Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____*Комп'ютерних наук*_____
(повна назва)
Кафедра_____*Системотехніки*_____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____другий (магістерський)_____

Дослідження та аналіз можливостей верифікації документів за допомогою блокчейн технологій
(тема)

Виконав:
студент _2__ курсу, групи _СПР*м-22-1*
_____Серкін *К. О.*_____
(прізвище, ініціали)

Спеціальність *122 Комп'ютерні науки* __
(код і повна назва спеціальності)
Тип програми __освітньо-професійна_____
Освітня програма _«Системне *проектування»* _____
(повна назва освітньої програми)

Керівник _____*проф. Іванов В. Г.*_____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ __*Гребеннік І.В.*__
(підпис) (прізвище, ініціали)

2024 р.

Кваліфікаційна робота оформлена у відповідності до вимог діючих стандартів та методичних вказівок.

Матеріали кваліфікаційної роботи не містять відомостей, що заборонені для опублікування у відкритих виданнях.

Попередній захист проведено 03 червня 2024 року.

Керівник кваліфікаційної роботи *В.Г. Іванов*

Харківський національний університет радіоелектроніки

Факультет *Комп'ютерних наук*

Кафедра *Системотехніки*

Рівень вищої освіти другий (магістерський)

Спеціальність *122 Комп'ютерні науки*
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма *«Системне проектування»*
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«___» _____ 20___ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові *Серкіну Кирилу Олексійовичу*
(прізвище, ім'я, по батькові)

1. Тема роботи: *Дослідження та аналіз можливостей верифікації документів за допомогою блокчейн технологій*
затверджена наказом по університету від *01.04.* 2024 р. № 259СТ
2. Термін подання студентом роботи до екзаменаційної комісії: 17.06.2024 р
3.Вихідні дані до роботи: *Дослідити можливості блокчейн технологій так наявних систем. Порівняти вразливості централізованих та децентралізованих способів зберігання даних. Виявити механізми захисту даних в блокчейн мережах. Проаналізувати можливості верифікації документів за допомогою блокчейн технологій. Проаналізувати вже наявні системи верифікації документів. Розробити концепт додатку для верифікації документів за допомогою блокчейн систем.*
4. Перелік питань, що потрібно опрацювати в роботі: *Вступ 1 Аналіз предметної області 1.1 Опис предметної області 1.2 Дослідження та аналіз блокчейн технологій 2 Постановка задачі 2.1 Проаналізувати вразливості централізованих систем зберігання даних 2.2 Проаналізувати вразливості децентралізованих систем зберігання даних 2.3 Порівняння недоліків та вразливостей централізованих та децентралізованих рішень 2.3.1 Порівнянти схожі вразливості систем 2.3.2 Проаналізувати небезпечні вразливості централізованих системи 2.3.3 Проаналізувати небезпечні вразливості децентралізованих систем 2.3.4 Бачення оптимального рішення для верифікації документів 2.4 Проаналізувати та порівняти наявні системи верифікації документів 2.4.1 Зробити детальний аналіз та розгляд обраних платформ 2.4.2 Визначити критерії для порівняння вже існуючих рішень з потенційним рішенням на основі блокчейн технологій 2.4.3 Проаналізувати отримані дані 2.5 Архітектура додатку 2.6 Вимоги до зберігання даних в блокчейн мережі 2.7 Вимоги до користувацького інтерфейсу 3 Визначення алгоритмів розроблюваної системи 3.1 Опис алгоритмів головних бізнес-процесів розроблюваного інструменту верифікації на базі блокчейну 4 Програмна реалізація 4.1 Архітектура застосунку 4.2 Імплементація безпеки застосунку 4.3 Реалізація основних механізмів підписання та верифікації документів Висновки*
5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій: _____

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
	Проф. Іванов Валерій Генадійович		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	*Отримання завдання на виконання роботи*	*11.01.2024*	Виконано
2	*Дослідження предметної області та наявних блокчейн технологій*	*02.02.2024-27.03.2024*	Виконано
3	*Пошук , дослідження та аналіз наявних сервісів*	*28.04.2024 – 22.04.2024*	Виконано
4	*Порівняння аспектів знайдених платформ*	*23.04.2024 – 1.05.2024*	Виконано
5	*Формування бізнес-вимог та бізнес ризиків*	*2.05.2024 – 7.05.2024*	Виконано
6	*Вибір технологій та інтеграцій*	*08.05.2024 – 13.05.2024*	Виконано
7	*Розробка альтернативного інструменту на базі блокчейн*	*14.05.24 – 17.05.24*	Виконано
8	*Написання кваліфікаційної роботи*	*18.05.2024 - 09.06.2024*	Виконано
9	*Представлення кваліфікаційної роботи в ДЕК*	*17.06.2024*	Виконано

Дата видачі завдання 01.04.2024 р.

Студент _____ Серкін К. О.
(підпис)

Керівник роботи _____ проф Іванов В. Г.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Атестаційна робота: 71 с., 23 рис., 23 джерел інформації, 1 таблиця.

БЛОКЧЕЙН, ВЕРИФІКАЦІЯ, СМАРТ-КОНТРАКТИ, EVM, ETHEREUM, WEB3, JS.

Об’єкт дослідження – блокчейн технології та блокчейн мережі на базі ядра Ethereum.

Предметом дослідження є можливість верифікації документів за допомогою використання блокчейн технологій та блокчейн мереж.

Мета роботи – дослідити та проаналізувати можливості блокчейн технологій в сфері верифікації документів. Порівняти захищеність централізованих та децентралізованих систем. Провести аналіз безпечності вже розроблених рішень, які надають можливість верифікації документів централізовано. Розробити концепт веб-додатку, який дозволити інтегруватись з блокчейн мережами й верифікувати документі за допомогою власного крипто-гаманця.

До методів дослідження належать вивчення та аналіз теоретичних відомостей й технічної документації. Окрім того, була використана методика аналогій, котра включає аналіз існуючих систем верифікації документів для визначення та порівняння сильних та слабких сторін вже розроблених рішень. Це допомогло виявити потенційні вразливості та можливості покращення нових систем на базі блокчейн технологій.

В результаті дослідження та розробки отримано концепт системи який має змогу верифікувати документи та підтверджувати, те що вони не були змінені з моменту їх першого завантаження в блокчейн мережу.

ABSTRACT

Explanatory note to the certification work of the master's degree contains: 71 pages, 23 pictures, 23 sources of information and 1 table.

BLOCKCHAIN, VERIFICATION, SMART CONTRACTS, EVM, ETHEREUM, WEB3, JS.

The object of research is blockchain technologies and blockchain networks based on the Ethereum core.

The subject of the study is the possibility of verifying documents using blockchain technologies and blockchain networks.

The purpose of the study is to investigate and analyze the capabilities of blockchain technologies in the field of document verification. To compare the security of centralized and decentralized systems. To analyze the security of already developed solutions that provide the ability to verify documents centrally. To develop a concept of a web application that will allow integration with blockchain networks and verification of documents using its own crypto wallet.

Research methods include studying and analyzing theoretical information and technical documentation. In addition, the analogies methodology was used, which includes the analysis of existing document verification systems to identify and compare the strengths and weaknesses of already developed solutions. This helped to identify potential vulnerabilities and opportunities to improve new systems based on blockchain technologies.

As a result of the research and development, a concept of a system was obtained that can verify documents and confirm that they have not been changed since they were first uploaded to the blockchain network.

Зміст

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

Ethereum – Назва блокчейн мережі та платформа для створення децентралізованих онлайн-сервісів на базі блокчейна.

Solidity — мова програмування яка використовується для написання смарт-контрактів.

EVM (Ethereum Virtual Machine) – Віртуальна машина ефіріуму, яка може виконувати заздалегідь скомпільований код.

Смарт-контракт — Вид програмного файлу який розгортається всередині блокчейн мережі.

Web3 – Концепція світової мереж заснованої на блокчейн технологіях

БД — База даних

ВСТУП

У сучасному світі цифрових технологій електронні документи стали невід'ємною частиною нашого життя. При таких умовах проблема забезпечення їхньої цілісності та достовірності стає все більш значущою. Традиційні методи зберігання документів, такі як паперові копії або централізовані бази даних, можуть не забезпечувати високого рівня захисту від підробки та викрадення. Електронні документи мають юридичну силу, та необхідність забезпечення єдиного достовірного джерела підтвердження оригінальності стає все більш очевидною з кожним днем.

Забезпечення єдиної та незмінної істини є дуже складним завданням. У таких випадках звичною практикою є зберігання даних в централізованих базах даних. Цей метод майже не змінювався роками, в той час як випадки підробки та викрадення даних з централізованих систем стає все більше.

Вирішення сучасних проблем зазвичай потребує сучасного підходу. У сфері зберігання достовірності електронних документів, централізовані системи можна замінити децентралізованими. Ключовим аспектом децентралізованого способу зберігання даних є їх незмінність та неможливість підробки даних після моменту їх запису у блокчейн мережу. Це становить ключову відмінність у методах збереження та захисту інформації. Блокчейн технології в такому випадку можуть слугувати єдиним джерелом достовірності, й одночасно бути платформою на якій можна розгортати великі системи обліку та верифікації документів.

Таким чином, це дослідження має на меті ретельно проаналізувати сучасні підходи до верифікації документів. Визначити основні вразливості різних методів та практик зберігання інформації. Знайти методи верифікації документів та вивчити можливості їхнього застосування у контексті блокчейн технологій. Використовуючи методи проектування та дослідження систем, планується розробити концепцію проекту, яка дозволить ефективно використовувати блокчейн для верифікації документів. Це дослідження спрямоване на розробку концепції проекту, який може відкрити нові можливості у сфері забезпечення достовірності та цілісності даних.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис предметної області

Проблемою винесеною на дослідження в даній атестаційній роботі – є безпека зберігання, забезпечення достовірності та незмінності документів у централізованих системах. Це дуже важливий аспект, особливо у рамках сучасного цифрового світу, де обмін електронних документів став невід'ємною частиною таких сфер діяльності як:

- правові відносини;
- бізнес та фінанси;
- сфера надання електронних послуг;
- освіта та наука;
- медицина;
- соціальна сфера.

Якщо поглянути на кожне твердження що приведено вище, можна без перебільшення сказати, що процес обміну електронних документів охоплює майже кожну сферу діяльності людини. Зважаючи на це гострота проблеми забезпечення цілісності та достовірності даних набуває особливої актуальності. На даний момент з традиційних методів зберігання та передачі інформації ми маємо:

- електронна пошта;
- файли PDF;
- централізовані бази даних;
- паперові архіви.

При більш детальному розгляді кожного з варіантів що приведено віще, справедливим твердженням буде що кожен з цих варіантів має ризики та у деяких випадках, недоліки які з часом ускладнюють, а іноді навіть унеможливлюють процес безпечного зберігання даних. Кожин з цих методів був детально розглянутий нижче.

Електронна пошта може беззаперечно бути зручним засобом комунікації та обміну інформації, але вона не може бути ефективним та надійним інструментом для збереження даних.

Файли PDF це найрозповсюдженіший формат у якому зберігаються документи для їх подальшого обміну, але навіть такий файл не захищено від примусового редагування чи повної підміни файлу включаючи метадані.

Продвинутим методом зберігання даних є централізовані бази даних. Типів централізованих баз даних дуже багато, але навіть якщо просто поверхово проаналізувати даний метод то можна зрозуміти що, вразливість лежить на поверхні терміну “централізована”. Це означає що усі дані зберігаються в одному місці й доступ до них мають, адміністратори, розробники чи будь-які співробітники компанії яким надані особливі права доступу.

Найстарішим з наявних методів є паперові архіви, цей тип зберігання даних не дає гнучкості, достовірності й схильний до впливу великого ряду чинників які можуть призвести до підробки та втрати інформації.

Зважаючи на проблеми які були описані вище виникає потреба в надійних та ефективних механізмах верифікації електроних документів. В рамках дослідницької діяльності одним з потенційних рішень є технологія блокчейн.

З перспективи досліджень та аналізу технологій такого формату дуже велика увага приділяється методам взаємодії, зберігання та гарантування безпеки запису та отримання даних. Задля цього треба поглиблено проаналізувати всі наявні ризики використання децентралізованих систем.

Важливою приміткою є те, що жодна система не може гарантувати стовідсоткової безпеки верифікації даних. Тому детальне дослідження наявних рішень у сфері децентралізованого зберігання інформації є ключовим аспектом цієї дослідницької роботи.

Головними критеріями вибору та порівняння вже наявних систем верифікації документів є безпека та незмінність даних які будуть зберігатись в рамках платформи чи мережі.

Науково-технічна задача даної атестаційної роботи полягає у детальному аналізі вже існуючих платформ для верифікації документів, порівнянні методів зберігання інформації, й найбільша увага буде приділятися вразливості систем до стороннього впливу. Як внесок в здобуток науково-технічної спільноти кваліфікаційна робота передбачає створення концепту веб-додатку який буде представляти новий підхід до зберігання та верифікації інформації

1.2 Дослідження та аналіз блокчейн технології

В процесі розглядання проблематики використання звичайних та розповсюджених методів зберігання й верифікації інформації було прийнято рішення використовувати технологію блокчейн. З огляду на це треба детальніше дослідити та проаналізувати основні принципи роботи подібних систем.

Що таке блокчейн? Технологія блокчейн - це вдосконалений механізм баз даних, який дозволяє прозоро обмінюватися інформацією в бізнес-мережі. База даних блокчейну зберігає дані у вигляді блоків, які пов'язані між собою в ланцюжок [1]. Дані є хронологічно послідовними, оскільки ми не можемо видалити або змінити ланцюжок без консенсусу в мережі, про це детальніше буде розтлумачено нижче. Як результат, ми можемо використовувати технологію блокчейн для створення незмінного реєстру або для відстеження замовлень, платежів, рахунків та інших транзакцій. Це безпосередньо і є основною перевагою перед централізованими базами даних. Система має вбудовані механізми, які запобігають несанкціонованому внесенню транзакцій і створюють узгодженість у спільному баченні цих транзакцій. Механізм консенсусу буде розглянуто більше детально в рамках дослидницької діяльності.

Оскільки блокчейн це велика розподілена база даних дуже важливо знати як зберігається інформація в рамках блокчейн мережі. Основним принципом зберігання даних у блокчей мережі є - Distributed ledger technology - технологія розподіленого реєстру. Технологія розподілених реєстрів (DLT) - це цифрова

система обліку операцій з інформацією або активами, в якій транзакції та їхні деталі реєструються в декількох місцях одночасно. На відміну від традиційних баз даних, розподілені реєстри не мають центрального сховища даних або функцій адміністрування [2].

У розподіленому реєстрі кожен вузол обробляє і перевіряє кожен елемент, створюючи таким чином запис про кожен елемент і формуючи консенсус щодо його достовірності. В рамках блокчейн мереж вузлом зберігання даних є віддалені серверні машини, кожна з яких зберігає одну й ту саму копію всієї мережі, тобто всіх даних які коли небудь потрапили у мережу. Схему зберігання даних при використанні технології розподіленого реєстру можна подивитись на рисунку 1.1.

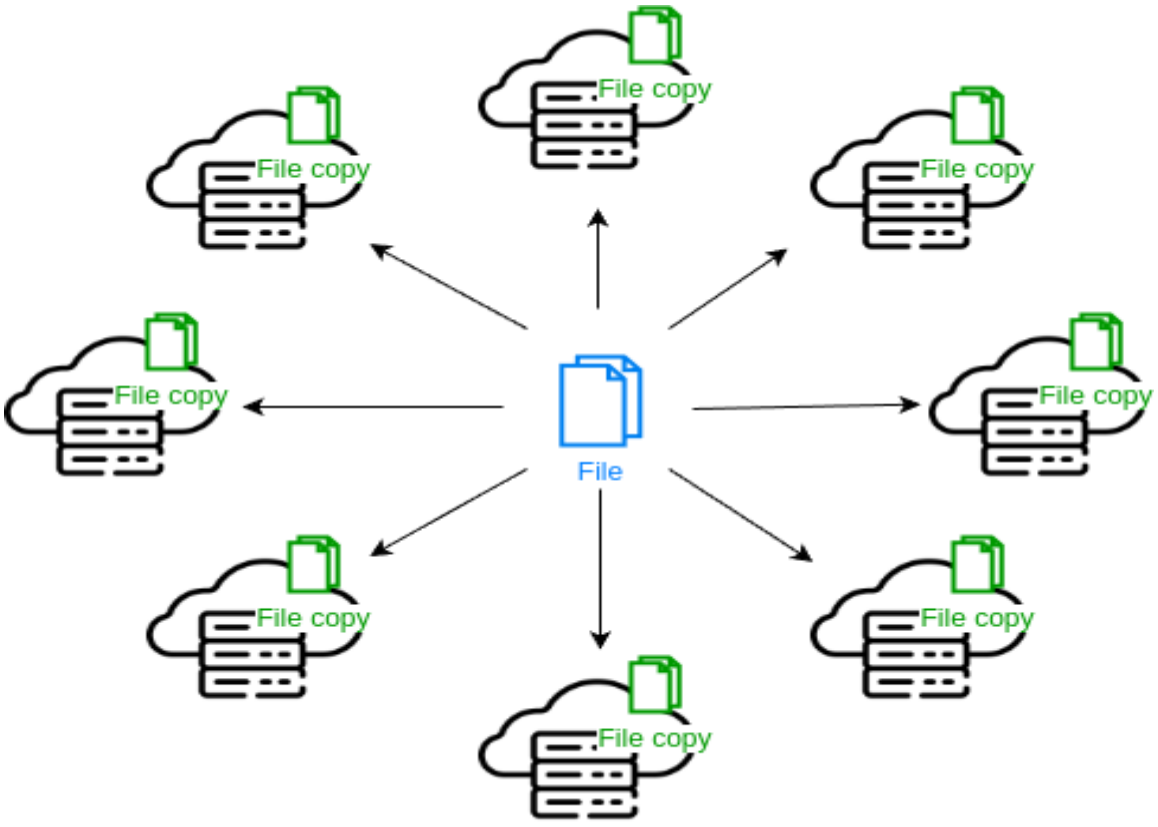

Рисунок 1.1 - Діаграма зберігання даних в мережі блокчейн.

Такий спосіб зберігання даних дає нам змогу у випадку виходу із ладу однієї чи декількох серверних машин, де зберігалась інформація, мати доступ до потрібних нам даних. Тобто в мережі блокчейн є десятки, а інколи і сотні екземплярів однієї й тієї самої інформації.

Децентралізоване розташування даних дає змогу не тільки не втрачати доступ до інформації, але й також перевіряти її незмінність. Перевірка незмінності, а отже достовірності інформації яка колись була записана, відбувається за допомогою основного механізму консенсусу блокчейна.

Під консенсусом ми розуміємо, що досягнуто загальної згоди. Згоди з приводу стану нової записаної інформації, а також інформації що колись була вже записана в мережу. Уявіть собі групу людей, які збираються в кінотеатр. Якщо немає розбіжностей щодо запропонованого вибору фільму, то консенсус досягнутий. Якщо ж розбіжності є, група повинна мати засоби, щоб вирішити, який фільм подивитися. В крайньому випадку група врешті-решт розділиться.

Що стосується блокчейну, то цей процес формалізований, і досягнення консенсусу означає, що принаймні 66% вузлів мережі погоджуються з глобальним станом мережі.

Саме завдяки консенсусу на всіх вузлах мережі завжди підтримується одна й та сама інформація. Якщо один або декілька вузлів мережі будуть представляти недостовірні дані, такий вузол буде оштрафовано. Оскільки кожен вузол для своєї участі в процесі формування блоку повинен представити фіксовану кількість токенів мережі в якій він працює, то штраф буде списано з рахунку такого вузла.

Кожна нова взаємодія з мережею називається “транзакцією”, усі такі транзакції виконуються в рамках поточного блоку який генерує мережа. Кожна транзакція після виконання отримує свій унікальний хеш. Кожен раз, коли додається новий набір транзакцій, він називається "блок" [3]

Хеш є унікальним ідентифікатором який ніколи не повторюється. Після того як блок в рамках якого була виконана взаємодія закривається, з усіх хешів транзакцій збирається так зване “дерево Меркла”. Дерево Меркла є фундаментальною частиною технології блокчейн. Це математична структура даних, що складається з хешів різних блоків даних і слугує підсумком усіх транзакцій у блоці. Вона також дозволяє ефективно і безпечно перевіряти вміст великого масиву даних [4]. Такі великі блокчейн мережі як Bitcoin та Ethereum використовують цю математичну структуру у багатьох системних процесах

обробки та запису даних. Дерева Меркла створюються шляхом багаторазового обчислення хешування пар вузлів, поки не залишиться тільки один хеш. Цей хеш називається коренем Меркла, або кореневим хешем. Дерева Меркла будуються за принципом "знизу вгору". Приклад генерації “кореня Меркла” за допомогою “дерева Меркла” приведено нижче на рисунку 1.2.

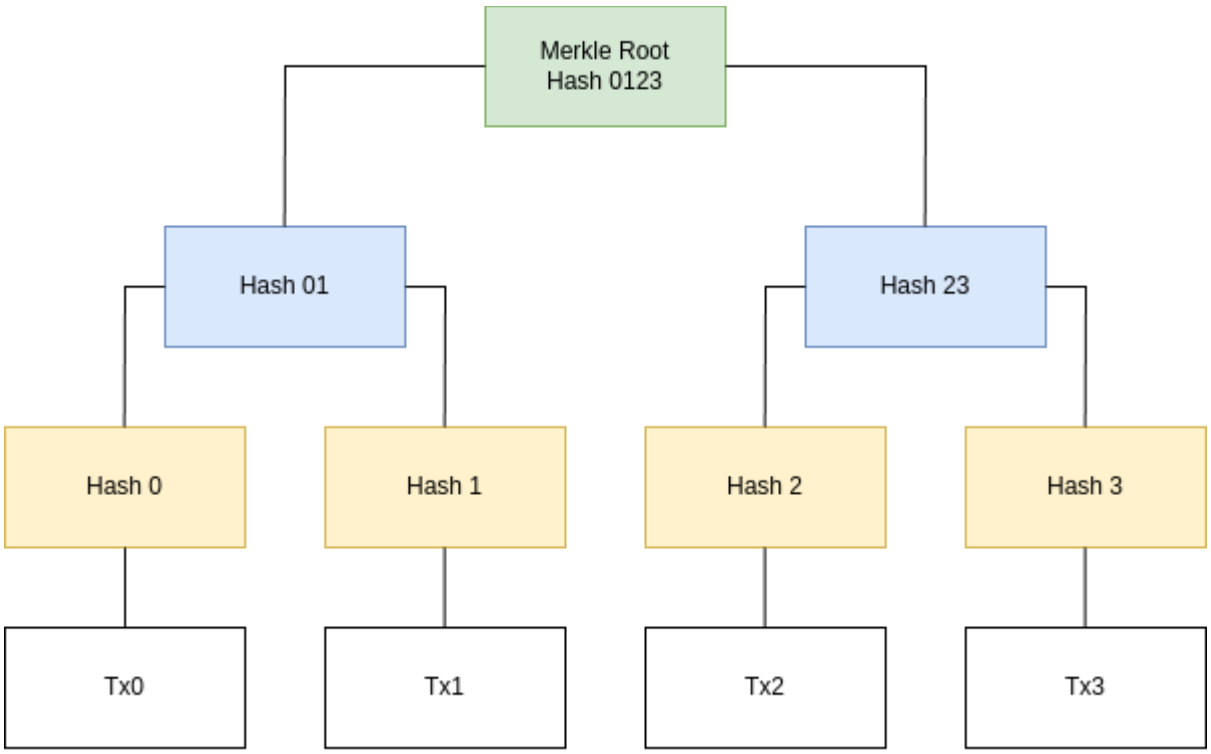

Рисунок 1.2 - Принцип генерації корінного хеша дерева меркла.

Результат побудови такої структури даних буде використовуватись в якості складової яка забезпечує незмінність порядку та результатів взаємодій що були зроблені користувачами в блокчейн мережі. Корінь дерева Меркла записується у новий блок у якості заголовку.

Кожен блок містить унікальний код, який називається "хеш", що дозволяє відрізнити його від будь-якого іншого блоку, а також "хеш" попереднього блоку в ланцюжку, що пов'язує їх між собою. Це створює ланцюжок блоків, або "блокчейн", який неможливо змінити або підробити, оскільки будь-яка зміна в блоці також змінить його хеш, а отже, він більше не буде відповідати хешу попереднього блоку. Саме це робить технологію блокчейн безпечною і захищеною від підробки.

Загалом структура блоків у різних мережах відрізняється. Оскільки зараз найпопулярнішим рішенням є мережі на базі блокчейн мережі Ethereum, треба детальніше розглянути структуру таких блоків.

Кожен блок в Ethereum складається з 2 основних частин:

– Заголовок;

– Тіло блоку.

Заголовок блоку Ethereum містить кілька полів, які надають інформацію про блок, майнер і поточний стан мережі в тому числі. Нижче приведено усі наявні значення які зберігаються в заголовку блоку.

Перше це - Parent block hash. Parent block hash, також відомий як "хеш попереднього блоку", є посиланням на хеш попереднього блоку в блокчейні. Він міститься в заголовку кожного блоку в блокчейні Ethereum і використовується для з'єднання блоків в ланцюжок. Це створює очевидний і прозорий спосіб перевірки цілісності всього ланцюжка блоків.

Далі в нас є так званий Uncle Hash - це посилання на хеш блоку, який не включений в основний блокчейн, але все ще вважається дійсним. В Ethereum, коли майнер знаходить новий блок, інші майнери також можуть працювати над пошуком нового блоку в той же час. Якщо майнери знаходять новий блок одночасно, той, чий блок буде додано до основного ланцюжка блоків першим, називається "головним блоком", а інший - "блоком-дядьком".

Дуже важливим є State Root. State Root - це посилання на корінь дерева стану в блокчейні Ethereum. Тріада станів - це структура даних, яка зберігає поточний стан мережі Ethereum, включаючи баланс всіх облікових записів, зберігання всіх контрактів і кількість всіх облікових записів. Дерево станів є модифікованою версією дерева Меркла, структури даних, яка дозволяє ефективно перевіряти вміст дерева.

Після State Root йде що один корінь під назвою Receipt root - це посилання на корінь дерева квитанцій в блокчейні Ethereum. Дерево квитанцій - це структура даних, яка зберігає квитанції про транзакції, включені в блок. Квитанція про транзакцію містить інформацію про результат транзакції,

наприклад, чи була вона успішною, кількість використаного газу та адресу контракту, якщо транзакція створила новий контракт.

Важливим фільтром є Logs bloom який включається в заголовок кожного блоку в блокчейні Ethereum. Він використовується для ефективної перевірки того, чи включена в блок подія логу від виконання контракту. Лог-подія - це запис події, яка відбулася під час виконання смарт-контракту, наприклад, переказ коштів або зміна стану контракту.

Рівень складності або Difficulty в заголовку блоку Ethereum вказує на рівень складності алгоритму підтвердження роботи, який використовується для перевірки нових блоків в блокчейні Ethereum. Рівень складності - це показник того, наскільки важко знайти дійсний блок, і він динамічно коригується на основі поточного стану мережі.

Number, також відома як номер блоку, міститься в заголовку кожного блоку в блокчейні Ethereum. Це скалярне значення, яке представляє позицію блоку в блокчейні. Перший блок в блокчейні Ethereum, також відомий як блок генезису, має номер блоку 0.

Gas limit в заголовку блоку Ethereum - це скалярне значення, яке представляє максимальну кількість газу, що може бути використана транзакціями в блоці. Газ - це внутрішній механізм ціноутворення, який використовується в Ethereum для оплати обчислень смарт-контрактів і транзакцій в мережі Ethereum.

Як і в багатьох системах зберігання даних ми маємо timestamp в заголовку блоку Ethereum - це скалярне значення, яке представляє час, коли блок було видобуто. Це часова мітка Unix, яка являє собою кількість секунд, що минули з 1 січня 1970 року, 00:00:00 UTC;

Останнім показником заголовку блоку є Extra data. Це поле додаткових даних в заголовку блоку Ethereum, також відоме як "додаткові дані" або "додаткове поле", - це 32-байтне поле, яке можна використовувати для включення додаткових даних в заголовок блоку. Поле додаткових даних не використовується протоколом Ethereum для будь-яких конкретних цілей і призначене для використання майнерами або іншими користувачами мережі.

Воно може використовуватися для включення повідомлення, підпису або інших даних, які можуть бути корисними для майнера або інших користувачів мережі. Основними полями в тілі блоку Ethereum є:

Список транзакцій, включених в блок. Кожна транзакція містить таку інформацію, як адреса відправника, адреса одержувача, кількість ефіру для переказу і кількість газу для споживання;

Пращури, тобто список застарілих блоків, які були включені в блок. Ці застарілі блоки включаються в блок як винагорода майнерам, які їх видобули, навіть якщо вони не були включені в основний блокчейн;

Корінь транзакцій - це поле, що містить корінь Меркла списку транзакцій у блоці. Корінь Меркла - це в даному випадку хеш всіх транзакцій в блоці, і він використовується для того, щоб довести, що певна транзакція включена в блок без необхідності включати всі транзакції в заголовок блоку;

Ще один значущій корінь це - корінь пращура. Представляє собою поле, що містить корінь Меркла зі списку пращурів у блоці. Корінь Меркла - це в даному випадку хеш всіх пращурів у блоці;

Ліміт газу, таке поняття також використовується в заголовку блоку. Це поле, яке містить максимальну кількість газу, що може бути використана транзакціями в блоці. Ліміт газу встановлюється майнером, який видобуває блок, і використовується для того, щоб запобігти перевантаженню мережі великою кількістю транзакцій;

Використаний газ, величина напряму визначаюча скільки коштували транзакції. Поле, що містить кількість газу, яка була фактично використана транзакціями в блоці;

Винагорода за блок, тобто винагорода за обчисленя. Поле, яке містить винагороду, що надається майнеру або стейкеру, який видобув блок. Ця винагорода є комбінацією винагороди за блок, винагороди пращурів і винагороди за транзакцію;

Стандартне поле "value" в транзакції в тілі блоку Ethereum використовується для вказівки кількості Ефіру, яка передається від відправника до одержувача в рамках транзакції. Це важливе поле, оскільки воно представляє

вартість, що передається, і використовується для обчислення загальної вартості, переданої в блоці;

Для визначення змін стерту є поле даних в тілі блоку містить всі транзакції, які були включені в блок майнером. Ці транзакції можуть включати різні типи транзакцій, такі як створення контракту, виконання контракту і передача токенів;

Поле "to" в транзакції в тілі блоку Ethereum використовується для вказівки адреси одержувача, на яку передається Ефір. Це важливе поле, оскільки воно представляє адресу одержувача і використовується для підрахунку загальної вартості, переданої на певну адресу в блоці.

Скорочена схема блоку зображена на рисунку 1.3.

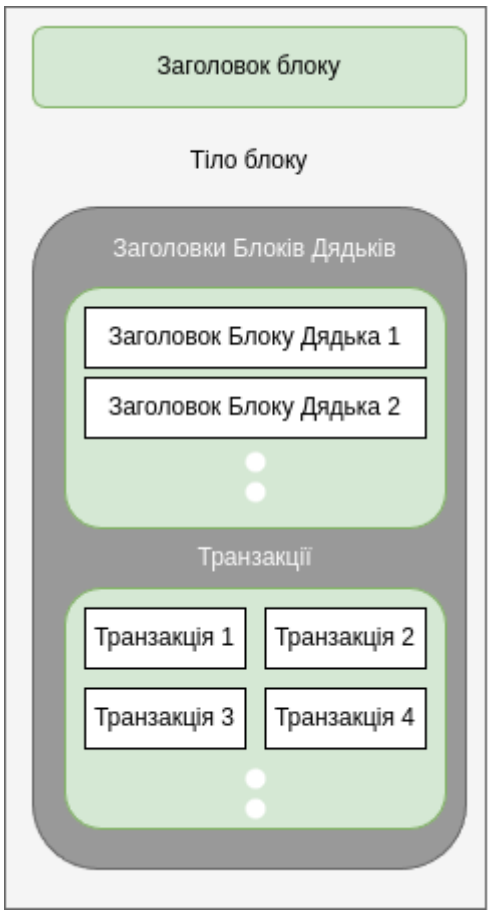

Рисунок 1.3 - Скорочена схема блоку

Блокчейн це не просто місце де зберігається інформація. В самих перших блокчейн системах все було саме так, але завдяки виникненню такої мережі як Ethereum блокчейн став ще й середою обробки даних та виконання невеликих программ. Всередині блокчейн мережі Ethereum розгорнуто віртуальну машину яка скорочено називається EVM, або Ethereum Virtual Machine. Ця віртуальна машина дозволяє виконувати код деяких програм які завантажуються безпосередньо в мережу блокчейн. Фізична реалізація EVM не може бути описана так само, як хмара, але вона існує як єдине ціле, яке підтримується тисячами підключених комп'ютерів, на яких працює клієнт Ethereum.

Сам протокол Ethereum існує виключно для того, щоб підтримувати безперервну, безперебійну і незмінну роботу цієї спеціальної машини стану. Це середовище, в якому живуть всі облікові записи Ethereum і смарт-контракти. У кожному блоці ланцюжка Ethereum має один і тільки один "канонічний" стан, і саме EVM визначає правила обчислення нового дійсного стану від блоку до блоку. У цьому сенсі Ethereum блокчейн це не просто розподілена база даних, це велика розподілена база станів окремих елементів системи.

Такими елементами системи є смарт-контракти. Смарт-контракт - це децентралізована комп'ютерна програма, що працює в мережі блокчейн, яка автоматично і детерміновано виконує угоди на основі заздалегідь визначених умов [5]. Смарт-контракти розміщуються і виконуються в мережі блокчейн. Кожен смарт-контракт складається з коду, що визначає заздалегідь визначені умови, виконання яких призводить до певних результатів. Працюючи на децентралізованому блокчейні замість централізованого сервера, смарт-контракти дозволяють кільком сторонам досягти спільного результату в точний, своєчасний і захищений від втручання спосіб.

Смарт-контракти є потужною інфраструктурою для автоматизації, оскільки вони не контролюються центральним адміністратором і не є вразливими до окремих точок атаки зловмисників. Застосовуючись у багатосторонніх цифрових угодах, додатки зі смарт-контрактами можуть зменшити ризик контрагента, підвищити ефективність, знизити витрати та

забезпечити новий рівень прозорості та надійності процесів. У випадку з мережею Ethereum та подібними мережами, смарт-контракти можуть бути написані на особливій мові програмування яка називається Solidity. Solidity - це об'єктно-орієнтована мова високого рівня для реалізації смарт-контрактів [6].

Solidity має статичну типізацію, підтримує успадкування, бібліотеки та складні користувацькі типи.

За допомогою Solidity ви можете створювати контракти для таких цілей, як голосування, краудфандинг, аукціони, гаманці з кількома підписами та зокрема зберігати інформацію яка може залишатися незмінною після запису.

Використання смарт-контрактів дає розробникам та компаніям можливості реалізації своїх платформ різного рівня, безпосередньо всередині блокчейн мережі. Й кожна зміна стану смарт-контракту буде записана раз і назавжди, що дає змогу уникнути несанкціонованого впливу будь-яких третіх осіб. Саме тому смарт-контракти це база для побудови децентралізованих рішень з максимальною прозорістю та захистом.

Завдяки детальному аналізу технології яка буде використана задля забезпечення безпеки інформації та верифікації документів ми отримуємо критерії за якими можна в подальшому порівнювати вже існуючі системи та методі верифікації даних.

2 ПОСТАНОВКА ЗАДАЧІ

Об’єктом дослідження є безпека та верифікації зберігання даних в централізованій і децентралізованій середі. Мають бути досліджені, проаналізовані та порівняні вразливості систем з використанням кожного з заявлених підходів. Знайти та провести аналіз наявних рішень верифікації документів для подальшої розробки концепту системи верифікації даних на базі технології блокчейн, який має бути реалізовано у вигляді веб-додатку. До основних можливостей, що мають бути реалізовані при розробці належать:

- завантаження документів для верифікації;
- верифікація того що документ або набір документів не було змінено;
- підписання документів за допомогою блокчейн мережі;

2.1 Проаналізувати вразливості централізованих систем зберігання даних

Оскільки централізовані системи зберігання даних зараз використовуються в більшості випадків, за весь час їх існування було виявлено дуже багато вразливостей, серед них відділенні основні:

- Централізований однопунктовий контроль;
- Загрози кібербезпеки;
- Одинична точка відмови;
- Складність масштабування;
- Проблеми приватності;
- Людський фактор адміністрування.

Далі ми детально розглянемо кожен з пунктів які були приведені вище.

Централізований однопунктовий контроль має велике значення у контексті верифікації документів та безпеки інформації.Перший пункт, що стосується "однопунктового контролю", має велике значення у контексті верифікації документів та безпеки інформації.

У централізованих системах існує одна центральна точка, через яку проходять всі транзакції та доступ до даних. Це може бути центральний сервер, база даних або система керування. Оскільки всі дані і всі процеси обробляються цією єдиною точкою, вона стає суттєвою для забезпечення правильності та безпеки інформації. Однак ця централізована точка контролю може стати потенційною слабким місцем системи з точки зору безпеки. Якщо центральний сервер або база даних піддається атакам, злому або внутрішньому порушенню безпеки, це може призвести до серйозних наслідків, таких як втрата або модифікація даних, несанкціонований доступ до конфіденційної інформації, або переривання роботи системи.

У контексті верифікації документів це також означає, що існує ризик, що централізована точка може бути скомпрометована, що призведе до недостовірності чи підробки документів. Така ситуація може підірвати довіру до системи верифікації та надійність збереження інформації.

Отже, для забезпечення високого рівня безпеки та надійності важливо розглядати альтернативні підходи, такі як децентралізовані системи, які розподіляють контроль інформації між різними вузлами мережі, зменшуючи тим самим ризики однопунктового контролю.

Загрози кібербезпеки мають дуже багато аспектів й видів, основними є: Кібербезпека централізованих систем стикається з різноманітними загрозами, які можуть призвести до порушення конфіденційності, цілісності та доступності даних. Найбільш поширені загрози поширені перечислень нижче.

Кібератаки та хакерські вторгнення. Централізовані системи можуть стати об'єктом кібератак та хакерських вторгнень, під час яких зловмисники намагаються отримати несанкціонований доступ до системи з метою викрадення, зміни або видалення даних, а також знищення або блокування роботи системи.

Фішинг. Атаки фішингу можуть бути спрямовані на користувачів централізованих систем з метою викрадення особистої інформації, такої як паролі, логіни, номери кредитних карток тощо. Це може призвести до незаконного доступу до системи через облікові записи користувачів.

Малвара і віруси. Централізовані системи можуть стати жертвами розповсюдження шкідливого програмного забезпечення, такого як віруси, троянці, рансомваре та інші види малвару. Це може призвести до втрати даних, викрадення конфіденційної інформації або навіть зупинки роботи системи.

DDoS-атаки. Атаки з відмови в обслуговуванні (DDoS) можуть призвести до недоступності централізованих систем для користувачів шляхом переповнення серверів запитами, що призводить до перерв у роботі системи та неможливості доступу до даних.

Внутрішні загрози. Внутрішні загрози включають дії зловмисників, які мають доступ до системи, таких як співробітники або партнери компанії. Вони можуть використовувати свій доступ для крадіжки конфіденційної інформації, зламу системи або зміни даних.

Соціальна інженерія. Централізовані системи також можуть бути вразливі до атак соціальної інженерії, коли зловмисники використовують маніпулювання та обман, щоб отримати доступ до системи або конфіденційної інформації через користувачів системи.

Єдина точка відмови (SPOF) - це будь-який програмний, апаратний або інший дефект, який може вивести систему з ладу, коли трапиться щось катастрофічне. Щоб запобігти простою і досягти високої доступності та надійності, критично важливі системи не повинні мати SPOF. Чого не завжди можна уникнути, системи багатьох платформ, продовжують збільшуватись й розвиватись. В такі моменти дуже складно знайти цю єдину точку відмови.

Складності масштабування централізованих систем у контексті верифікації документів та безпеки інформації може виявитися проблематичним через обмеженість обчислювальних ресурсів та потенційний ризик "одиничної точки відмови". Централізовані сервери, на яких зазвичай зберігається інформація, можуть стати перевантаженими при збільшенні обсягу даних або кількості користувачів, що може призвести до зниження продуктивності та доступності системи. Крім того, ця обмеженість ресурсів може ускладнити виявлення та усунення проблем в системі, що може відбутися через

перевантаження серверів або недостатність обчислювальних ресурсів для виконання необхідних операцій.

Дуже важливим аспектом у масштабуванні централізованих систем є питання безпеки даних. Збільшення обсягу інформації може зробити систему більш вразливою перед потенційними кібератаками та порушеннями безпеки. Крім того, централізована архітектура може зробити всю систему уразливою перед атаками, оскільки компрометація одного сервера може відразу ж поширитися на всю систему.

Таким чином, масштабування централізованих систем в рамках дослідження верифікації документів та безпеки інформації може стати складною задачею через обмеженість обчислювальних ресурсів, ризик "одиничної точки відмови" та підвищену вразливість перед кіберзагрозами.

Проблеми приватності у таких системах, де всі дані зберігаються централізовано на серверах чи в обмеженому наборі серверів, існує значний ризик незаконного доступу до цих даних. Це може виникнути через несанкціонований доступ з боку зловмисників, атаки з використанням вразливостей системи або внутрішні проблеми безпеки, такі як зловживання працівниками.

Недостатня захищеність даних у централізованих системах може мати серйозні наслідки для приватності користувачів. Якщо зловмисники отримають доступ до централізованих баз даних, вони можуть отримати доступ до великої кількості конфіденційної інформації, включаючи особисті дані користувачів, банківські дані, інформацію про транзакції та інші чутливі дані.

Безпека централізованих систем також може бути піддана ризику через недоліки в програмному забезпеченні, недостатню захищеність мережі або вразливості в архітектурі системи. Порушення безпеки даних у централізованих системах може призвести до витоку конфіденційної інформації, шантажу, крадіжки ідентифікаторів або інших видів кіберзлочинності. Такі вразливості створюють серйозний ризик для приватності і безпеки користувачів і піднімають питання щодо надійності централізованих систем зберігання та обробки даних.

Проблема людського фактору в адмініструванні централізованих систем та баз даних виникає через можливі помилки, недбалість або зловживання з боку адміністраторів або інших уповноважених осіб, які мають доступ до системи. В такому випадку недостатня кваліфікація адміністраторів або їх недбалість можуть призвести до неправильної конфігурації системи або відсутності необхідних заходів безпеки. Наприклад, слабкі паролі, недостатня обробка прав доступу або неправильна настройка файрволів можуть створити вразливості, через які зловмисники можуть отримати доступ до системи. Адміністратори також можуть випадково видалити або пошкодити дані через недбалість або невірні дії. Наприклад, вони можуть випадково видалити важливі файли або встановити неправильне програмне забезпечення, що може спричинити втрату даних або недоступність системи. Адміністратори мають значний рівень доступу до системи, що може спричинити можливість зловживання цими привілеями. Вони можуть неправомірно переглядати, копіювати або змінювати конфіденційну інформацію, а також використовувати свій доступ для здійснення шкідливих дій. Деякі адміністратори можуть не приділяти достатньої уваги заходам безпеки або не виявляти серйозності до потенційних загроз безпеці. Це може призвести до ігнорування важливих попереджень безпеки, не вчасного оновлення програмного забезпечення або недостатньої реакції на інциденти безпеки. Ці аспекти підкреслюють важливість ретельного управління доступом, навчання персоналу з питань безпеки та встановлення механізмів контролю та моніторингу для запобігання небажаним діям адміністраторів та мінімізації ризиків від людських помилок.

2.2 Проаналізувати вразливості децентралізованих систем зберігання даних

В сучасному цифровому світі децентралізовані системи зберігання даних стають все більш поширеними, пропонуючи нові можливості для забезпечення безпеки та надійності інформації. Однак вони також відкривають деякі унікальні вразливості, які можуть стати об'єктом атак з боку зловмисників.

Аналіз цих вразливостей стає надзвичайно важливим для розуміння ризиків, пов'язаних з використанням децентралізованих систем. У цьому розділі ми ретельно проаналізуємо найбільш поширені вразливості, що можуть виникнути в децентралізованих системах зберігання даних, і розглянемо можливі наслідки цих вразливостей для безпеки та надійності інформації.

Серед найбільш розповсюджених вразливостей блокчейн технологій зазначають такі як:

- Вразливості Смарт-контрактів;
- Мережеві атаки;
- Атака 51 відсотку;
- Вразливості протоколів;
- Соціальні інженерні атаки;

Кожен з зазначених вище ризиків треба розглянути детально для більше поглибленого порівняння підходів до створення систем.

Проблеми безпеки смарт-контрактів в децентралізованих системах можуть мати серйозні наслідки і становити значний ризик для всієї мережі. Давайте розглянемо деякі з найбільш поширених проблем безпеки смарт-контрактів:

Уразливості програмування: Смарт-контракти, написані на мові програмування, можуть містити програмні помилки або баги, які стають джерелом потенційних вразливостей. Навіть невеликі помилки у коді можуть призвести до серйозних наслідків, включаючи втрату коштів чи неправомірний доступ до даних.

Атаки на переповнення стеку (Stack Overflow Attacks): Атаки на переповнення стеку можуть викликати аварійне завершення смарт-контракту або неправильне виконання операцій. Це може призвести до втрати активів або виконання небажаних операцій.

Використання токенів безпеки (Security Tokens) із недостатньою безпекою: У деяких випадках смарт-контракти можуть використовувати токени безпеки, які не мають належного рівня безпеки. Це може викликати ризики, такі як втрата токенів чи неправомірний доступ до ресурсів або інформації.

Використання функцій відкладеного виклику (Reentrancy Attacks) [7]: Атаки з використанням функцій відкладеного виклику можуть призвести до некоректного виконання смарт-контракту та втрати активів. Ця вразливість може бути використана для виклику функцій контракту до завершення попередньої функції, що призводить до неправильного розподілу ресурсів.

Фішинг атаки (Phishing Attacks): Хакери можуть створювати фішингові смарт-контракти, які намагаються підмінити офіційні контракти або видаватися за них, з метою обману користувачів та викрадення їхніх активів.

Мережеві атаки можуть стати серйозною загрозою для децентралізованих систем, особливо в контексті збереження даних та верифікації інформації.

Давайте розглянемо кілька типів мережевих атак і їх можливі наслідки в рамках вашої теми диплому:

DDoS атаки (Distributed Denial of Service): Ця атака полягає у спробі зробити ресурс недоступним для користувачів, перевантажуючи його мережеві ресурси. У децентралізованих системах, особливо якщо вони базуються на блокчейні, DDoS атаки можуть призвести до зниження продуктивності та завершення роботи вузлів, що може вплинути на доступність інформації та верифікацію документів.

Маніпулювання мережевим трафіком: Атаки на мережевий трафік можуть включати в себе вплив на передачу даних між вузлами системи. Наприклад, хакери можуть спробувати змінити або перехопити мережевий трафік, що призведе до некоректної верифікації документів або навіть до втрати даних.

Атаки на протоколи комунікації: У децентралізованих системах важливою ролью відіграють протоколи комунікації між вузлами. Атаки на ці протоколи можуть призвести до перехоплення, модифікації або блокування комунікаційних каналів, що порушить процеси збереження та верифікації даних.

Атаки на мережеву інфраструктуру: Хакери можуть спробувати атакувати саму мережеву інфраструктуру, включаючи сервери, маршрутизатори та

комутатори. Це може призвести до втрати доступу до системи, зниження продуктивності або навіть до втрати даних.

Усі ці атаки можуть стати серйозними загрозами для безпеки та надійності децентралізованих систем, тому важливо приділяти належну увагу заходам захисту та мережевій безпеці.

Атака 51% (або мажоритарна атака) - це потенційна загроза цілісності системи блокчейн, при якій одному зловмиснику або організації вдається контролювати більше половини загальної потужності хешування мережі, що потенційно може призвести до збою в роботі мережі.

Якщо один зловмисник або група зловмисників, що діють разом, контролюють понад 50% загальної потужності хешування в мережі блокчейн, вони зможуть обійти механізм консенсусу в мережі і вчинити зловмисні дії, такі як подвійне витрачання коштів.

Атака на 51% відбувається тоді, коли зловмисник має достатньо потужності, щоб навмисно змінити порядок транзакцій, запобігаючи підтвердженню деяких або всіх транзакцій. Це також називається відмовою в обслуговуванні транзакцій.

Вразливість протоколів є серйозною проблемою в контексті децентралізованих систем зберігання даних і верифікації інформації. Багато протоколів, які використовуються у децентралізованих системах, можуть мати недоліки в безпеці. Наприклад, протоколи передачі даних можуть бути підвернуті атакам перехоплення або модифікації даних, якщо вони не використовуються з ефективними методами шифрування та аутентифікації.

Криптографічні протоколи, які використовуються для захисту даних під час їх передачі та зберігання, також можуть бути вразливими. Недоліки в криптографічних алгоритмах або недоліки у реалізації можуть призвести до компрометації конфіденційності та цілісності даних.

Протоколи ідентифікації та автентифікації в децентралізованих системах можуть бути підвернуті атакам перехоплення і фальсифікації ідентифікаційних даних. Це може призвести до недозволеного доступу до системи або даних.

Атаки, спрямовані на протоколи комунікації між вузлами системи, можуть призвести до перехоплення, модифікації або блокування мережевого трафіку. Це може порушити процеси збереження та верифікації даних, а також призвести до втрати конфіденційності та цілісності інформації.

У випадку розгляду вразливості протоколів, потрібно розуміти що це здебільшого людський фактор, який притаманний усьому що створило людство.

Соціально-інженерні атаки є серйозною загрозою для безпеки інформації в децентралізованих системах зберігання даних. Ці атаки використовують маніпуляцію та обман, замість технічних засобів, для отримання неповного доступу до системи або конфіденційної інформації. Нижче наведені приклад подібних соціальних вразливостей.

Фішинг - це одна з найпоширеніших соціально-інженерних атак, яка полягає в тому, щоб змусити користувача розкрити свої конфіденційні дані, такі як паролі або особиста інформація. Це може бути зроблено шляхом надсилання підозрілих електронних листів або повідомлень, що містять посилання на підроблені веб-сайти, або шляхом видачі себе за легітимних осіб або організацій.

Соціальний інжиніринг - метод полягає в отриманні доступу до системи шляхом маніпуляції людьми, зазвичай через маніпуляцію довіри або використання дієвих соціальних інтеракцій. Наприклад, атакувальник може видавати себе за співробітника підтримки технічної служби та переконати користувача надати йому свій пароль або іншу конфіденційну інформацію.

Доступ до приміщення. Ця атака включає отримання фізичного доступу до комп'ютерних систем шляхом отримання доступу до приміщення, де вони знаходяться. Атакувальник може використовувати підроблені ідентифікаційні картки або методи соціального інжинірингу для незаконного проникнення до будівлі або приміщення, щоб отримати доступ до комп'ютерних систем.

2.3 Порівняння недоліків та вразливостей централізованих та децентралізованих рішень.

2.3.1 Порівняти схожі вразливості систем

Зважаючи на всі досліджені вразливості централізованих та децентралізованих систем, потрібно порівняти та визначити більш надійну систему для зберігання даних та їх подальшої верифікації. Спочатку треба розглянути однакові характери вразливості систем, та порівняти їх вплив на безпеку системи.

Спочатку треба звернути увагу на DDos-атаки. Атаки такого характеру вже відомі давно й у випадку таких атак на централізовані системи відомі чисельні випадки коли вся система вимикалась на тривалий термін. Все це тому що зазвичай у таких сервісів всього одна точка входу. В таких випадках децентралізований підхід має багато переваг. Точок входу у системі блокчейн дуже багато, а машин які обробляють інформацію дуже багато. Тож для того щоб така система вимкнулась під навантаженням, потрібно знати де знаходяться всі великі вузли мережі, й спрямовувати свої атаки на кожен з них.

Також треба розуміти що запити на зміну стану смарт-контракту, тобто стану блокчейн мережі, коштують реальних грошей, хоч і представлених у форматі токенів мережі. Чудовим захисним механізмом є те що чим більше навантажується мережа, тим дорожче кожна транзакція й кожний запит. Дому DDos-атаки верхніх рівнів не ефективні стосовно децентралізованих рішень. Захистити таку систему простіше.

Перейдемо до людського фактору та адміністрування систем. В даному випадку можна побачити значну різницю й значну перевагу на боці блокчейн технологій. Усі централізовані системи мають адміністраторів, співробітників з особливими привілеями, тому вони легко можуть стати жертвами фішингу, соціального інжинірингу або зіткнутися з вірусом. В той час як адміністрування блокчейн мережі проходить автоматично, тобто адміністратора немає, все вирішує механізм консенсусу.

Спосіб забезпечення цілісності та незмінності даних в системі яка використовує звичайний підхід з базою даних, простий. Є база даних, є її бекапи стан того чи іншого може бути змінений будь-якою людиною яка має доступ потрібного рівня. Вже існує багато доказів коли інформація в великих захищених базах даних, була викрадена, видалена або змінена. У випадку розподіленого зберігання інформації у рамках блокчейн системи, будь-яка зміна інформації записується в історію завдяки якій можна відстежити стадії зміни стану будь-якої змінної. Звісно тут треба зазначити що майже вся інформація в блокчейн є публічною. Але це не заважає нам зберігати зашифровану інформацію, чи просто корінь для підтвердження того, що файл у централізованій системі ніколи не був зміненим.

2.3.2 Проаналізувати небезпечні вразливості централізованих системи.

У випадку традиційних систем найбільш небезпечною вразливістю є централізований однопунктовий контроль, про який йдеться у пункті 1.3. В цьому випадку ми маємо систему, контроль якої належить людям. Саме людина є найбільш слабкою ланкою у цьому випадку. Завдяки цій вразливості користувач не може бути до кінця впевненим що його дані захищені та не були змінені. Є дуже багато прикладів коли саме цей фактор став фатальним для багатьох платформ та систем.

2.3.3 Проаналізувати небезпечні вразливості децентралізованих систем

Технологія блокчейн майже не вразлива до шкідливого впливу ззовні системи. Механізм консенсусу дає змогу запобігти будь-яким зовнішнім втручанням особи або групи осіб в процес зберігання даних. Але відомі випадки втручання в процес зберігання даних зсередини. Це так звана атака 51 відсотку про яку йдеться у пункті 1.4. Для того щоб провести таку атаку, група осіб повинна володіти більше ніж 50 відсотками мережі. Цей сценарій можливий, але потрібно розуміти блокчейн мережа постійно генерує блоки. Й

для змінення більше ніж декількох блоків потрібно багато часу та грошей. Блокчейн мережі такі як Ethereum дуже великі, вузлів у цій мережі занадто багато щоб зробити можливою атаку такого роду. Це непомірні по своїм об’ємам гроші та час який знадобиться для того щоб змінити інформацію всередині системи. В цей час виникає логічне питання, що робити іншим блокчейн мережам які не такі великі? Зараз дуже багато мереж базуються як новий рівень над основною мережею Ethereum, й усі блоки які записуються у невелику мережу стискаються й в рамках однієї транзакції записуються в мережу Ethereum. Це процес дозволяє використовувати величезний й стійкий механізм консенсусу не вразливої мережі в цілях забезпечення цілісності та незмінності інформації в мережі яка не має такої великої кількості вузлів.

2.3.4 Бачення оптимального рішення для верифікації документів.

Після ретельного аналізу будови блокчейн мереж та технології в цілому. Оцінивши усі вразливості обо типів систем. Стає очевидним що технологія децентралізованих систем зокрема блокчейн систем дає суттєву перевагу в сфері верифікації документів. Тож на основі проведеного дослідження було прийнято рішення, розробити концепт веб-додатку що зможе реалізувати процес верифікації документів за допомогою всіх переваг блокчейн технологій.

2.4 Проаналізувати та порівняти наявні системи верифікації документів

Для того щоб зрозуміти вимоги розробки додатку та його акутальність в межах сфери верифікації документів та даних необхідно дослідити та проаналізувати вже наявні системи верифікації документів.

Наявні системи що вже працють на території нашої країни повинні бути обрані за двома основними критеріями:

- Кількість користувачів що довіряють цій централізованій системі;
- Доступність для звичайного громадянина.

З відкритих джерел статистика була здобута інформація на основі якої було проведено аналіз да ключовими параметрами. За наявності відкритої статистики основними й актуальніми системами верифікації документів визнано:

- FREDO ДокМен;
- СОТА
- Вчасно.

Інформація про наявні інструменти була отримана з відкритих джерел та детально проаналізована. Для ознайомлення з зазначеними вище платформами було зроблено короткий опис кожного інструменту.

Ці платформи дозволяють вести електронний документообіг зберігаючи електронні документи для подальшої верифікації та підписання наявних документів. Завдяки наявності цих систем реалізується можливість створення документів які будуть доступні визначеному колу юридичних осіб або фізичних осіб підприємців. Юридичні особи або фізічні особи підприємці мають можливість укладати електронні версії договорів та завдяки особистим ключам, які є кваліфікованими електронними підписами або скорочено КЕП, затвердити свою згоду на підписання документу.

Після підписання документу усіма сторонами які заявлені у переліку осіб що мають затвердити документ, вище зазначені платформи надають можливість верифікувати належність документів та їх валідність для кола осіб осіб що були зазначені у документі який був підписаний усіми сторонами договіру. Таким чином кожна сторона договіру тощо, може бути впевнена у тому що договорі або контракт набуває юридичної сили.

За документообігом такого формату можна визначати діяльність компаній, юридичних осіб або фізичних осіб підприємців, та підтвердити легальність та силу підписаних документів.

В данному випадку за верифікацію документів та їх зберігання у недоторканному вигляді відповідає платформа що веде електронний документообіг. Механізми захисту такої платформи є цільковитим надбанням та сферою відповідальності розробників данної платформи. Також варто

пам’ятати що інформація яка зберігається завдяки даним сервісам лежить в базі даних яка є цільковитою властністю обраної платформи. Також не виключено що ресурси для зберігання інформації у захищеному вигляді належать ряду зовнішніх компаній які забезпечують виділині місця у хмарних сховищах.

2.4.1 Зробити детальний аналіз та розгляд обраних платформ.

Для більш детального дослідження вже наявних платформ було вірішено дослідити доступну у відкритих джерелах інформацію за обраними інструментами. Розглянему обрані в порядку що зазначено вище.

FREDO ДокМен — це сервіс для ведення електронного документообігу та верифікації документів. Сервіс має інтеграції з різними системами для автоматизації процесів бізнесу. Найвідомішими інтеграціями є — BAS та UA-Бюджет. BAS інтеграція в цьому випадку забезпечує основну автоматизацію бізнесу, ця технологія розроблена щоб допомогати підприємствам в ефективному керуванні різними типами своєї діяльності. Таке інтегроване програмне забезпечення допомагає вести облік та управління даними для подальшого аналізу отриманої інформації у різних галузях бізнесу. Також платформа забезпечує ефективний обмін електронними документами між контрагентами. Інтегрований інструмент автоматично завантажує та вивантажує документи за допомогою BAS. А також платформа дозволяє безпосереднє використання кваліфікованих електронних підписів, скорочено КЕП.

FREDO ДокМен використовує як основу зберігання інформації централізовані бази даних для подальшого зберігання та обробки документів. Саме це і забезпечує інтеграцію з різними системами автоматизації бізнесу. В якості інструменту для шифрування та підписання використовується інша інтеграція з системою платформи ПТАХ. Тобто за підписання підписання, шифрування і надсилання документів відповідає стороння платформа.

Також важилвою частиною використання подібних сервісів є їх користувацький інтерфейс. Який напряму може як заохотити потенційого клієнта так і відштовхнути. Для прикладу було взято скріншоти головної сторінки та частини інтерфейсу сервісу. Приклади головної сторінки та частини інтерфейсу наведено на рисунку 2.4.1 та 2.4.2. Зображення взяті з відкритого джерела інформації - https://portfel.ua/obmin-elektronnimi-dokumentami-cherez-fredo-dokmen-ru

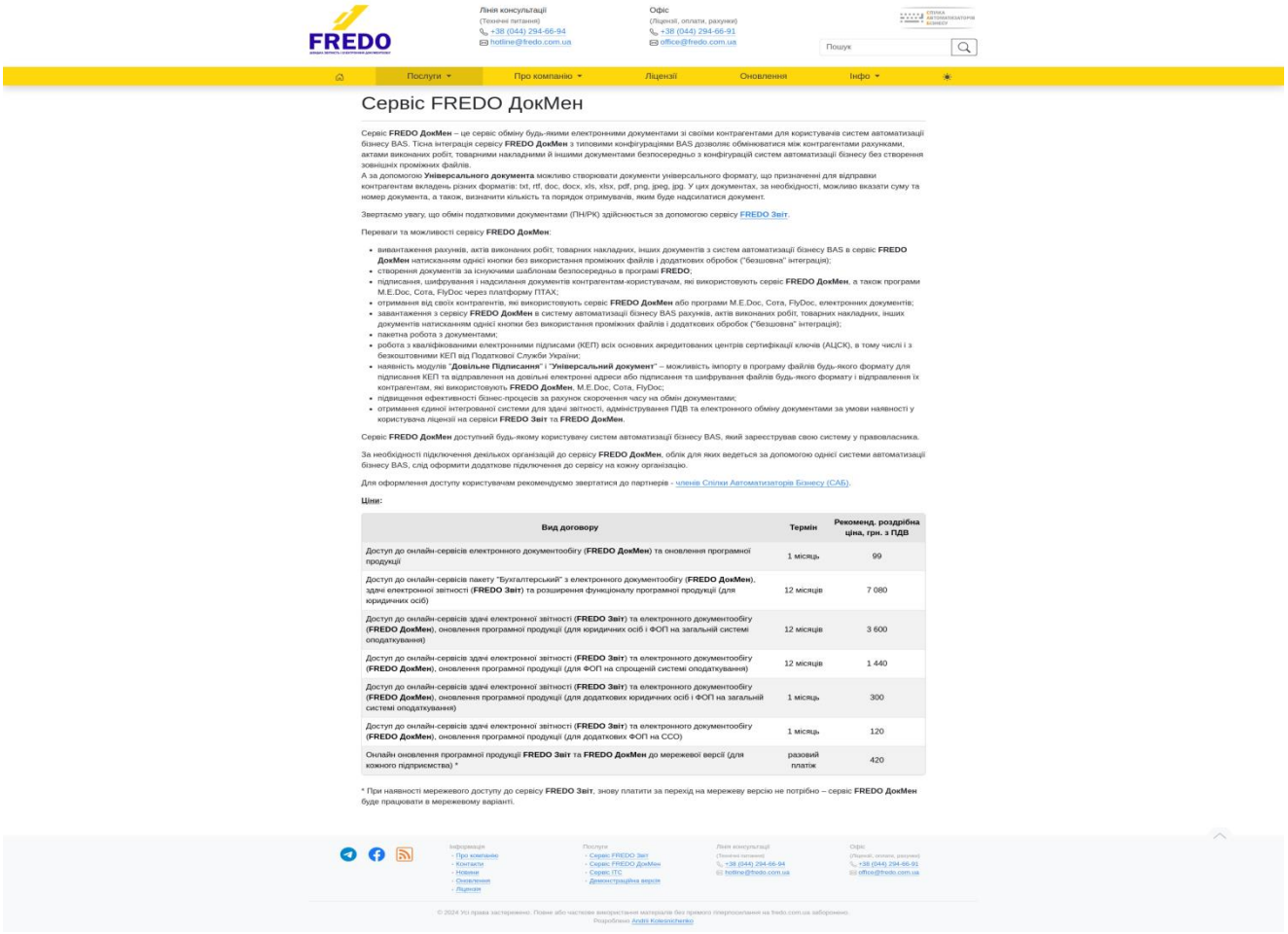

Рисунок 2.4.1 – Зображення головної сторінки сервісу FREDO ДокМен.

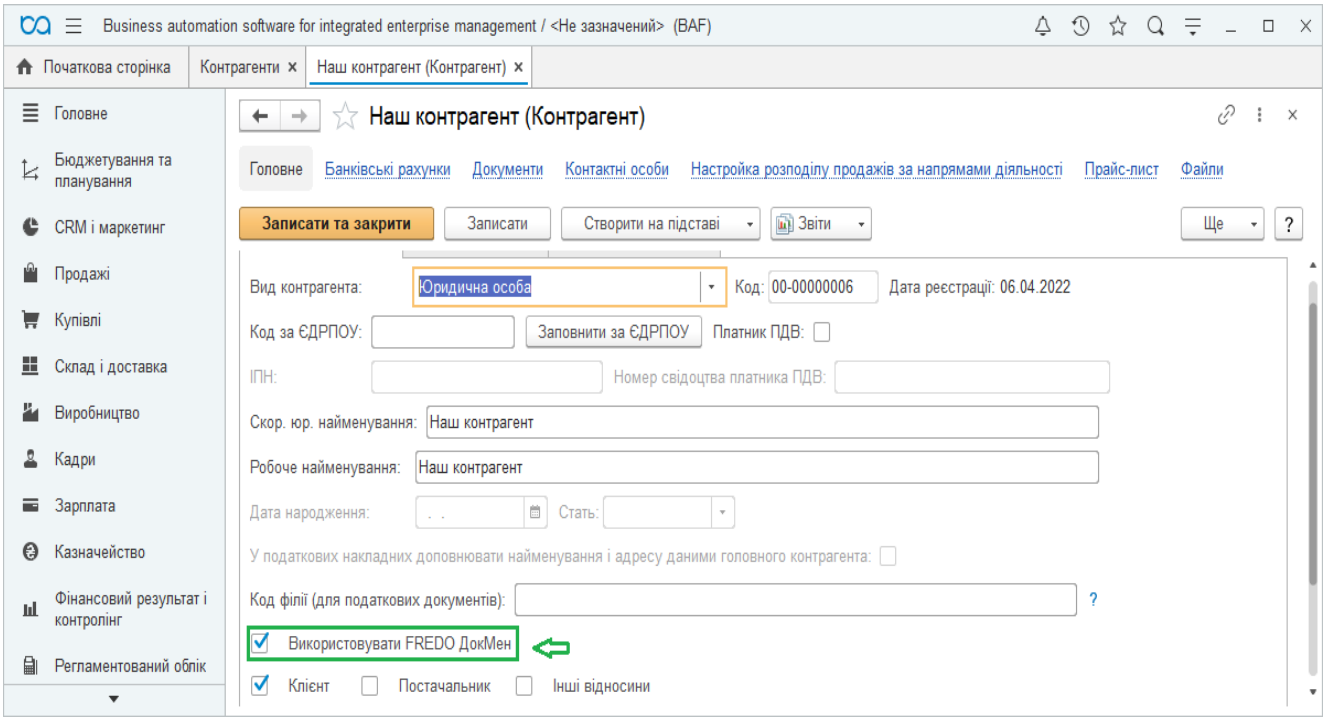

Рисунок 2.4.2 – Зображення інтерфейсу сервісу FREDO ДокМен.

СОТА також є платформою для ведення електронного документообігу та верифікації документів. Цей сервіс має майже усі інструменти що пропонує попередня платформа. Сота має інтеграції з такими сервісами BAS, Бюджет-UA, ПТАХ та інші. Але деякі специфічні можливості реалізуються в даному сервісі та є цілковитим набуттям саме сервісу СОТА.

Формати зберігання та захисту даних схожі з попердньою платформою. Використовуються великий набір різноманітних централізованих баз даних. Тобто їх захист від кібер-атак та несанкціонованого доступу повністю є відповідальністю компанії розробника.

Як і для попереднього сервісу було проаналізовано користувацький інтерфейс. Зі складових можно відмітити що даний сервіс використовую так званий підхід до створення інтерфейсу “User friendly”, що без сумнівів полегшує використання даного сервісу. Приклади вигляду головної сторінки та частини користувацьких інструментів сервісу СОТА приведено на рисунку 2.4.3 та 2.4.4.

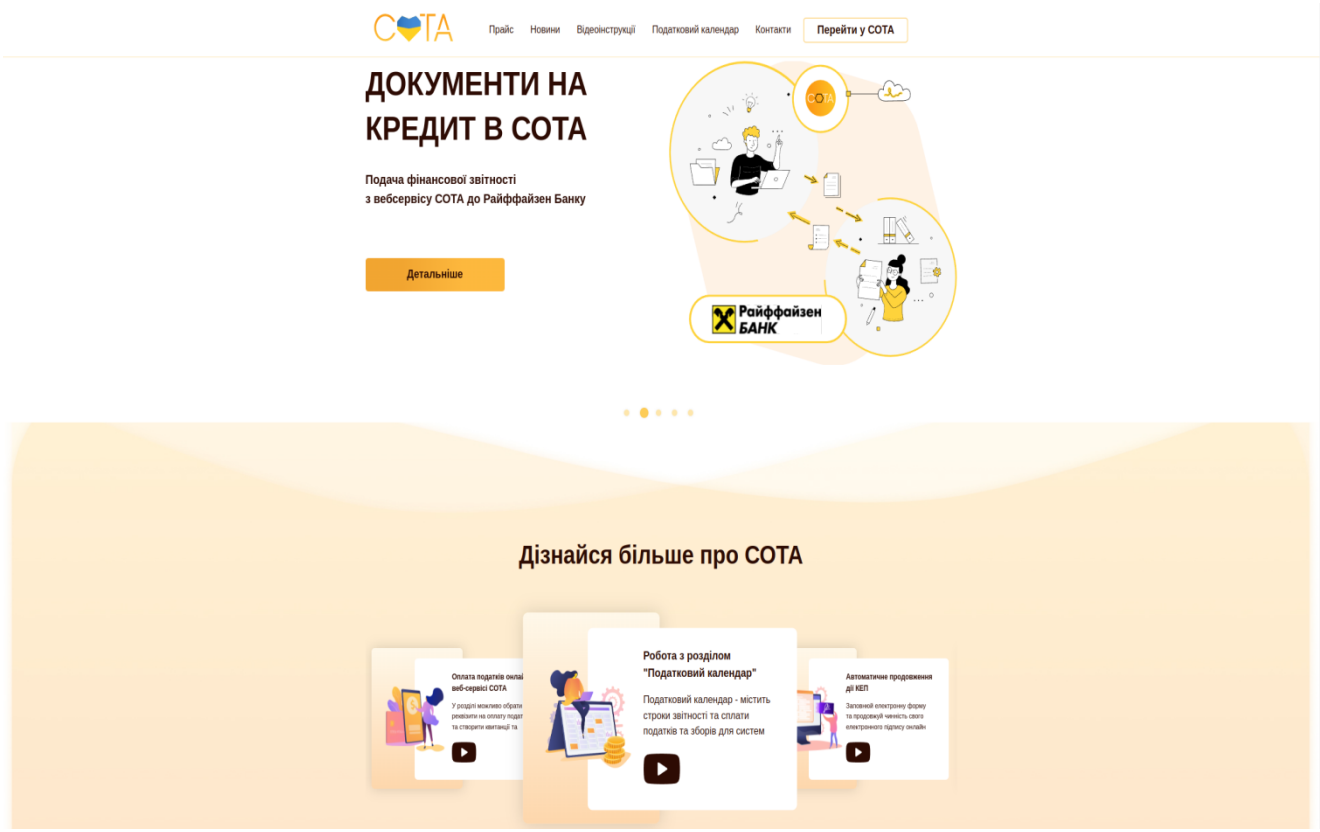

Рисунок 2.4.3 – Зображення головної сторінки сервісу СОТА.

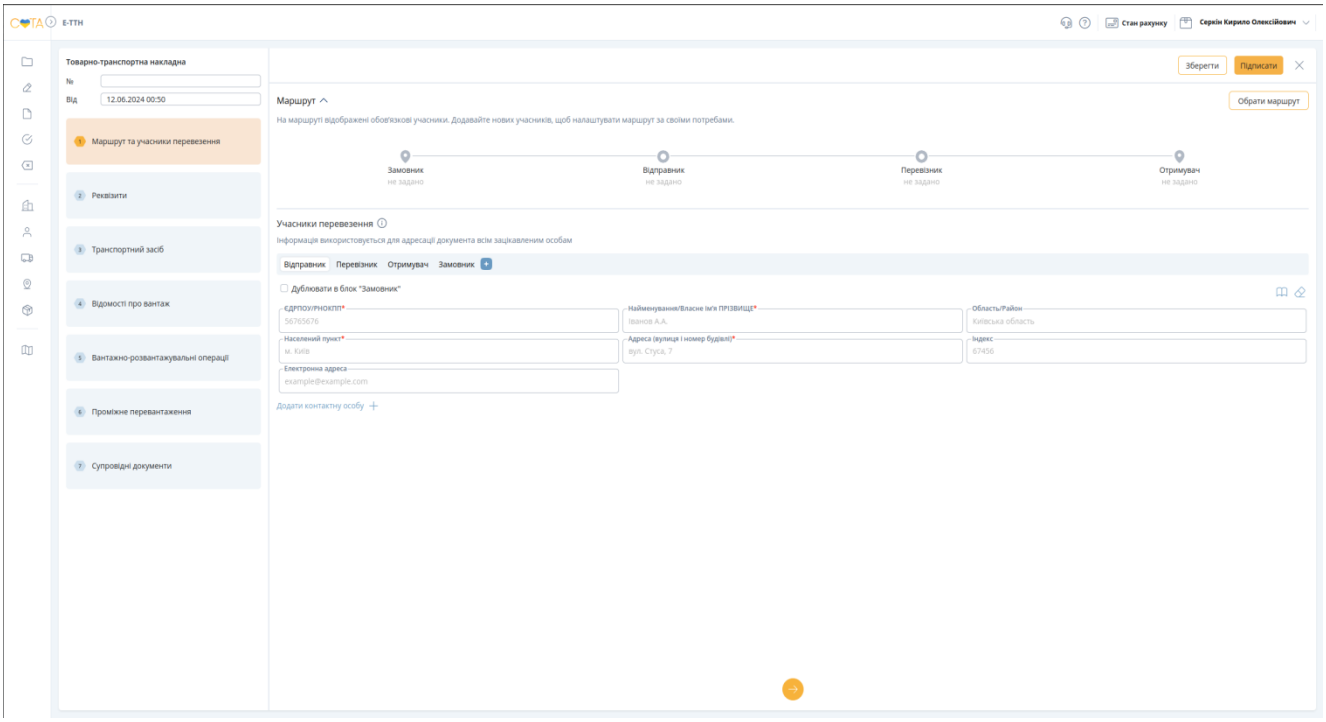

Рисунок 2.4.4 – Зображення інтерфейсу сервісу СОТА.

Вчасно — це ще один інструмент (веб-сервіс) для проведення електронного документообігу та обміну верифікованиими документми. Цей

сервіс має дуже багато спільного з попередніми рішеннями. Зокрема список технологій які інтугрує в себе сервіс Вчасно такий: BAS, Бюджет-UA, ПТАХ. Підтримка підпису документів за допомогою КЕП також включена в можливості даної платформи.

Відмінною характеристикою даного сервісу є більш легке рішення для того щоб дозволити користувачу швидку почати використовувати інструменти від Вчасно.

Усі данні також зберігаються в централізованих сховищах безпека яких є сферою відповідальності платформи, що як відомо не є надійним способом зберігання даних.

Користувацький інтерфейс має усі характеристики простої та інтуітивно зрозумілої клієнтської частини, це забезпечує низкий рівень входу для багатьох користувачів та приветрає увагу потенційних клієнтів. Нижче на рисунках 2.4.5 та 2.4.6 відповідно зображено головну сторінку платформи та частину інтерфейсу.

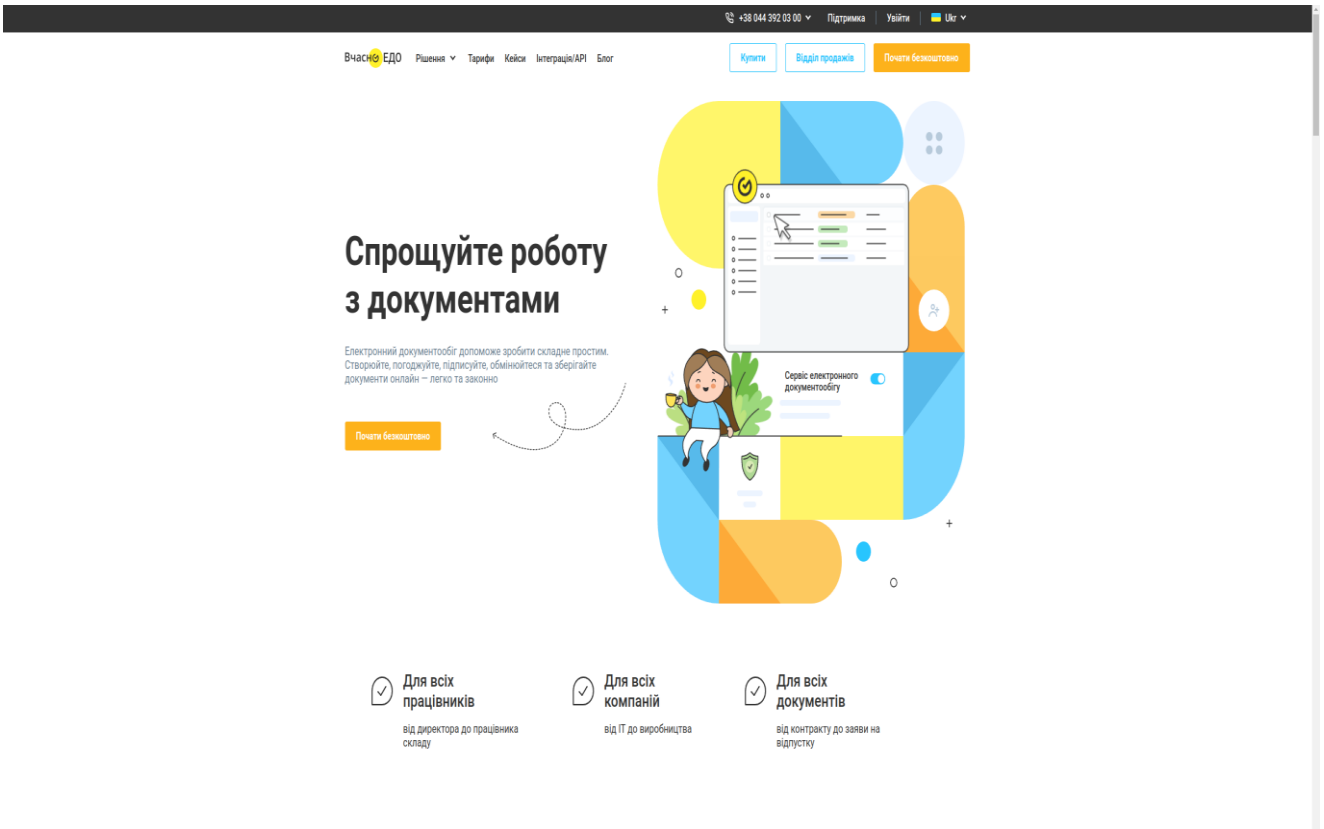

Рисунок 2.4.5 – Зображення головної сторінки сервісу Вчасно.

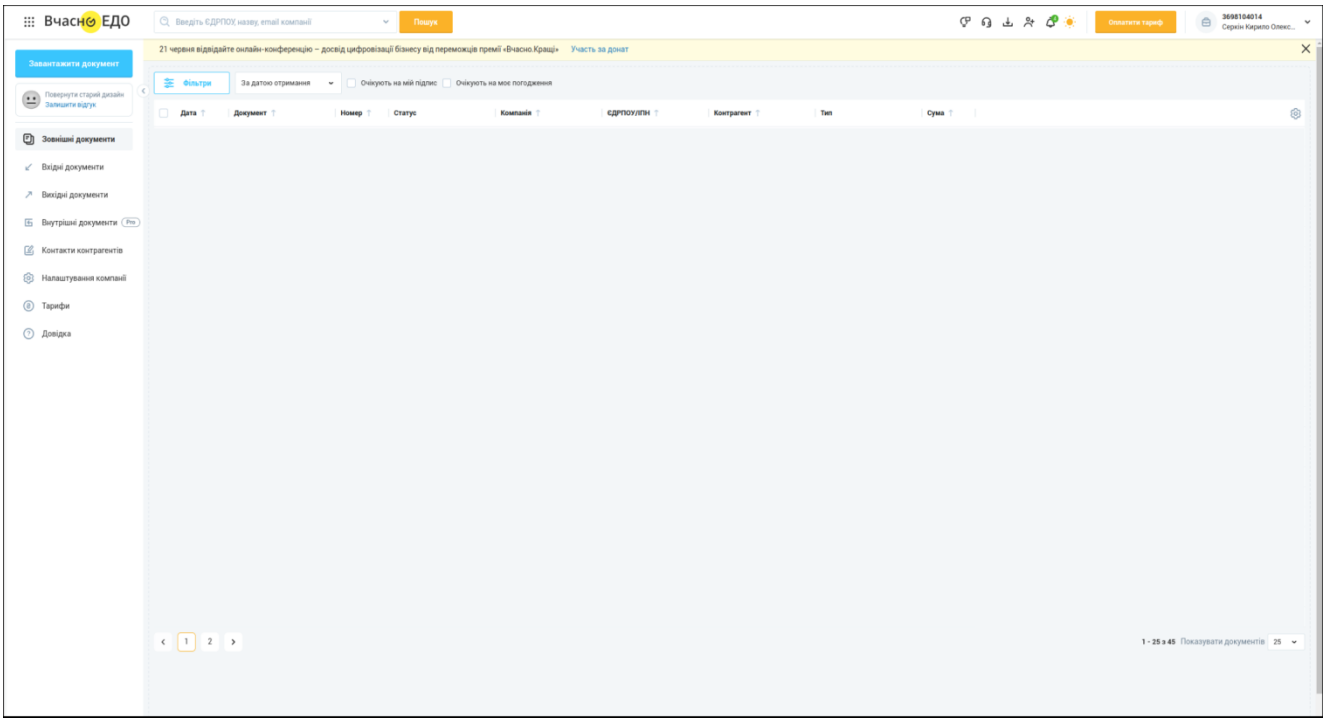

Рисунок 2.4.6 – Зображення інтерфейсу сервісу Вчасно.

2.4.2 Визначити критерії для порівняння вже існуючих рішень з потенційним рішенням на основі блокчейн технологій

Для порівняння вже існуючих платформ з потенційним рішенням на основі блокчейн технологій треба провести аналіз та визначити основні критерії оцінки. Знаходження ключових аспектів порівняння є однім з найважливіших етапів роботи, оскільки саме правильне порівняння, безпосередньо впливає на подальші дії та взагалі на доцільність розроблення рішення на базі блокчейн технологій. Критерії повинні у першу чергу відповідати на такі питання:

- Що є захисним механізмом системи;
- Хто має доступ до зміни вже існуючої інформації в системі;
- Який тип бази даних використовується, мається наувазі централізована чи децентралізована;
- Чи можливі зміни даних які вже колись були записані та затверджені.

За цими основними критеріями оцінки проведена дослідницька робота з визначення відповідей на поставлені запитання. На основі отриманих відповідей створено таблицю номер 2.4.1 для порівнянна наявних платформ документообігу та верифікації з потенційною розробкою інструменту верифікації документів за допомогою блокчейн технологій.

Таблиця 2.4.1 — Основні критерії порівняння наявних платформ з потенційним рішенням на основі блокчейн.

Найменування критерію для порівняння	FREDO ДокМен	COTA	Вчасно	Рішення на основі блокчейн технології
Що є захисним механізмом системи для належного захисту даних	Методи які були створені розробниками даної платформи	Методи які були створені розробниками даної платформи	Методи які були створені розробниками даної платформи	Механізм консенсусу який лежить в основі обраної блокчейн мережі
Хто має доступ до зміни вже існуючої інформації в системі	Доступ до зміни даних у системі має адміністратор або розробник який має ключі доступу до бази даних. В залежності від основної архітектури сервісу доступ до зміни даних може мати коло осіб які працють на компанію.	Доступ до зміни даних у системі має адміністратор або розробник який має ключі доступу до бази даних. В залежності від основної архітектури сервісу доступ до зміни даних може мати коло осіб які працють на компанію.	Доступ до зміни даних у системі має адміністратор або розробник який має ключі доступу до бази даних. В залежності від основної архітектури сервісу доступ до зміни даних може мати коло осіб які працють на компанію.	Можливо тільки додати нову інформацію, заміна старої інформації може бути реалізована тільки якщо платформа потребує цієї логіки, але навіть так історія усіх змін будет записана.
Який тип бази	Централізована	Централізована	Централізована	Децентралізован

даних використовується, мається наувазі централізована чи децентралізована				а
Чи можливо підмінити данні які вже колись були записані та затверджені	Так, оскільки база даних є централізованою.	Так, оскільки база даних є централізованою.	Так, оскільки база даних є централізованою.	Майже неможливо, завдяки механізму зберігання інформації в блокчейн мережі.

2.4.3 Проаналізувати отримані данні

В результаті аналізу трьох існуючих платформ для верифікації документів, зроблено висновок, що потреба в ефективному та достовірному інструменті для підтвердження автентичності електронних документів є актуальною. В процесі досліду було порівнянно наявні платформи за такими критеріями як: який механізм використовує система для належного захисту данних, хто має доступ до зміни вже існуючих даних системи, який тип бази даних використовується системою та чи можливо підмінити данні які вже існують та затверджені системою.

Враховуючи швидкий розвиток технологій, появу нових можливостей та загрози сучасної кібербезпеки, розглянуто можливість розробки децентралізованого рішення на базі блокчейн технологій. Зроблено висновок що блокчейн може забезпечити надійну та безпекову систему верифікації, яка може підвищіти безпеку зберігання даних та покращити прозорість процесу.

На базі аналізу блокчейн технологій та порівнянні існуючих платформ було вірішено що розробка децентралізованого інструменту для верифікації документів на основі блокчейн технологій є доцільною з кількох причин:

– По-перше, головним є те що блокчейн забезпечує високий рівень безпеки, формат зберігання даних представляє собою розподілену мережу, що робить будь-яке втручання в цілісність даних важкодоступними для зловмисників.

– По-друге, децентралізована будова блокчейну дозволяє уникати централізованого контролю або маніпуляцій з боку одніє особи або групи осіб що володіють системою.

– По-третє, за допомогою технології смарт-контрактів можна досягти автоматизації процесів верифікації, що сприяє збільшенню ефективності та підвищенню незалежності системи від зовнішнього впливу.

Отже, розробка децентралізованого інструменту верифікації документів на базі блокчейн технологій має потенціал стати новим захищеним стандартом в цій галузі, забезпечуючи безпечну, надійну та ефективну систему верифікації документів.

2.5 Архітектура додатку

Створення концепту для верифікації документів на базі блокчейн вимагає ретельного проектування архітектури та планування, щоб забезпечити захищеність даних, масштабованість платформи та надійність використання. Отже вимоги до архітектури мають включати в себе:

– захищеність вхідних даних;

– легкість масштабування;

– висока стійкість до збоїв та доступність;

– можливість легкої інтеграцій з іншими сервісами;

– оригінальність та неповторюваність підписів користувача.

Захищеність вхідних даних передбачає що внутрішні сервіси системи не повинні ніяким чином змінювати або модифікувати документи які завантажує користувач й гарантувати їх цілісність в момент завантаження та збереження. Всі процеси обробки та формування даних перед записом верифікаційноїго

підпису мають бути спрощені та відбуватися здебільшого в середині блокчейн мережі.

Легкість масштабування означає здатність програмного забезпечення витримувати періодичні або постійні збільшення навантаження, розширюючи свої ресурси або додаючт нові. Розширення може бути як горизонтальним, що передбачає додавання більшої кількості копій сервісів, так і вертикальним, коли ресурси додаються до існуючих сервісів. Веб-застосунок повинен бути розроблений так, щоб легко підтримувати обидва види розширення.

Висока стійкість до збоїв та доступність - для цього архітектурного фактору блокчейн мережа на які буде збудовано веб-додаток, має бути обрана заздалегідь таким чином щоб в рамках взаємодії була забезпечена стійкість до збоїв та доступність у будь-який момент часу. Також характер побудови архітектури в такому випадку має бути гнучким та легким до перенесення на будь-яку іншу сумісну блокчейн мережу. Тоже обрана блокчейн мережа має мати перевірине ядро яке використовують та дворіють багато наявних блокчейн мереж.

Можливість легкої інтеграцій з іншими сервісами означає що основное ядро додатку має бути універсальним в контексті інтеграції з іншими додатками. Тобто клієнтська частина не повинна залежати від механізму верифікації та навпаки. Для сумісності з будь-яким сервісом потрібно розробити гнучкий та максимально зрозумілий механізм, результат роботи якого можна буде використовувати для інтеграції на різних рівнях системи.

Оригінальність та неповторюваність підписів користувача - повинна бути гарантована безпосередньо самими механізмом блокчейн технологій без впливу сторонніх осіб. Підписи повинні бути неповторними та генеруватись завдяки факторам на які не можуть впливати ані сервіси з якими інтегрується додаток, ані інші користувачі додатку чи блокчейн мережі.

Створення рішення на базі блокчейн наразі представляє собою своєрідний симбіоз технологій. Блокчейн мережа представить нам єдине джерело достовірних даних. Клієнтська частина стане місцем яке буде забезпечувати зв’язок з блокчейн мережею, для обміну даних в двох напрямках.

А сервена частина архітектури буде слугувати у якості сховища зі швидких доступом до даних зареєстрованих користувачів. Серверна централізована база даних буде просто копією бази даних що знаходиться у блокчейн мережі й буде синхронізована зі сховищем смарт-контракту що знаходиться в блокчейн мережі.

2.6 Вимоги до зберігання даних в блокчейн мережі

Смарт-контракт який буде використано як ядро системи не повинен зберігати чутливої особистої інформації, щоб у контексті публічного блокчейну ніхто не мав змоги отримати дані про документи іншого користувача, котрі потім можна буде використати для особистих цілей зловмисників. Основними даними які будуть зберігатись в так званому стейті або іншими словами сховищі смарт-контракту будуть:

– Nonce - унікальний та неповторний внутрішній номер транзакції верифікованого документа(-ів). Такий номер постійно збільшується та виступаю в ролі id кожного підпису;

– Signature - сигнатура, оригінальний та неповторний підпис користувача;

– RootHash - хеш який буде сформовано за допомогою дерева Меркла побудованого на базі хешів зашифрованих документів користувача;

– UserSignes - це хеш таблиця значень підписів користувчив на випадок якщо документ або набор документів потребує підпису одруз декількох користувачів, підписати такий документ або їх набор зможе лише обмежена група осіб;

– Timestamp - штамп дати й часу коли була записана інформація.

У випадку якщо набір документів буде відправлений користувачеві на підписання, тобто у рамках ділових відносин тощо. Впроваджено поле яке можна змінити лише один раз. В середені змінної UserSignes зазначеного у 4 пункті списку що був описаний вище знаходиться особливе поле isSigned - поле показувати чи підписав користувач або група користувачів даний документ.

Таке поле буде булевою змінною, та спочатку буде завжди мати стан “false”, змінити цей статус на “true” зможе тільки користувач якому було відправлено ці документи на підписання.

2.7 Вимоги до користувацького інтерфейсу

Інтерфейс користувача повинен бути інтуїтивно зрозумілим і легкоим для сприйняття. Елементи управління мають бути послідовними і зрозумілими. Макети різних сторінок повинні зберігати послідовність та витримку єдиного стилю. Елементи управління, які виконують схожі функції, слід групувати для полегшення навігації користувача. Текстові та графічні елементи повинні мати достатню контрастність і бути чіткими для полегшення сприйняття контенту. Стиль інтерфейсу повинен бути мінімалістичним та лаконічним без зайвих нагромаджень які будуть заважати або заплутувати користувача. Логотип повинен бути унікальним і мати виразний символічний елемент. Усі зображення та шрифти мають бути створені самостійно або взяті з відкритих джерел.

3 ВИЗНАЧЕННЯ АЛГОРИТМІВ РОЗРОБЛЮВАНОЇ СИСТЕМИ

3.1 Опис алгоритмів головних бізнес-процесів розроблюваного інструменту верифікації на базі блокчейну

У контексті бізнес-процесу розроблюваного інструменту мається наувазі веб-додаток який включає в себе розробку ключових кроків та аналіз діяльностей, які є необхідними для досягнення конкретної мети в рамках використання інструменту для верифікації документів. Основною метою осіб що будуть використовувати веб-додаток з точки зору розробки є реалізація можливості створення документів котрі після завантаження повинні бути підписані групою осіб або створення можливості для користувачів верифікувати документи для подальшого доказу інсування даних документів в незмінному вигляді. Основними Бізнес процесами, що мають сприяти досягненню основної мети – є:

- можливість реєстрації та авторизації в системі;
- підключення різних типів крипто-гаманців;
- можилвість завантаження документів для подальшої верифікації;
- створення документів для підписання групою осіб;
- верифікація вже підписаних документів та їх перевірка на незмінність.

Процеси реєстрації та підключення крипто-гаманця продемонстровано на рисунку 3.1.

Наразі оскільки цей інстурмент є лише концептом при реєстрації необхідним є ПІБ, номер телефону, електронная пошта та підключений крипто-гаманець. ПІБ буде ідентифікувати особу. Електронна пошта буде використовуватись як система сповіщень якщо користувачу прийде запит на підписання документів. Крипто-гаманець буде виступати в ролі оригінального та непідробного підпису оскільки публічний ключ крипто-гаманця є унікальним та ніколі не повторюється. Індивідуальний код платника подактів потрібен для реалізації можливості додавння осіб котрих можна буде додати для підписання документів як сторону контрагента тощо. Звісно при

подальшому розвитку ідеї веб-додатку буде додана ідентифікація особистості за допомогою таких інструментів як “Дія” або bankID, наразі це не є можливим варіантом оскільки це надає доступ до чутливої інформації.

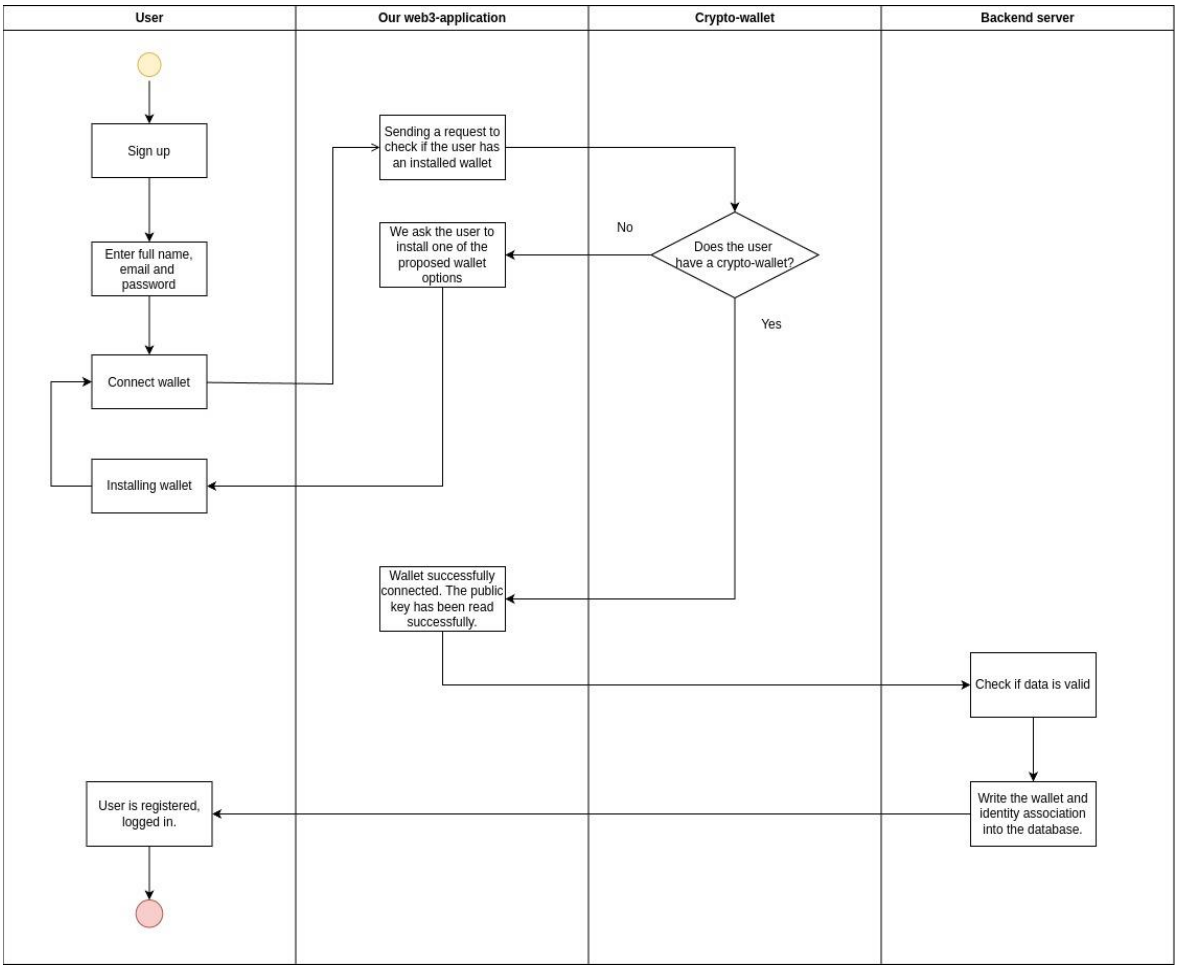

Рисунок 3.1 – UML діаграма реєстрації та підключення крипто-гаманця

Після реєстрації та підключення крипто-гаманця користувач має доступ до інструментів які предумовлені системо. Користувач може завантажити свої документи та верифікувати їх створивши доказ їх існування в смарт-контракті блокчейн мережі. Після верифікації документів користувач може бачити які документи вже були підписані за допомогою його оригінального підпису.

Процес створення заявки на верифікацію наведено на рисунку 3.2. Даний процес реалізовує вимогу «можилвість завантаження документів для подальшої верифікації». У цьому випадку розглядається поведінка системи коли користувач використовує можливості інструменту для процесу схожого на підписання документів електронним ключем.

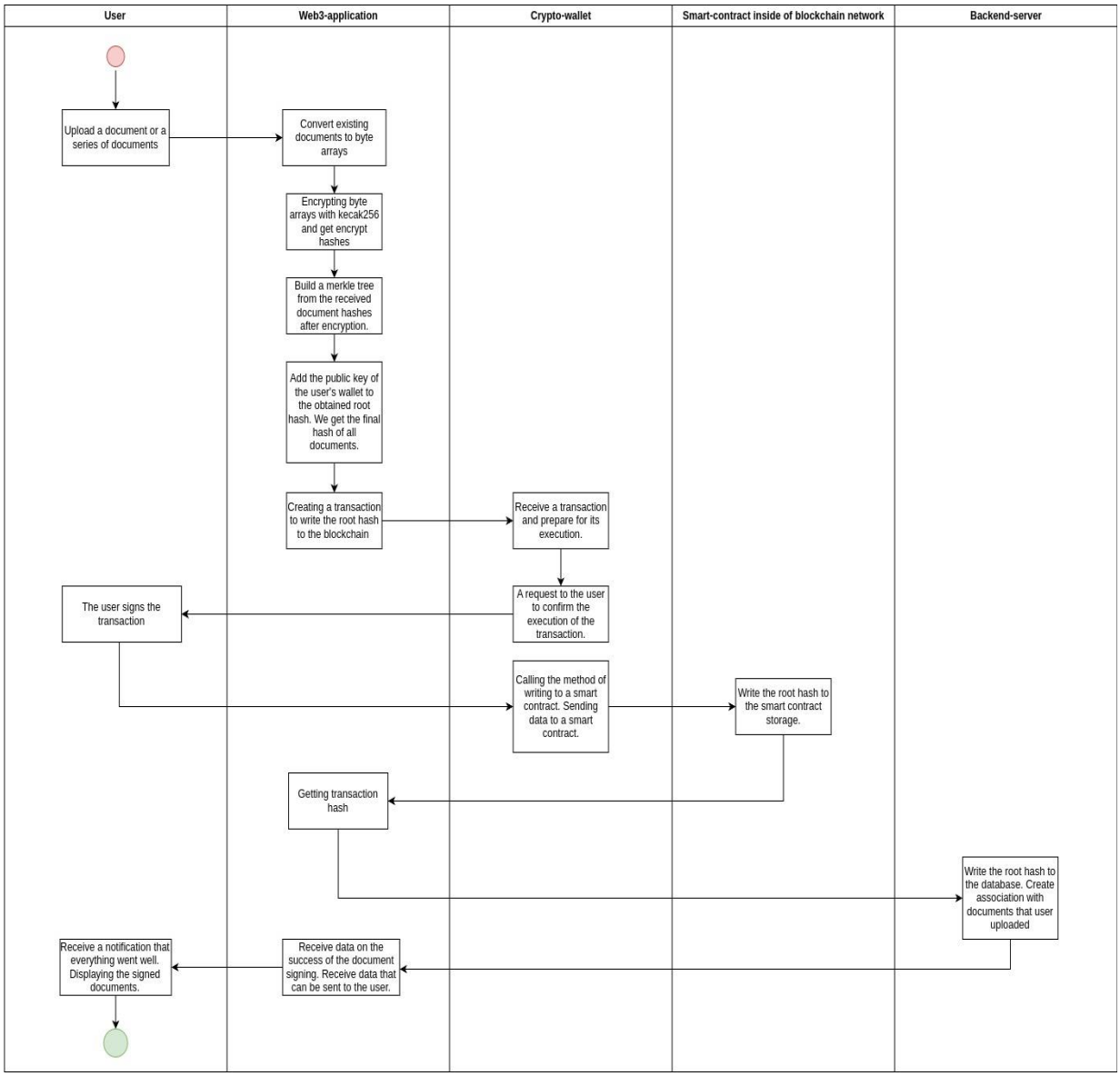

Рисунок 3.2 – UML діаграмма завантаження та підписання документів

Реалізація бізнес-процесу створення документу або низки документів для підписання групою осіб, є більш ускладненим варіантом попереднього процесу. Для набуття валідності документу або низки документів в разі підписання групою осіб користувачу треба пройти декілька зазначених етапів:

– Завантаження низки документів як у варіанті з бізнес процесом персональної верифікації та підписання документів. Схема цього процесу зображена вище на рисунку 3.2;

– Пошук та додавання користувачів чий підпис буде потрібен для набуття валідності документу або їх групи. Для пошуку використовується індивідуальний код платника податків;

– Відправка документів на підписання. Реалізація цього етапу описана вище оскількі бізнес-процес майже однаковий за винятком того що для набуття валідності треба підписання усіх зазначених користувачів.

Для додавання валідного контрагенту до підписання низки документів реалізовано пошук наявних контрагентів платформи. Пошук осіб які зареєстровані на платформі є необхідним процесом для додавання контрагента як сторони підписання низки документів. Процесс пошуку відбувається за індивідуальним номером платника податків.

Схема реалізації алгоритму пошуку контрагентів зображено на рисунку 3.3. На цих діаграмах фактично зазначається як має бути реалізована поставлена вимога можливості пошуку ініціатив та фахівців.

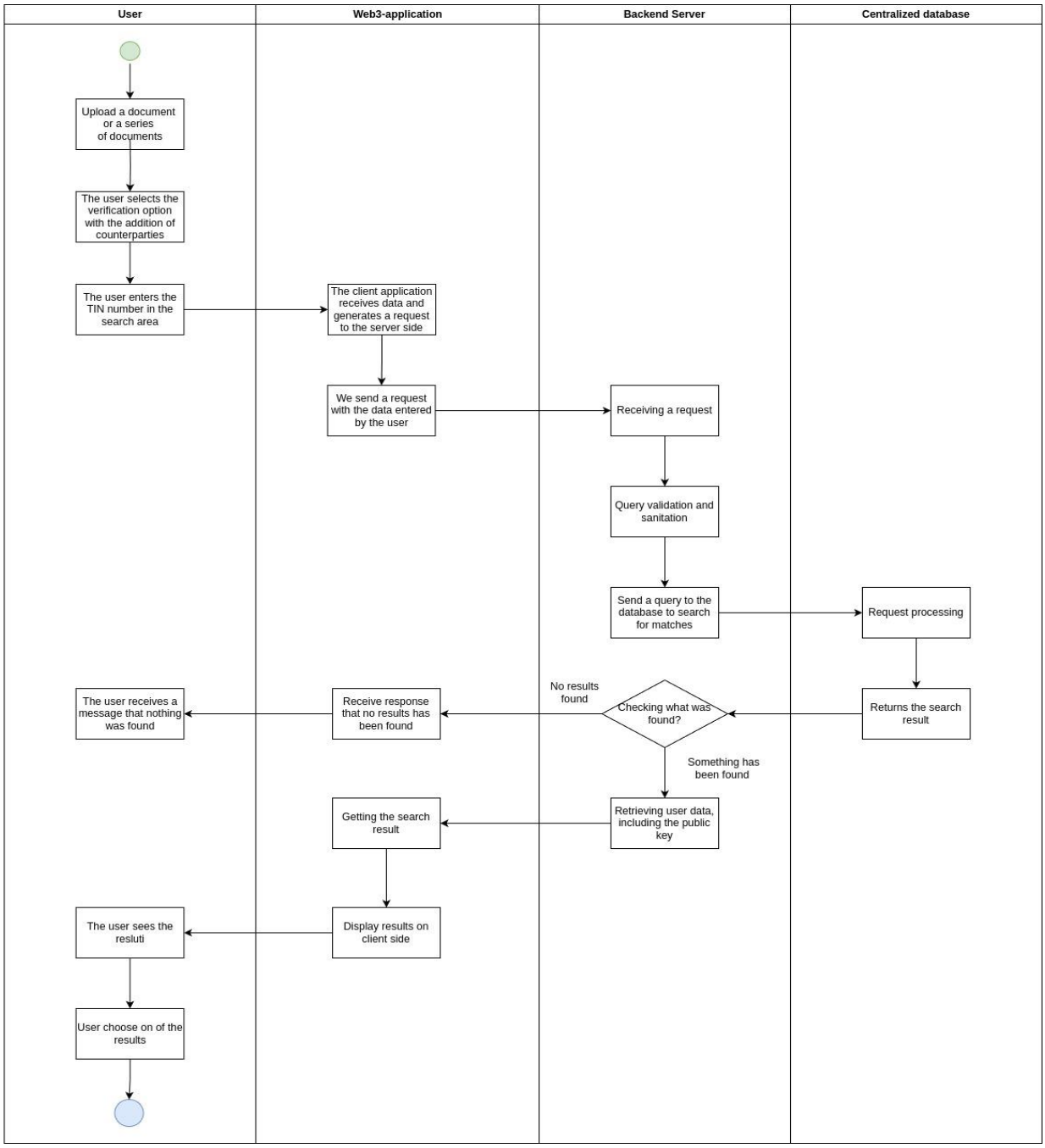

Рисунок 3.3 – UML діаграма алгоритму пошуку контрагентів

Після того як документи були завантажені та було обрано усіх потрібних контрагентів. Починається процес шифрування документів та додавання їх до сховища смарт-контракту як у випадку зі звичайною верифікацією документу за винятком того, що до сховища в блокчейн мережі буде додана особлива структура. Ця структура буде включати в себе усі публічні ключі користувачів що були зазначені контрагентами, задля забезпечення підписання документів тільки визначеним колом осіб. Тобто особа яка не була обрана як сторона договору не може поставити свій підпис, відкликати підпис неможливо.

Схема реалізації процесу завантаження та верифікації документів для підписання групою осіб показана нижче на рисунку 3.4.

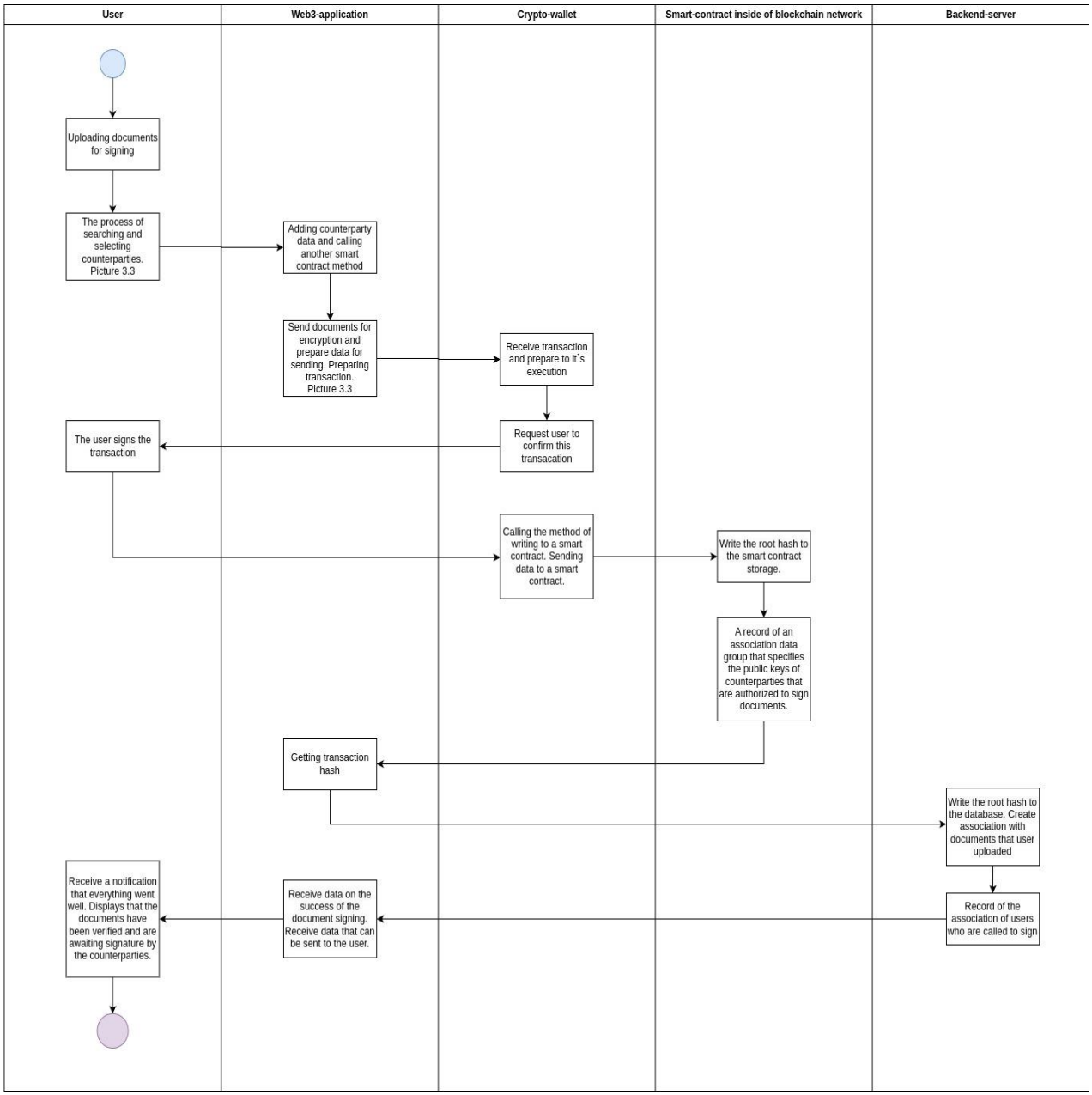

Рисунок 3.4 – UML діаграма процесу завантаження та верифікації документів для підписання групою осіб

Бізнес-процес отримання інформації особою що він є стороною підписання документу зображено на рисунку 3.5.

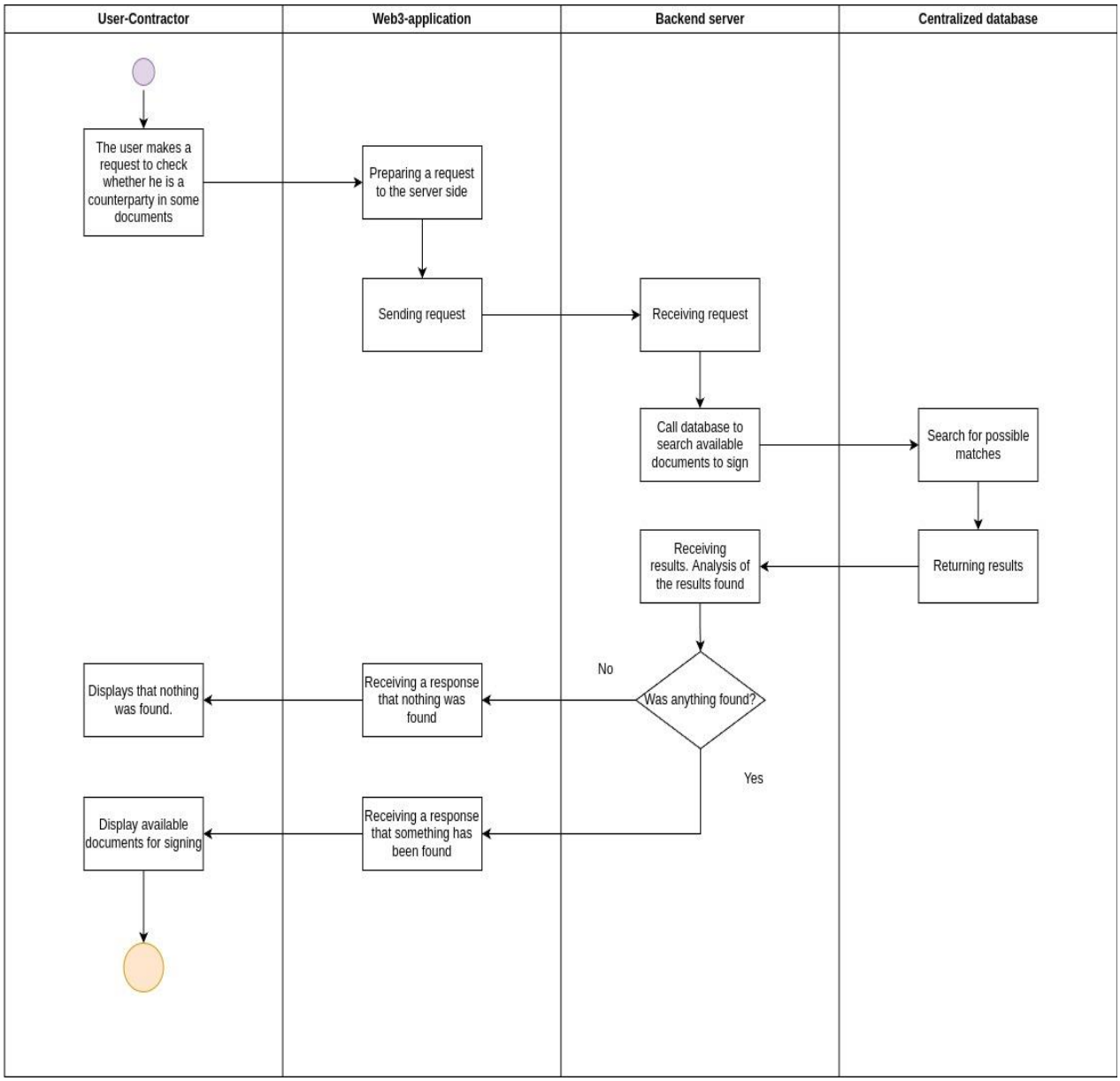

Рисунок 3.5 – UML отримання інформації що особа є стороною підписання документу.

Безпосередньо підписання особою що є контрагентом будь-якого документу або низки документів зображено на рисунку 3.6.

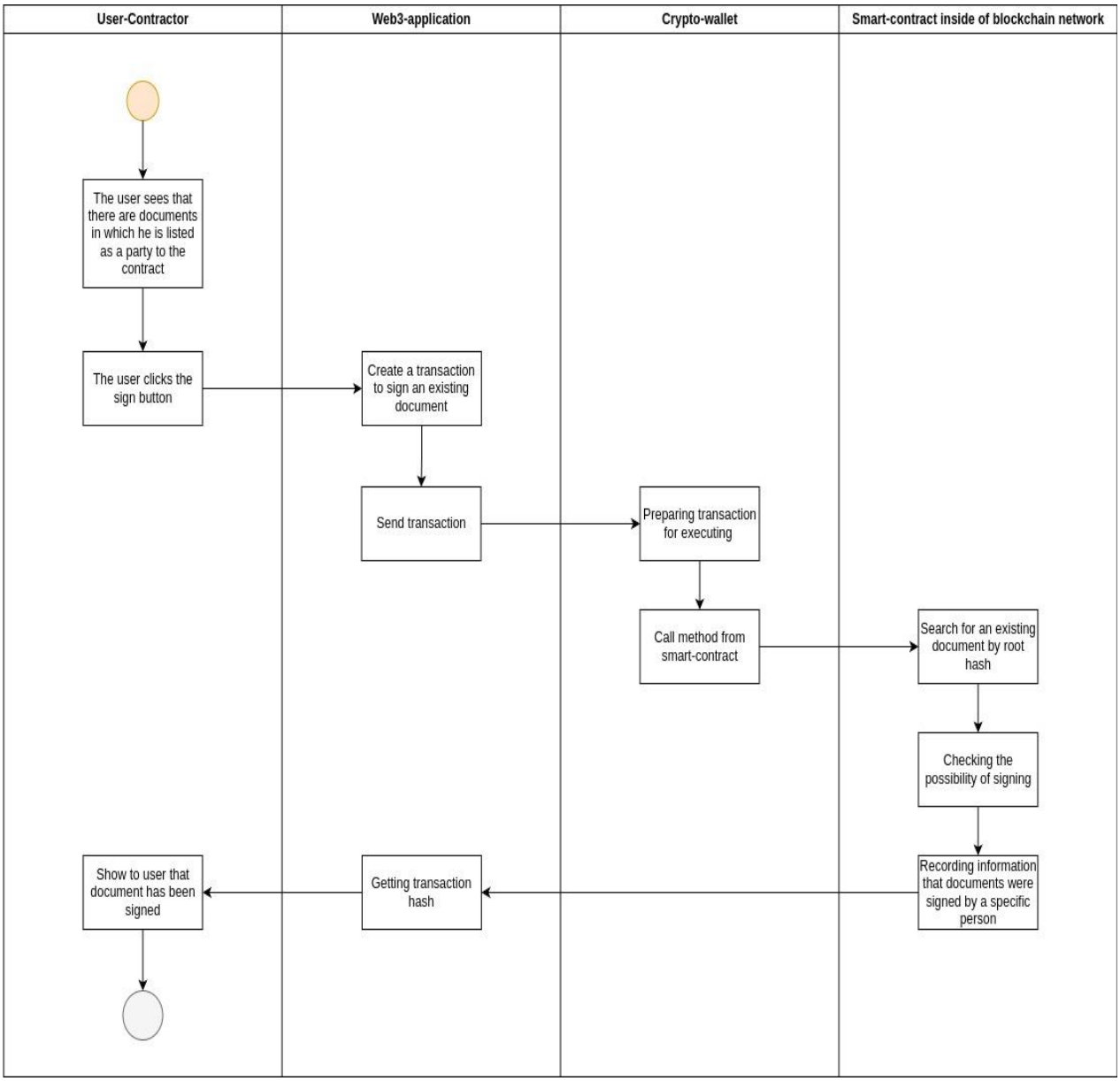

Рисунок 3.6 – UML підписання особою що є контрагентом будь-якого документу або низки документів.

Дані про статус документів та сторін підписання документу або низки документів зберігаються в середені смарт-контракту, який розташований у блокчейн мережі. Для більш швидокого доступу до даних та передачі не чутливої інформації існує копія цих даних в централізованій базі даних.

Такак міра є необхідною задля того щоб не звертатись до блокчейн мережі кожен раз коли нам потрібна якась інформація, тим паче чим більш простий функціонал буде реалізовувати смарт-контракт тим краще для захисту.

Централізована база даних буде представляти з себе копію даних які зберігаються у блокчейн мережі й звіряти наявні данні з даними в мережі блокчейн для підтримки оригінальності інформації.

Метою введення даного функціоналу є збереження коштів на утримання потужностей розроблюваного інструменту, а також підвищення швидкості отримання даних в моменти коли блокчейн мережа може бути під навантаженням.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Архітектура застосунку

Дивлячись на вимоги до архітектури застосунку, що включають масштабованість, гнучкість та балансування навантаження є доцільним реалізація мікросервісної архітектури. Мікросервісна архітектура зараз є найбільш розповсюдженою практикою для подальшого розширення та масштабування проектів.

Основною ідеєю мікросервісної архітектури є використання окремих сервісів які передають данні та спілкуються між собою, замість створення великого монолітного застосунку. Монолітність проекту ускладнює розробку та підтримку застосонку при подальшому масштабуванні. Натомість використання великої кількості малих сервісів надає можливість роздилити функціональність додатку та рівномірно розподілити наватаження яке бути отримувати система.

Використовуючи такий підхід знімається такі обмеження як:

- Використання однієї мови програмування для всіх інструментів застосунку;
- Використання однієї бази даних одного типу для отримання даних різного походження;
- Інкапсуляція одного сервісу від іншого в середені одного моноліту.

Взаємодія між сервісами обмежується лише обраними підходами комунікацій: HTTP, GRPC або через збережені в базі даних процедури виконання.

З огляду на основні ідуї бізнес-процесів та бізнес задачу в цілому, а також зважаючи на загальновизнані підходи побудови мікросервісної архітектури вдалося визначити потребу в наступних сервісах:

— Golang Gin server – буде використовуватись як API Gateway сервіс;

— Prometheus – мікросервіс, відповідальний за збирання метричних даних усіх мікросервісів системи;

— PostgreSQL — реляційна база даних яка буде відповідати за зберігання даних в таблицях з побудованими відносинами;

— Elasticsearch — база даних що зберігає усі данні у віді документів;

— React Truffle Application — клієнтська частина додатку що відображає UI частину та забезпечує взаємодію з блокчейн мережею.

Golang Gin server – сервіси, є відповідальним за реалізацію точки входу так званий Gateway.

Головною ідеєю Gateway є - абезпечення єдиної точки входу до системи в той час як система представляє собою якусь кількість мікросервісів. Усі запити з клієнтської частини спочатку надходять до так званого Gateway, який потім перенаправляє ці запити до потрібних мікросервісів. Такий підхід в рази спрощує архітектуру з точки зору клієнта, та в одночас дозволяє реалізувати централізоване управління безпекою, балансуванням навантаження, логуванням та іншими аспектами системи.

Одним з важливих факторів архітектурного впровадження даного рішення є - забезпечення необхідного рівня абстракції між клієнтською частиною та серверної частиною застосунку. Це значить, що клієнтському застосунку не треба знати щось про внутрішню роботу сервісів, персональні точки входу, порти, хости і тд. Натомість вони покладаються на API шлюз для обробки запитів та отримання відповідей.

В такому випадку значно полегшується процес масштабування системи, тому що додавання чи зміна сервісів не потребує змін у клієнтського додатку. Одночасно з цим виникає ризик отримання такої вразливості системи як єдина точка входу. Цю проблему можна вирішити при правіильному використанні проксі-серверу та балансуванню навантаження на контейнери сервісів.

Типовий вигляд архітектури такого підходу наведено на рисунку 4.1.1.

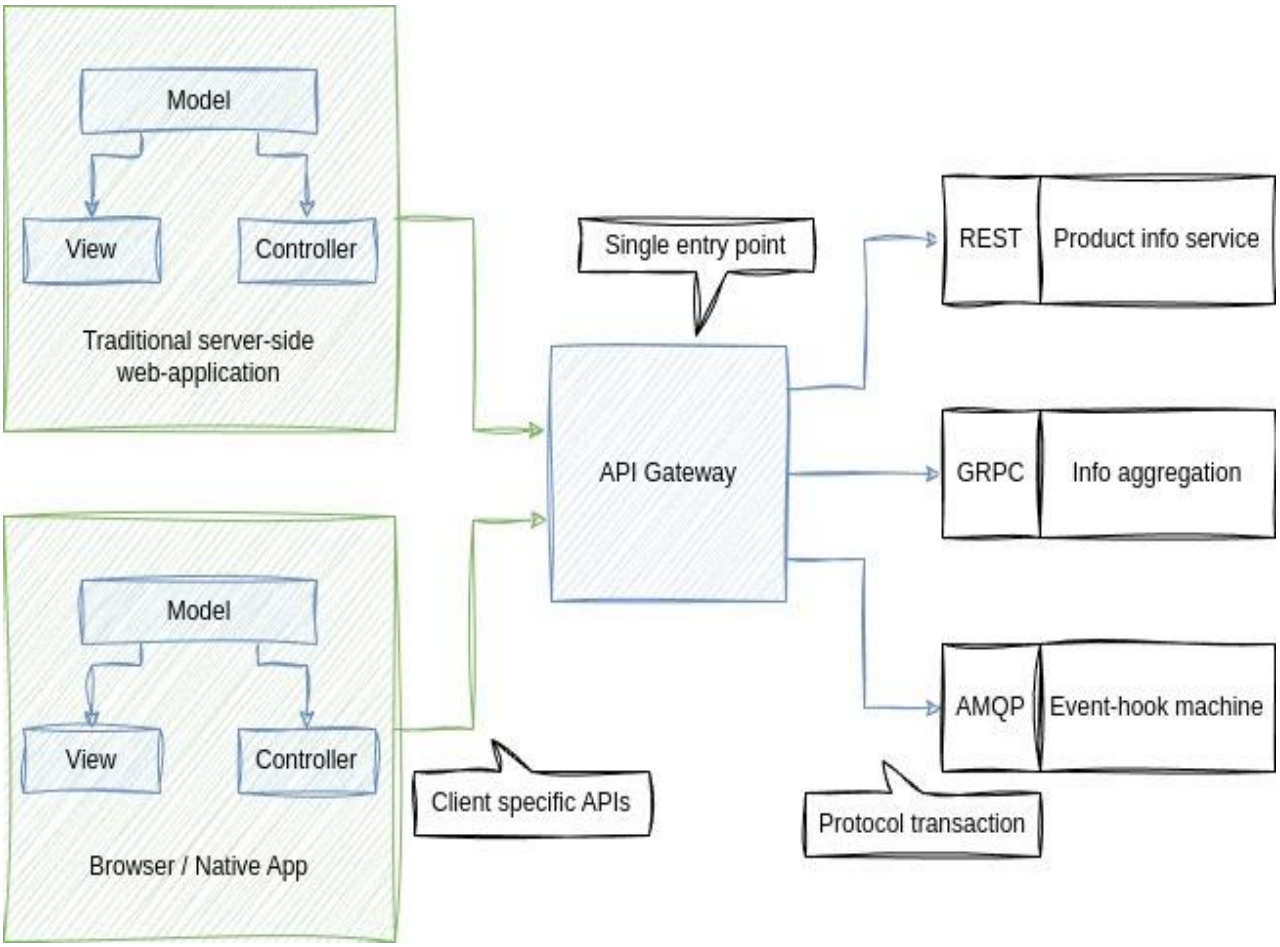

Рисунок 4.1.1 – Типове використання API Gateway архітектури.

Другим архітектурним рішенням було визначенно використання Kubernetes для горизонтального масштабування та роутінгу трафіку. Kubernetes являє собою потужну платформу для оркестрації контейнерів, він забезпечує автоматизацію, масштабованість та управління життєвим циклом сервісів. В Kubernetes є система забезпечення горизонтального масштабування яка забезпечує динамічне розширення кількості активних сервісів на основі аналізу навантаження на конкретний сервіс. Це досягається за допомогою спеціального механізму, такого як Horizontal Pod Autoscaler (HPA), який контролює та відстежує показники навантаження (наприклад, використання ЦП або пам'яті) і на основі цих даних створює новий екземпляр сервісу для того щоб розподілити між ними навантаження. Уявимо що в якийсь момент у нас буде дуже багато запитів до нашого backend сервісу. В залежності від налаштувань, коли використання ресурсів цим сервісом досягне 50% від можливостей нашого центрального процессору, Kuberenetes автоматично створить копію такого

самого backend сервісу та буде розподіляти запити вже між двома сервісами. Схема балансування навантаження та реплікації сервісів зображена на рисунку 4.1.2.

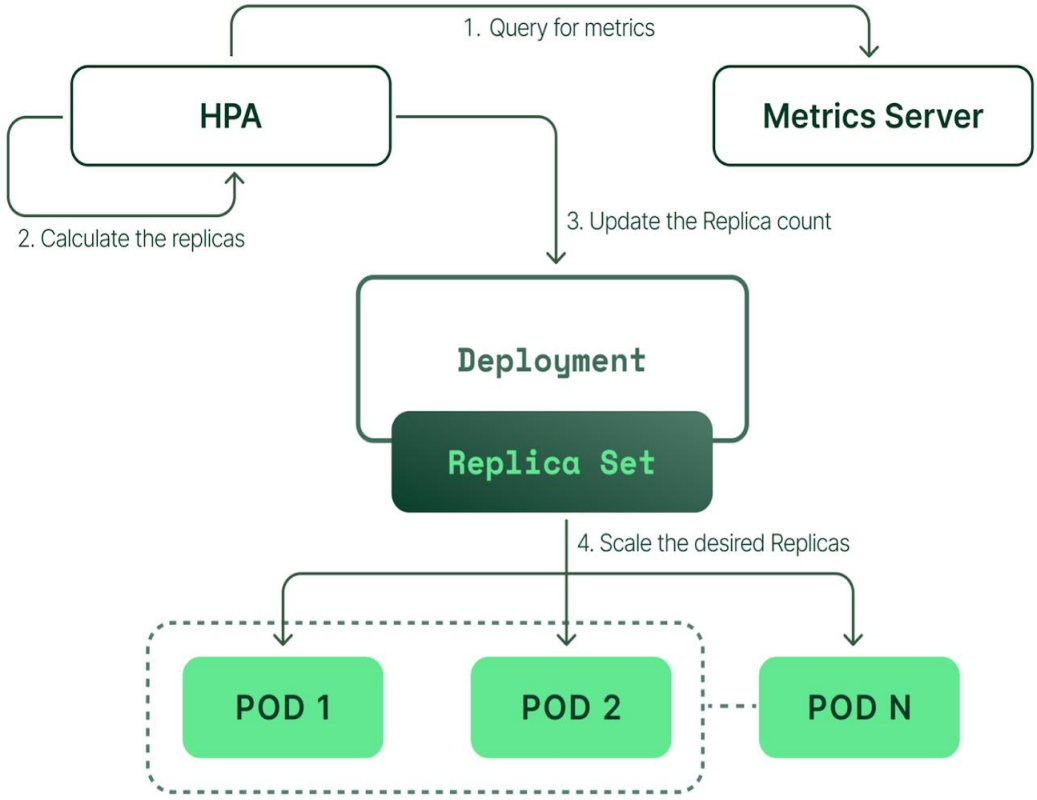

Рисунок 4.1.2 – Схема балансування навантаження та реплікації сервісів.

На основі схеми балансування та при використанні оркестратора контейнерів Kubernetis було розроблено спеціальний маніфест котрий виступає в ролі файлу конфігурації для масштабування серверної частини додатку. Масштабування має такі умови розширення що при навантаженні на процесор сервісу вище 50 відсотків, буде автоматично розгорнуто ще один екземпляр данного сервісу. Максимальною кількістю реплік такого сервісу є 10 екземплярів одночасно. Код розробленого манфісту наведено в лістингу коду 4.1.1.

```
apiVersion: autoscaling/v2
kind: HorizontalPodAutoscaler
metadata:
  name: document-verify-backend-scalling
spec:
  scaleTargetRef:
    apiVersion: apps/v1
    kind: Deployment
    name: document-verify-backend
  minReplicas: 2
  maxReplicas: 10
  metrics:
    - type: Resource
      resource:
        name: cpu
        target:
          type: Utilization
          averageUtilization: 50
```

Лістинг коду 4.1.1 — Манфест автоматичного масштабування серверної частини додатку.

Третє архітектурне рішення стосується вибору блокчейн мережі у якій буде розгортатися частина системи. Це є майже найважливішим кроком який напряму впливає на дуже велику кількість факторів. Зокрема до таких факторів відноситься вартість розробки, тестування та виконнання транзакцій. Швидкість завантаження даних в мережу та швидкість отримання інформації з мережі блокчейн. Найважливішим чинником що впливає на вибір блокчейн мережі є безпека блокчейн мережі, зазвичай безпека мережі корелює з її розміром та кількістю користувачів.

Вибором стала блокчейн мережа Arbitrum котра є мережею другого рівня для масштабування основної і всім відомої мережі Ethereum яка є гарантом надійності та піонером функціональних блокчейнів загального призначення. Тобто арбітрум є частиною мережі Ethereum та усі транзакції затверджуються в основній мережі та в одночас є більш дешевими через технологію optimistic rollups.

Завдяки цій технології Arbitrum забезпечує свою надійність та безпеку. Такий підхід дозволяє зменьшити витрати на виклик транзакції та їх обробку, але основною перевагою є те що більшість обчислень відбуваються поза

основною мережею, а транзакції групуються в так звані ролапи. Ролапом є великий набір інформації про зміни в мережі зокрема в смарт-контрактах, але цей набір інформації оптимізовано, що прибирає більшу частину надлишкових даних. Ці ролапи потім відправляються до основного ланцюга для підтвердження.

Зображення роботи технології optimistic rollups зображено на рисунку 4.1.3.

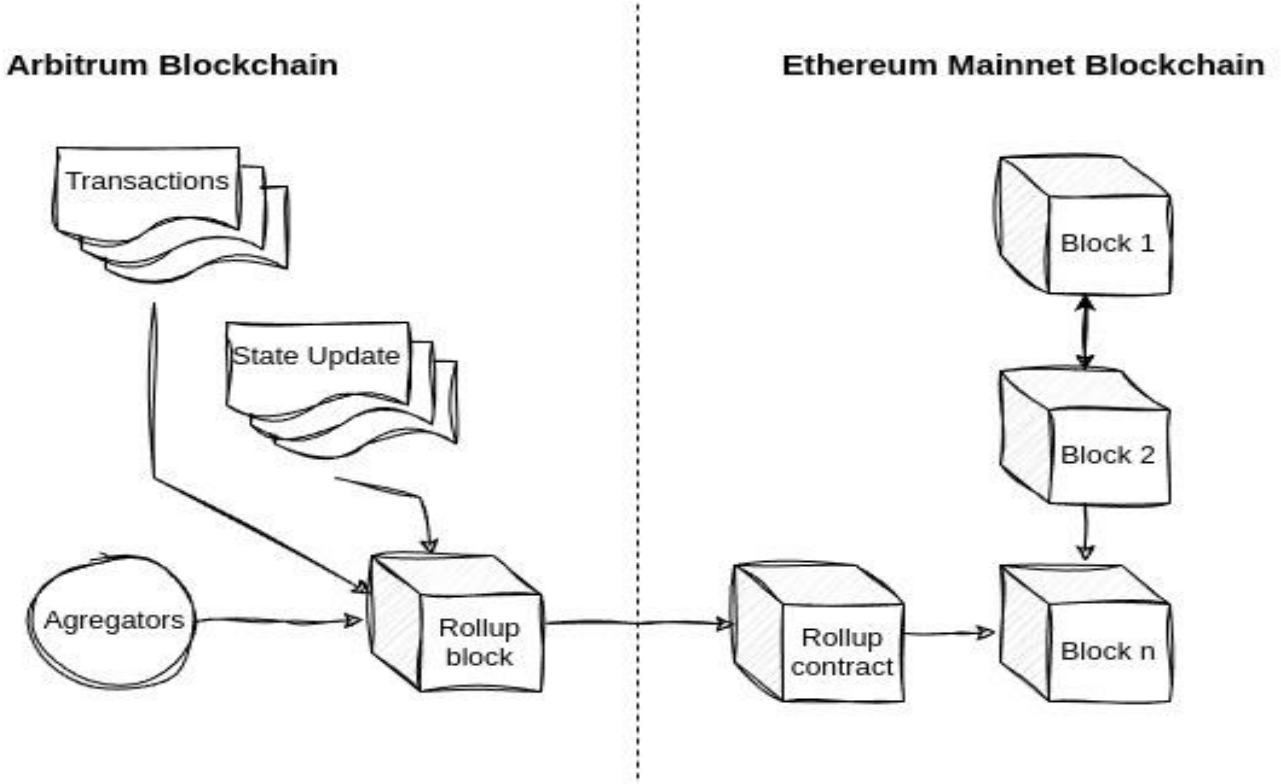

Рисунок 4.1.3 – Робота технології optimistic rollups

Тобто транзакція відбувається на рівні мережі Arbitrum що забезпечує швидкість та дешевизну використання мережі, а ось безпека та достовірність даних покладається на велику та перевірену мережу Ethereum. Основна мережа є найбільш надійною мережею з усіх що існує, оскільки має величезну кількість блоків які накопичувались роками, розподіленність даної мережі вражає. На сьогодні мережа Ethereum Mainnet має приблизно 6,655 вузлів зберігання інформації.

В такому випадку побудова інструменту на базі мережі Arbitrum яка затверджує свій стан в мережі Ethereum Mainnet забезпечить нам максимально можливий рівень захисту у співвідношенні до рентабельності.

4.2 Реалізація безпеки застосунку

Як було визначено віще, обраним є мікро-сервісний тип архітектури додатку, а отже кожен з сервісів має такий тип як RESTful API. При побудові додатку на базі RESTful API важливо дотримуватись основних принципів цього архітектурного стилю. До основних принципів належить:

– client-server – відокремлення серверної та клієнтської частин;

– Stateless-app — означає що кожен запит який клієнт надіслає на сервер, вже містить усі необхідні дані для його виконання та обробки. Сервер не має зберігати інформацію про попередні стани системи;

– cacheable — означає можливість зберігати відповідь на конкретний запит для подальшого повторного використання якщо стан не даних не змінився. Це робиться заради зменшення навантаження на інщі сервіси;

– Uniform Interface — для визначення способу взаємодії клієнта та сервера;

– багатошарова архітектура.

Спираючись на основні принципи котрі були перелічені у перших двох пунктах, архітектура не може передбачати збереження сесії на стороні серверу. Тож для підтримки сесії та авторизованості користувача треба розробити окремий сервіс який буде займатись аутентифікацією. В такому випадку сесію можна зберігати на стороні клієнта за допомогою JWT токена.

За потребою наявності такого сервісу було вирішено написати його самостійно. В такому випадку було використано високоефективну мову програмування Golang та бібліотеку для роботи з JWT токенами. Приклад токену який генерує окремий розроблений сервіс показано на рисунку 4.2.1.

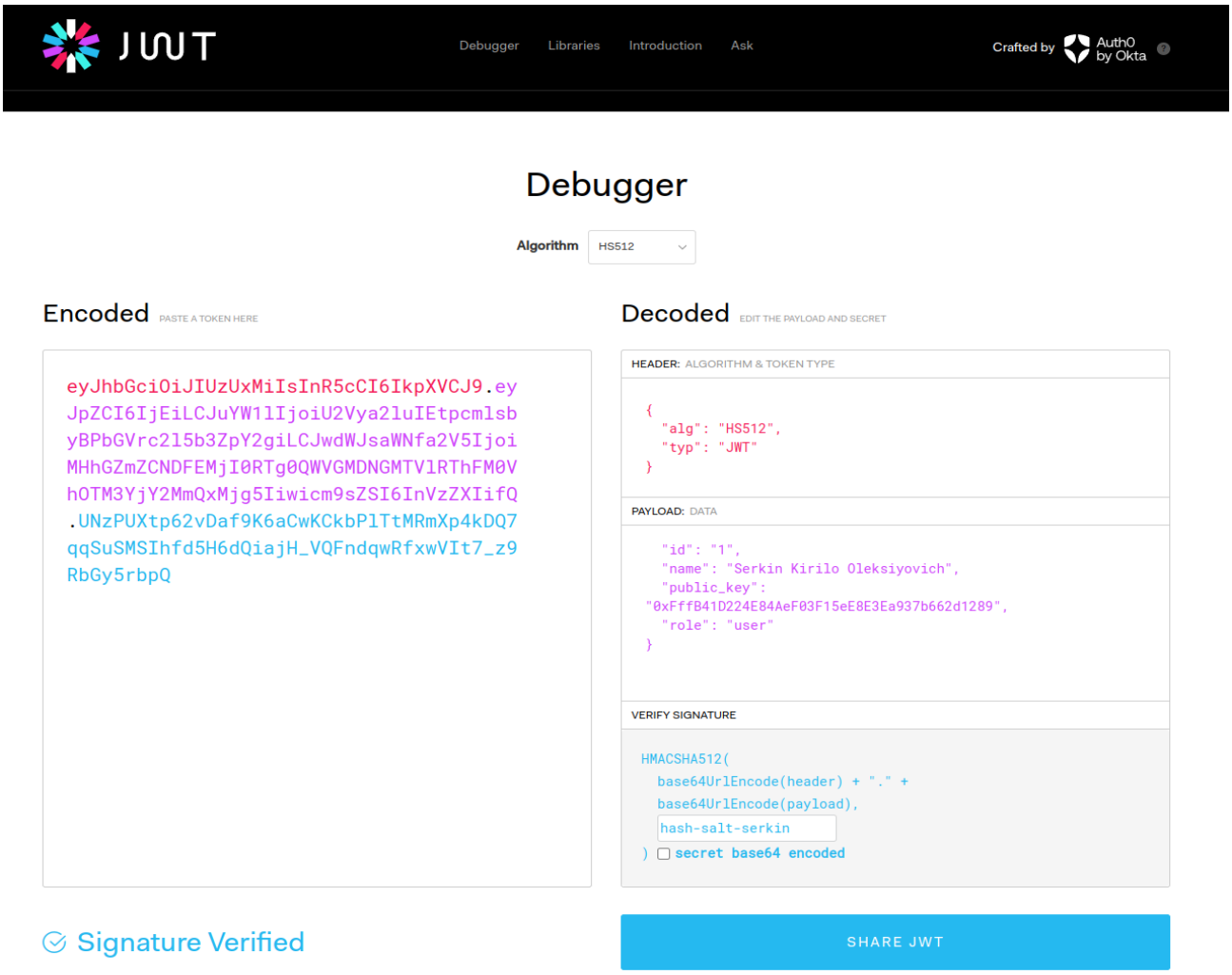

Рисунок 4.2.1 – Приклад сгенерованого токену сервісом авторизації.

Сервіс повинен був мати такі якості як:

– гнучкість;
– дінамічну конфігурованість;
– масштабованість;
– можливість кастомізації.

В такому випадку сервіс повинен відповідати усім стандартам зазначеним вище. Та орієентуватись на основні особливості платформи, так наприклад замість логіну користувачу потрібно буде підключити свій гаманець та ввести пароль для того щоб авторизуватись. Це означає що сервер буде отримувати замість логіна публічний ключ гаманця користувача. Прикалд підключення крипто-гаманця до веб-додатку зображено на рисунку 4.2.2.

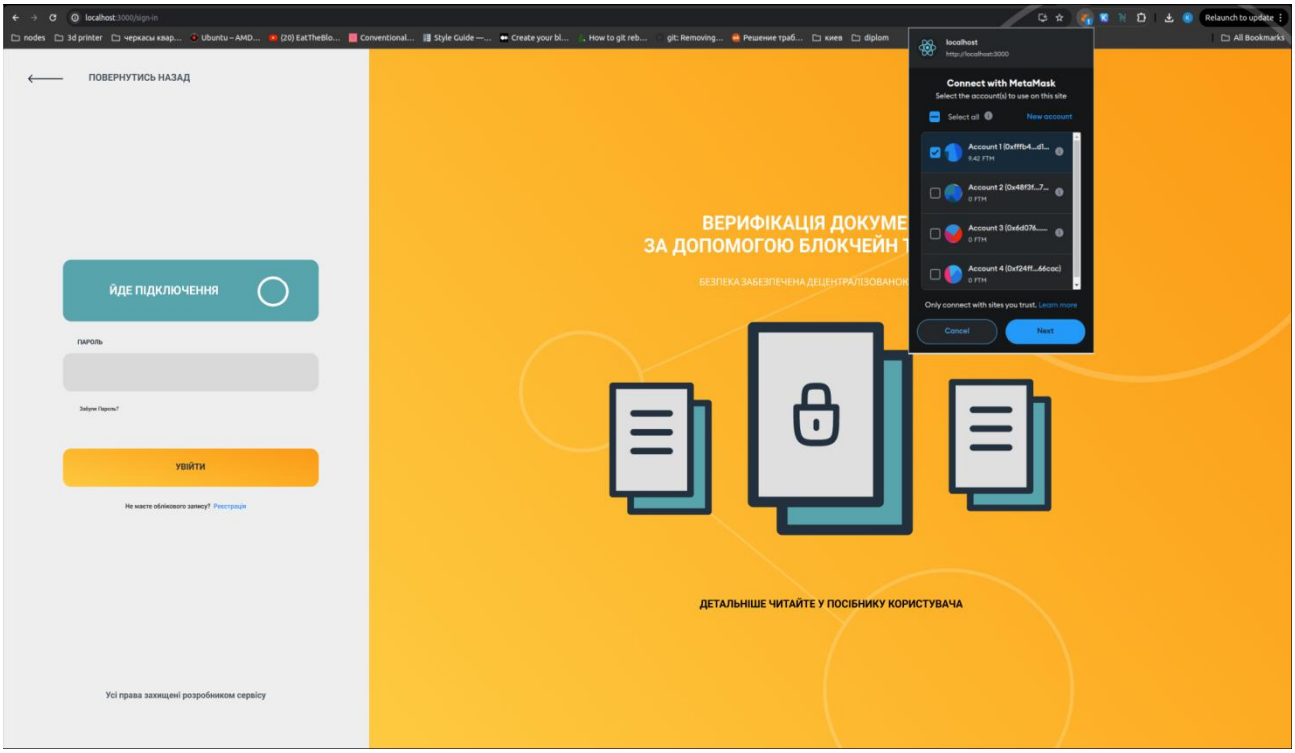

Рисунок 4.2.1 – Приклад сгенерованого токену сервісом авторизації.

Такий тип авторизації було обрано за причини що вкрасти данні авторизації гаманця набагато складніше ніж просто вкрасти логін та пароль. Фраза для отримання доступу до гаманцю, так звана мнемонічна фраза, складається з 12-24 слів, що є більш безпечним способом доказу володіння аккаунтом ніж логін та пароль. В такому випадку якщо важко отримати доступ до гаманця, то й отримати доступ нашої платформи буде набагто важче.

4.3 Реалізація основних механізмів підписання та верифікації документів

Найважливішим компонентом розроблюваного інструменту є механізм верифікації документів. Оскільки саме тут треба забезпечити максимальну захищеність більша частина усіх операцій має бути проведена в середені мережі блокчейн.

Для користувача в свою чергу цей процес має бути легким та не потребувати від нього багато дій. Все що потрібно користувачу це увійти та

підключити свій крипто-гаманець, підключитися до потрібної мережі, завантажити низку документів та підтвердити транзакцію в гаманці. В середені обраного блокчейну вже розгорнуто нашу програмну реалізацію механізму верифікації. Після додавання потрібних документів до списку належних до підписання документів, користувачу потрібно підтвердити транзакцію та почекати її виконання. Наглядно процес підписання транзакції після додавання потрібних документів можна розглянути на рисунку 4.3.1.

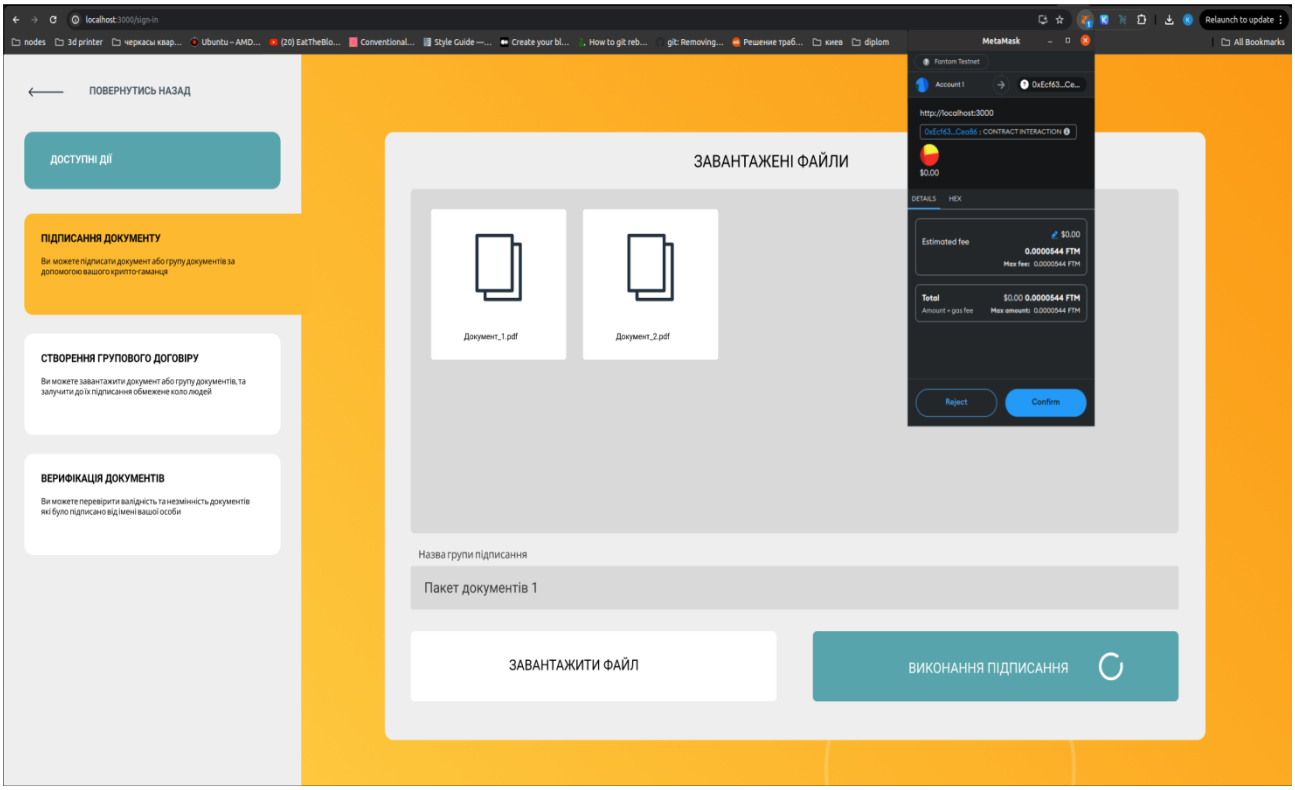

Рисунок 4.3.1 – Процес підписання транзакції після додавання потрібних документів.

Після успішного підписання документів користувач може перевірити валідність даних, тобто впевнитись що вони не були змінені. Для цього користувач має обрати який саме набір документів він хоче перевірити, а потім власноруч завантажити тіж файли що він додавав в процесі підписання. Якщо у користувача вже не має тих документів що були у підписаній групі, їх можна завантажити прямо у веб-додатку. Після того як користувач зробив вище описані кроки, потрібно лише натиснути кнопку “Верифікувати” та підписати

транзакцію, процес підписання транзакції для верифікації документів показано нижче на рисунку 4.3.2.

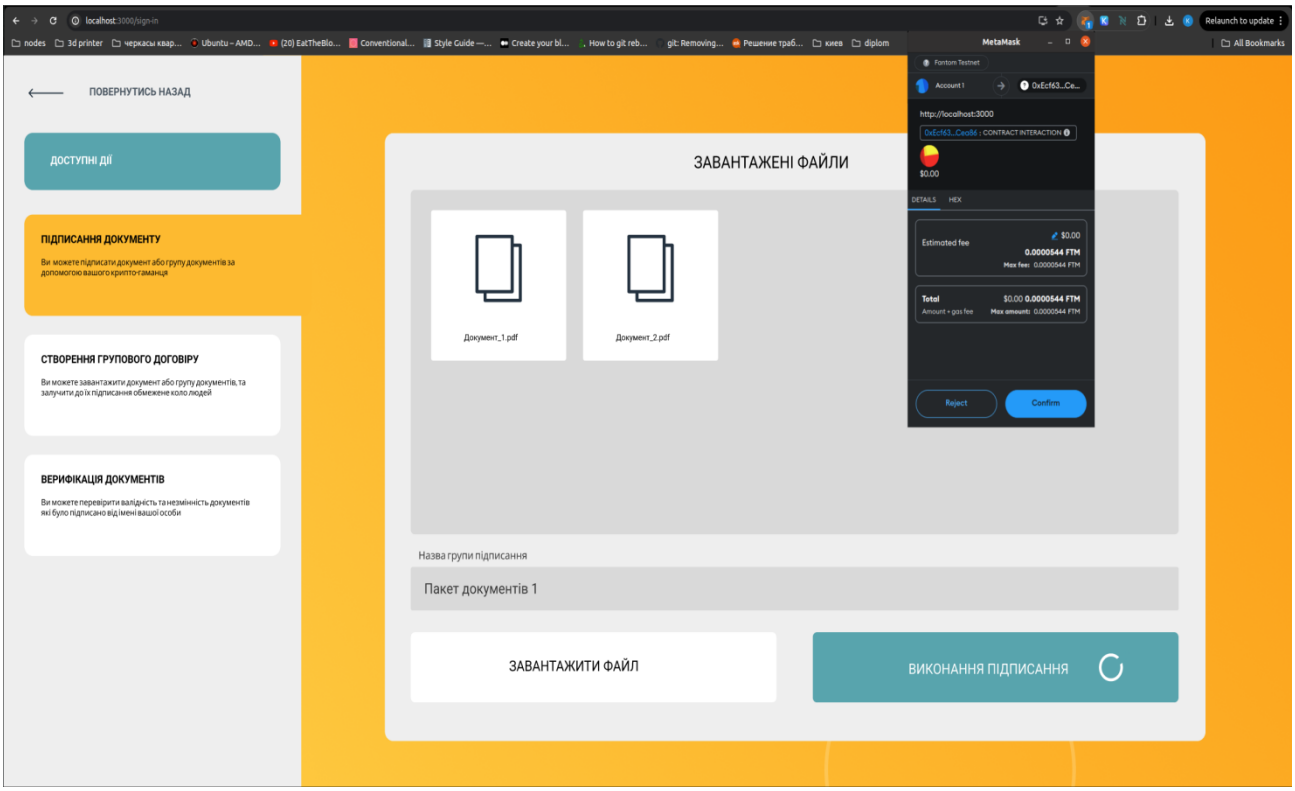

Рисунок 4.3.2 – Процес підписання транзакції для верифікації документів.

Одним з найскладніших реалізованих процесів є підписання документів групою осіб тобто групою контрагентів. Для цього треба перейти на зазначену вкладку, далі обрати та завантажити потрібні документи як у випадку з вкладкою підписання документів. Після цього натиснути далі і перейти до процесу пошуку контрагентів, їх можна знайти за ІПН оскільки це унікальний ідентифікатор платника подактів, він ідеально підходить для пошуку контрагентів. Звісно можна використовувати й інші дані для пошуку, але поки інструмент має на меті сам верифікацію та валідацію документів, функціонал працює за цим принципом.

Після того як було обрано потрібного контрагента або декількох контрагентів все що потрібно це підписати транзакцію на створення групового договору. Приклад підписання такої транзакції наведено нижче на рисунку 4.3.3.

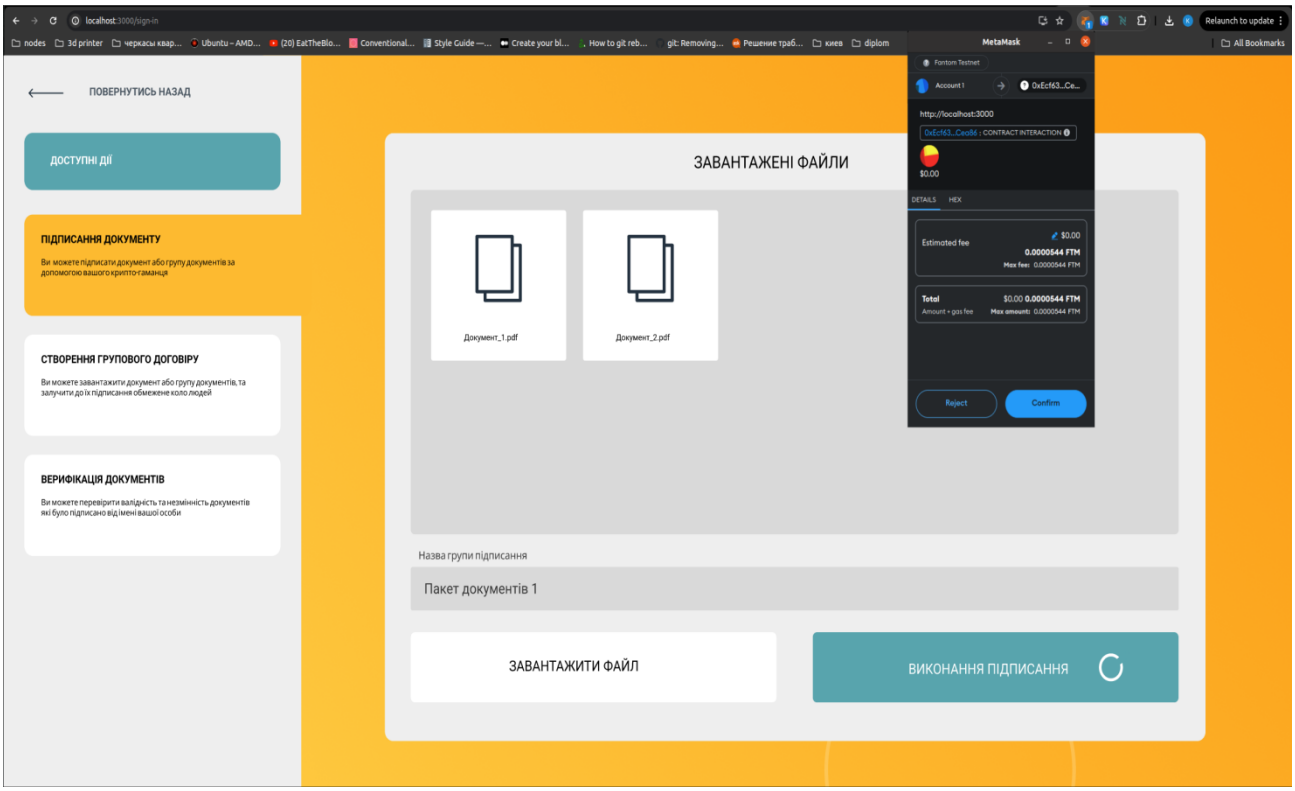

Рисунок 4.3.3 – Процес підписання транзакції для створення групового договору.

ВИСНОВКИ

В процесі створення кваліфікаційної роботи було проведено дослідження та аналіз безпеки верифікації й зберігання даних в централізованій та децентралізованій середі. В результаті проведеного дослідження було виявлено вразливості та недоліки обох з представлених методів зберігання інформації, а також порівняно їх схожі та розхожі вразливості. В результаті порівняння виявлених вразливостей було зроблено висновок, що децентралізовані системи мають менше точок нестабільності та вразливості.

На основі висновків попередніх досліджень було здійснено пошук та аналіз вже наявних систем верифікації документів та зберігання даних. В результаті цього була обрана низка платформ які займають перші місця серед інструментів для ведення документообігу, верифікації документів та захисту даних.

Були створені основні критерії порівняння критичних частин архітектури наявних платформ з потенційним рішенням на базі блокчейн. Критерії були розроблені за допомогою проведених раніше досліджень та попереднього аналізу захисту та зберігання даних. Результатом цього порівняння стало рішення розробити концепцію системи верифікації та збереження документів на базі блокчейн мережі. Було вирішено реалізувати такий додаток у вигляді веб-застосунку.

Сформовано основні вимоги до методів зберігання даних та описано їх первинний вигляд у ядрі системи. Також в процесі формування вимог було зазначено створення гнучкої система яку можна буде інтегрувати з іншими сервісами як допоміжний інструмент захисту

В процесі виконання кваліфікаційної роботи було створено концепт веб-додатку який реалізує основні механізми підписання та верифікації документів на базі блокчейн технологій.

Результати проведених досліджень були опубліковані на двох міжнародних конференціях [22][23].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. What is blockchain technology? [Електронний ресурс] - Режим доступу: https://aws.amazon.com/what-is/blockchain/?nc1=h_ls&aws-products-all.sort-by=item.additionalFields.productNameLowercase&aws-products-all.sort-order=asc

2. TechTarget [Електронний ресурс] - Режим доступу: https://www.techtarget.com/searchcio/definition/distributed-ledger#:~:text=Blockchain%20is%20a%20well%2Dknown,helps%20ensure%20trust%20and%20transparency.

3. Ethereum blockchain [Електронний ресурс] - Режим доступу: https://ethereum.org/en/what-is-ethereum/

4. JavaPoint blockchain merkle tree [Електронний ресурс] - Режим доступу: https://www.javatpoint.com/blockchain-merkle-tree

5. Chain Link Education [Електронний ресурс] - Режим доступу: https://chain.link/education/smart-contracts

6. Solidity Lang [Електронний ресурс] - Режим доступу: https://docs.soliditylang.org/en/v0.8.25/

7. Blaze Tech [Електронний ресурс] - Режим доступу: https://blaize.tech/article-type/web3-security/9-most-common-smart-contract-vulnerabilities-found-by-blaize/#:~:text=Curve%20LP%20Silos.-,2.%20REENTRANCY,-Reentrancy%20is%20one

8. "Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications" [Електронне видання] / Автор: Andreas M. Antonopoulos – 2018. – 416 с.

9. "Blockchain Basics: A Non-Technical Introduction in 25 Steps" [Електронне видання] / Автор: Daniel Drescher – 2017. – 210 с.

10. «Чиста архітектура. Мистецтво розробки програмного забезпечення» [Електронне видання] / Автор: Роберт Мартін – 2018. – 352 с.

11. "Blockchain Revolution: How the Technology Behind Bitcoin and Other Cryptocurrencies is Changing the World" [Електронне видання] / Автори: Don Tapscott, Alex Tapscott – 2016. – 368 с.

12. "Blockchain Applications: A Hands-On Approach" [Електронне видання] / Автор: Arshdeep Bahga, Vijay Madisetti – 2018. – 324 с.

13. "Learning React: Modern Patterns for Developing React Apps" [Електронне видання] / Автор: Alex Banks, Eve Porcello – 2020. – 334 с.

14. "React Design Patterns and Best Practices: Build easy to scale modular applications using the most powerful components and design patterns" [Електронне видання] / Автор: Michele Bertoli – 2017. – 434 с.

15. "Mastering Ethereum: Building Smart Contracts and DApps" [Електронне видання] / Автори: Andreas M. Antonopoulos, Gavin Wood – 2018. – 416 с.

16. "Solidity Programming Essentials: A beginner's guide to build smart contracts for Ethereum and blockchain" [Електронне видання] / Автор: Ritesh Modi – 2018. – 220 с.

17. "Building Ethereum Dapps: Decentralized Applications on the Ethereum Blockchain" [Електронне видання] / Автор: Roberto Infante – 2019. – 290 с.

18. "Clean Architecture: A Craftsman's Guide to Software Structure and Design" [Електронне видання] / Автор: Robert C. Martin – 2017. – 432 с.

19. "Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems" [Електронне видання] / Автор: Martin Kleppmann – 2017. – 616 с.

20. "Kubernetes: Up and Running, Dive into the Future of Infrastructure" [Електронне видання] / Автори: Kelsey Hightower, Brendan Burns, Joe Beda – 2017. – 202 с.

21. "The Kubernetes Book: Updated December 2020 for Kubernetes 1.20" [Електронне видання] / Автор: Nigel Poulton – 2020. – 271 с.

22. Серкін К.О, Науковий керівник Іванов В.Г. Дослідженная та аналіз можливостей технології блокчейн у сфері підписання та верифікації документів. Інформаційні технології в соціокультурній сфері, освітіта економіці: матеріали Міжнародної науково-практичної конференції студентів і молодих учених, м. Київ, / М-во освіти і науки України; Київ. нац. ун-т

культури і мистецтв. Київ: Вид. центр КНУКіМ, 2024.

23. Серкін К. О., Науковий керівник - Іванов В.Г.Дослідження та аналіз можливостей верифікації документів за допомогою блокчейн технологій. Матеріали 28-го Міжнародного молодіжного форуму «Радіоелектроніка та молодь у XXI столітті» . Т. 6, –Харків: ХНУРЕ. 2024.-с. 704-705. DOI: https://doi.org/10.30837/IYF.IIS.2024.704