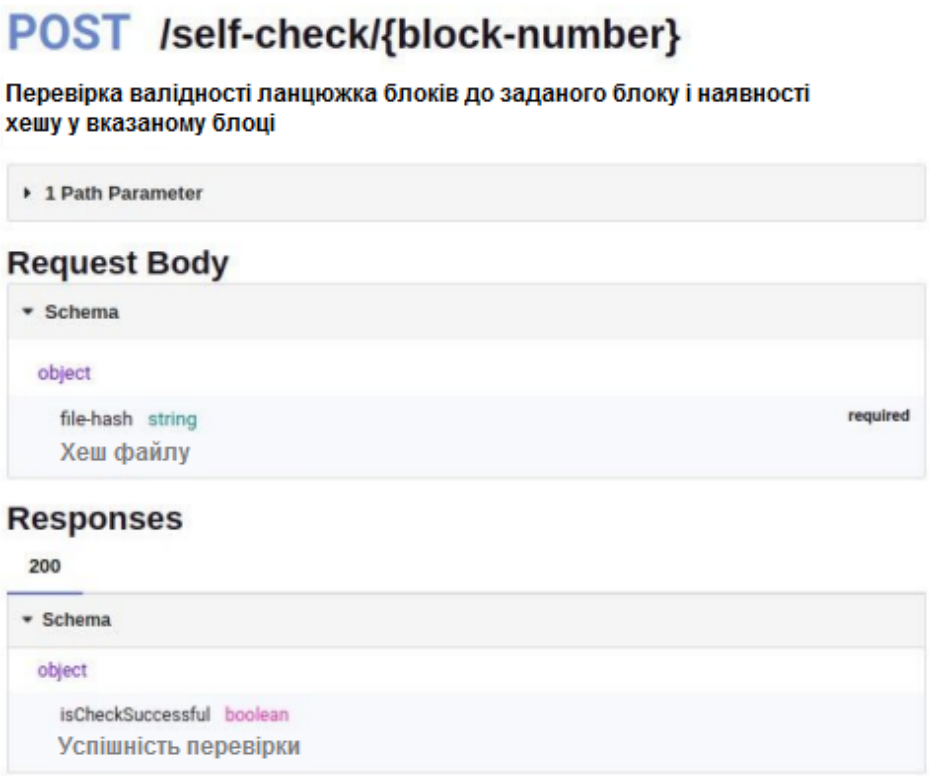

Рисунок 3.21 – Опис методу перевірки валідності ланцюга до заданого блоку

6. Копіювання блоків - метод GET "/ copy-blocks". Метод описано на рисунку 3.22.

Реалізація методу:

– при відправленні запиту за допомогою HTTP GET методу "/ Copy-blocks" приймається номер блоку, починаючи з якого потрібно передати коректні блоки;

– відбувається пошук потрібних блоків в БД;

– метод повертає знайдений список блоків, починаючи з заданого номера блоку.

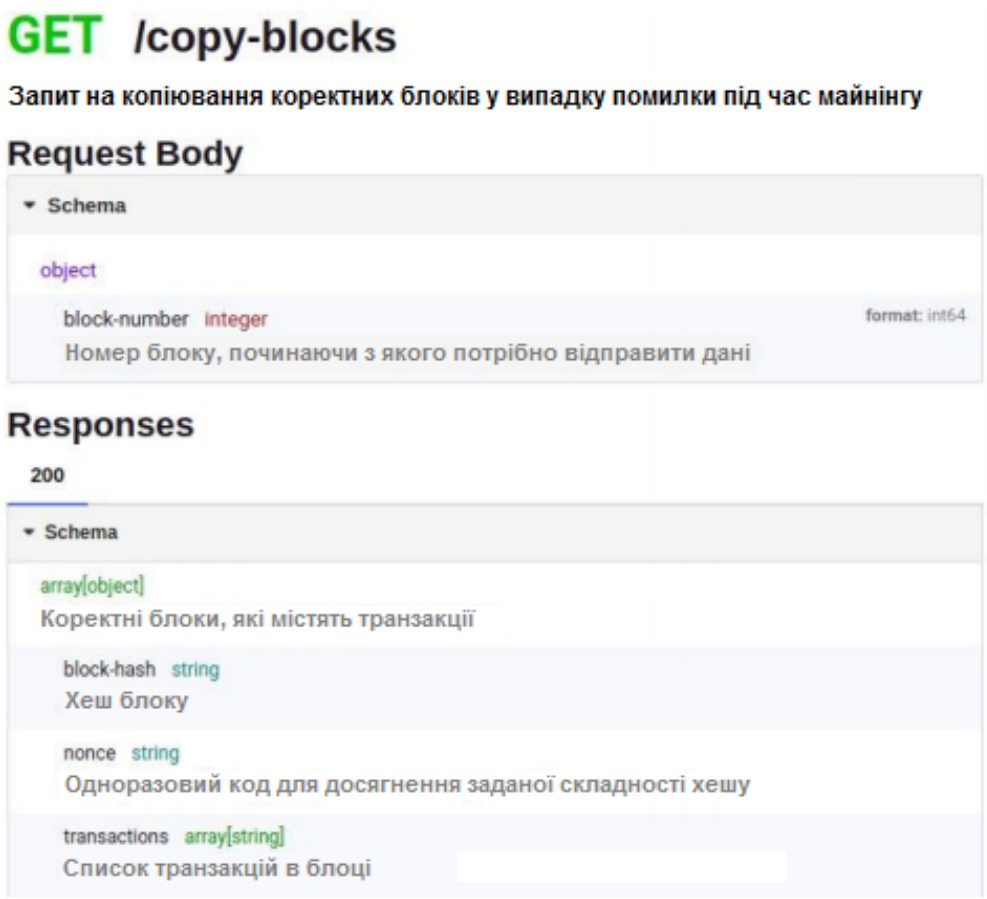

Рисунок 3.22 – Опис методу копіювання блоків

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Навчання працюючих і інструктажі з охорони праці

Однією із складових ефективної роботи з профілактики виробничого травматизму є належна підготовка, навчання та підвищення кваліфікації працівників з питань охорони праці. Загальний порядок проведення навчання з питань охорони праці встановлений Законом України «Про охорону праці» (ст. 18. «Навчання з питань охорони праці»).

Виконання вимог Закону України «Про охорону праці» в частині проведення навчання та перевірки знань з питань охорони праці здійснюється відповідно до Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держкомітету України з нагляду за охороною праці 26 січня 2005 р. № 15 (далі — Типове положення).

Нагляд за дотриманням вимог Типового положення здійснюють органи державного нагляду за охороною праці, а координацію та методичний супровід — Головний навчально-методичний центр та навчальні підрозділи експертно-технічних центрів Держгірпромнагляду.

Вивчення предмета «Охорона праці» при підготовці, перепідготовці та підвищенні кваліфікації працівників, які залучаються до виконання робіт з підвищеною небезпекою, на підприємстві регламентується п. 2.3 Типового положення. На підприємствах згідно з п. 1.1 Додатку 3 Типового положення навчання та перевірку знань з питань охорони праці повинні проходити керівники, заступники керівників, головні спеціалісти, керівники основних виробничих та технічних служб, безпосередньо пов’язані з організацією безпечного ведення робіт. Крім цього, згідно з п. 5 Додатку 3, навчання та перевірку знань з питань охорони праці мають проходити керівники, спеціалісти служб охорони праці, члени комісій з перевірки знань з питань охорони праці, особи, відповідальні за технічний стан і безпечну експлуатацію об’єктів підвищеної небезпеки підприємств.

Типове положення встановлює порядок та місце проведення навчання та перевірки знань з питань охорони праці посадових осіб (п. 5.2 та п. 5.3). Посадові особи, перелік яких наведено в п. 5.2, проходять навчання у Головному навчально-методичному центрі Держнаглядохоронпраці. Перевірка знань цієї категорії посадових осіб проводиться комісією, створеною наказом Держгірпромнагляду.

Організацію навчання та перевірки знань з питань охорони праці працівників на підприємстві здійснюють працівники служби кадрів або інші спеціалісти, яким роботодавець доручив організацію цієї роботи. Навчання та перевірка знань з питань охорони праці працівників (виконавців і посадових осіб), які не залучаються до виконання робіт підвищеної небезпеки, проводиться не рідше ніж один раз на три роки. Посадові особи та працівники, які виконують роботи підвищеної небезпеки, проходять спеціальне навчання та перевірку знань відповідних нормативно-правових актів з охорони праці не рідше одного разу на рік.

Посадові особи малих підприємств (п. 5.4), які не мають можливості створити власні комісії з перевірки знань з питань охорони праці та провести навчання з питань охорони праці, проходять навчання та перевірку знань в навчальних закладах, які мають відповідний дозвіл.

Спеціальне навчання з питань охорони праці може проводитись безпосередньо на підприємстві або навчальним закладом, який має відповідний дозвіл. При проведенні такого навчання на підприємстві навчальні плани та програми розробляються з урахуванням конкретних видів робіт, виробничих умов і функціональних обов’язків працівників і затверджуються наказом керівника підприємства.

Періодичність інструктажів, навчання та перевірки знань з питань охорони праці залежить від видів виконуваних робіт та встановлюється Типовим положенням. Перевірка знань з питань охорони праці після проведення спеціального навчання проводиться комісією підприємства.

Якщо на підприємстві неможливо створити комісію з перевірки знань з питань охорони праці (п. 4.4 Типового положення), перевірка знань проводиться комісією спорідненого підприємства або Теруправління Держгірпромнагляду.

Всі працівники та посадові особи підприємства, включаючи посадових осіб, відповідальних за виконання робіт підвищеної небезпеки (крім зазначених в п. 5.2 та п. 5.3 Типового положення), проходять навчання та перевірку знань з питань охорони праці на підприємстві. Типове положення не зобов'язує, але й не забороняє проводити навчання всіх виконавців та посадових осіб (особливо тих, що виконують роботи підвищеної небезпеки) в навчальних закладах. У нашій країні є багато підприємств, де таке навчання проводиться, і це має позитивні наслідки. Ті витрати, які при цьому несуть підприємства, перекриваються створенням більш безпечних умов праці і в результаті збереженням життя та здоров'я працівників.

Також в навчальних закладах проходять навчання та перевірку знань із загальних питань охорони праці всі посадові особи та фахівці, які проводять інструктажі або навчання підлеглих працівників з питань охорони праці, виконують роботи з проектування об'єктів, а також інші працівники, незалежно від того, передбачено таке навчання Типовим положенням чи ні.

4.2 Заходи захисту від випромінювань оптичного діапазону

До випромінювання оптичного діапазону відносяться інфрачервоні й ультрафіолетові хвилі, видиме світло, лазерне випромінювання.

По фізичній природі інфрачервоні промені мають хвильові (довжина хвилі 0,78-540 мкм) і квантові властивості. Генератором випромінювання є будь-яке тіло, температура якого вище абсолютного нуля. За законом Стефана-Больцмана інтегральна густина випромінювання, Вт/м^2, абсолютно чорного тіла пропорційна четвертому ступеню його абсолютної температури [17].

З підвищенням температури тіла змінюється спектральний склад його випромінювання. Чим вища температура тіла, тим коротша довжина хвилі, максимального випромінювання.

Інфрачервона енергія, яка потрапляє на тіло людини, діє передусім на незахищені його частини (лице, руки, шию, груди), причому конвективне тепло впливає на зовнішній шкіряний покрив, тоді як інфрачервоне випромінювання може проникнути на деяку глибину в тканину. При довготривалому перебуванні людини в зоні інфрачервоного випромінювання, як і при систематичній високій температурі наступає різке порушення теплового балансу в організмі. Для вимірювання густини потоку випромінювання на робочому місці застосовують актинометр – прилад, який дозволяє вимірювати густину потоку інфрачервоного випромінювання у діапазоні від 0 до 14кВт/м^2 . Основні види захисту від інфрачервоного випромінювання–захист часом, захист віддалю, усунення джерела тепловиділення, теплоізоляція, охолодження гарячої поверхні, забезпечення тепловіддачі тіла людини та індивідуальні засоби захисту. Потужність випромінювання можна знизити за рахунок конструкторських і технологічних рішень (зміною нагрівання виробів у нагрівальних пічках індукційним нагріванням та ін.) і за рахунок покриття поверхні, яка нагрівається, тепло ізолювальним матеріалом. Для захисту очей застосовують світлофільтри зі спеціального жовто-зеленого або синього скла.

Ультрафіолетове випромінювання змінює склад виробничої атмосфери. Утворюється озон, оксиди азоту і пероксид водню. Короткохвильове випромінювання іонізує повітря, утворює в атмосфері ядра конденсації, які зменшують освітленість робочих місць і призводять до утворення туманів.

Основні засоби захисту. Першочергові заходи – це конструкторські і технологічні рішення, які виключають генерацію або понижують інтенсивність випромінювання. Спеціальні засоби захисту (екранування джерел випромінювання, фарбування стін у світлі кольори) попереджують розповсюдження і знижують інтенсивність цих випромінювань у виробничих приміщеннях. Очі захищають окулярами або щитками зі склом – світлофільтром.

Для захисту шкіри використовують мазі з речовинами – світлофільтрами для цих променів (салол, саліцилово-метиловий ефір та ін.), а також спецодяг з бавовняних тканин і грубововняного сукна. Руки захищають рукавицями [18].

Діапазон довжин хвиль які випромінюють оптичні квантові генератори (ОКГ) – лазери, охоплює видимий спектр і розповсюджується в інфрачервоній і ультрафіолетовій областях.

Найбільш чутливими до дії випромінювання ОКГ є очі. Випромінювання викликають опіки і пошкодження сітківки ока, це може призвести до сліпоти. Небезпечне не тільки пряме випромінювання, але й відбите від стін, обладнання.

Існують спеціальні норми, до яких ввійшли організаційні та інженерно-технічні заходи, які можуть забезпечити зменшення густини потоків енергії (потужності) на робочих місцях до величин, значно менших від допустимих. ОКГ розміщують в окремих або відгороджених приміщеннях. Саме приміщення і обладнання не повинні мати дзеркальної поверхні. Стіни, стелі, обладнання й інші предмети фарбують матовою фарбою з малою сорбційною здатністю. Приміщення повинно мати високу освітленість, а також припливно-витяжну вентиляцію. При розміщенні в одному приміщенні декількох ОКГ їх огороджують ширмами, шторами або екранами, що не пропускають випромінювання. Надійним захистом від випадкового попадання випромінювання на людину є світловод, який екранує промінь на усьому шляху його дії (від ОКГ до мішені) [18].

ВИСНОВКИ

Робота присвячена актуальній технічної задачі реалізації ЕА, використовуючи технологію Blockchain. В рамках даної роботи виконано:

– досліджено способи зберігання документів тривалого терміну зберігання в архіві;

– досліджено поняття ЕА, проблеми ЕА;

– досліджено способи забезпечення достовірності документів тривалого терміну зберігання в ЕА;

– розглянуто існуючі реалізації системи забезпечення справжності ЕА на основі технології Blockchain;

– реалізовано прототип системи ЕА, який використовує технологію Blockchain для вирішення проблеми автентичності документів тривалого терміну зберігання.

Надалі розроблений прототип системи захисту ЕА буде використаний як основа виробничого рішення.

Можливі шляхи удосконалення розробки.

1. Додати функціонал для роботи з версіями документами.

2. Реалізувати функціонал для зберігання ланцюжка блоків на комп'ютерах користувачів для резервної копії даних.

3. Додатки будуть перенесені на більш досконалу мікросервісну модель за рахунок підключення функціоналу з програмної бібліотеки Spring Cloud.

4. Збільшити швидкодію системи.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Алексєєва Е. В. Архівознавство: М.: Видавничий центр «Академія», 2004. — 276 с.
2. Державна архівна служба України [Електронний ресурс] // Офіційний вебпортал органу виконавчої влади. - Режим доступу до ресурсу: https://archives.gov.ua/ua/ - (дата звертання: 02.04.2021).
3. Кодекс законів про працю України. [Електронний ресурс] // Кодекс законів про працю України. - Режим доступу до ресурсу: https://zakon.rada.gov.ua/laws/show/322-08/stru - (дата звертання: 02.04.2021).
4. Державна служба статистики України [Електронний ресурс] // Офіційний веб-сайт. - Режим доступу до ресурсу: http://www.ukrstat.gov.ua/ - (дата звертання: 12.04.2021).
5. Мельник Надія, Марковець Олександр. Електронні архіви, особливості їх функціонування [Електронний ресурс] // Режим доступу до ресурсу: http://ena.lp.edu.ua/bitstream/ntb/33199/1/061-136-137.pdf - (дата звертання: 16.04.2021).
6. Марченко П. М. Шляхи вирішення проблеми зберігання електронних документів в архіві (аналіз Інтернет-ресурсів). Студії з архівної справи та документознавства. 2004. Т. 12. С. 81–84.
7. Закон України «Про електронні довірчі послуги» [Електронний ресурс] // Режим доступу до ресурсу: https://zakon.rada.gov.ua/laws/show/2155-19#n294 – (дата звертання: 26.04.2021).
8. Why is Certificate Expiration Necessary? [Електронний ресурс]. - Режим доступа: https://www.entrustdatacard.com/blog/2016/october/why-is-certificate-expirati on-necessary - (дата звертання: 26.04.2021).
9. How it all works. [Електронний ресурс]. - Режим доступа: https://blocksign.com/about - (дата звертання: 08.05.2021).

10. Генкин А. Blockchain. Как это работает и что ждет нас завтра / Генкин А., Михеев А. – Москва: «Альпина Диджитал», 2017. – 680 с.

11. Melanie Swan. Blockchain: Blueprint for a New Economy. — Sebastopol: «O'Reilly Media, Inc.», 2015 – 152 с.

12. Биткойн для чайников. — М.: «Вильямс», 2017. – 3036 с.

13. Тапскотт А. Технология Blockchain - то, что движет финансовой революцией сегодня.– М.: «Эксмо», 2017 – 448 с.

14. Скиннер К. ValueWeb. Как финтех-компании используют Blockchain и мобильные технологии для создания интернета. – М.: «Манн, Иванов и Фербер», 2017 – 446 с.

15. Могайар У. Blockchain для бизнеса. / Могайар У., Бутерин В. – Москва: «Эксмо», 2017 – 224 с.

16. Wattenhofer R. The Science of the Blockchain. / Roger Wattenhofer – North Charleston: «CreateSpace Independent Publishing Platform», 2016 – 124 с.

17. Толок А.О. Крюковська О.А. Безпека життєдіяльності: Навч. посібник. – 2011. – 215 с.

18. Яремко З. М. Безпека життєдіяльності: Навч. посіб. — Львів., 2005. – 301 с.

19. Желібо Є. П. Заверуха Н.М., Зацарний В.В. Безпека життєдіяльності. Навчальний посібник. / Є. Желібо Є.П., Н.М. Заверуха П., В.В. Зацарний. – К.; Каравела, 2004. -328 с.