Доброго дня! Шановні члени комісії, Вашій увазі представляється бакалаврська робота за темою:

Слайд 1(Вступ) - Протягом сучасної історії людства завжди гостро стояло питання коректності даних, їх валідності та можливості їх фальсифікації. В епоху комп'ютерних технологій, почали з'являться безліч способів довести оригінальність даних, але більшість способів мають одну спільну проблему – вони можуть бути змінені зсередини системи, а отже в теорії дані можна з фальсифікувати, якщо мати прямий доступ до бази даних.

Дана робота призвана дослідити дане питання та надати можливі рішення.

Слайд 2 – Об'єкт, предмет та ціль роботи

Об'єктом дослідження є сучасні методи збереження інформації.

Предметом дослідження є системи токенізації на основі технології блокчейн.

Метою роботи є аналіз проблеми токенізації даних та документів.

Слайд 3 – Задачі дипломної роботи

- Провести аналіз предметної області
- Проаналізувати проблеми
- Порівняти існуючі програмні рішення та виявити недоліки
- Проаналізувати методи рішення поставлених проблем
- Сформувати вимоги до програмного забезпечення
- Виконати програмну реалізацію
- Виконати тестування системи

Слайд 4 – Виявлення проблем та актуальності теми:

Згідно зі статистичними даними Генеральної прокуратури України з кожним роком випадки підроблення документів залишаються незмінно

високими. Це негативно впливає на економіку, витрачає багато ресурсів спецслужб, та сприяє збільшенню кількості шахраїв та некваліфікованих людей в державі.

Слайд 5 – На даному слайді можна побачити кількість судових справ Вінницького експертно-криміналістичного центру, що розглядали підробку документів землеволодіння.

На основі цих даних можна зробити висновок, що проблема фальсифікації з кожним роком залишається або не змінно поганою, аьо навіть погіршується.

Слайд 6 – Необхідним кроком у вирішенні проблеми, є огляд уже існуючих систем токенізації, а саме Ethereum та Bitcoin.

Основними критеріями для системи токенізації було виявлено наступні три характеристики: надійність, функціональність та швидкість обробки та внесення даних.

Ethereum є надійний, проте існує можливість похибки зі сторони розробника смарт-контрактів.

Bitcoin – є також надійним.

Ethereum - має майже нескінчену кількість способів використання завдяки смарт контрактам.

Bitcoin – є обмеженим, та надає функціонал лиш в керуванні активами.

Ethereum є швидшим за Bitcoin, але має простіші алгоритми хешування.

Отже порівнявши сучасні аналоги можна зробити висновок, що недоліки є у всіх представлених рішень, а отже доцільність побудови нової системи є виправдованою.

Слайд 7 – проаналізувавши сучасні проблеми та мінуси сучасних аналогів, можна створити методи рішення поставлених проблем, а саме:

1) Використання блокчейну для збереження даних

2) Використання електронних -підписів
3) Використання надійних алгоритмів хешування
4) Розробка простого у використані програмного забезпечення

Слайд 8 – Також можна вирішити основні методики, алгоритми та технології, що слід використовувати для вирішення проблеми:

Технологія блокчейн представляє собою децентралізоване сховище даних, котрі мають зв'язки між собою, що не дає можливість зміни уже доданої та підтвердженої інформації.

На слайді представлено схематичне зображення роботи системи токенізації.

Слайд 9 - Технологія Peer-To-Peer

P2P є важливою технологією, що добре доповнює блокчейн, тому що надає децентралізацію, що дає додатковий шар захисту від підробок. Бо підробивши весь ланцюжок даних у себе на пристрої, фальсифіковані дані все одно будуть відхилені іншими користувачами мережі, котрі мають також коректний ланцюг даних, але уже синхронізований між більшістю користувачами мережі та довшим за фальшивий варіант.

На слайді зображено різниця в звичній клієнт серверній архітектурі та P2P.

Слайд 10 – На даному слайді представлені функціональні вимоги до майбутнього програмного продукту:

1) Система має надавати користувачу змогу зареєструватися чи увійти.

2) Система має зберігати інформацію користувачів.

3) Система має надати можливість користувачу додавати нову інформацію до блокчейну.

4) Система має надавати користувачу повний доступ до інформації, що зберігається в блокчейні.

5) Система має захищати інформацію в блокчейні від змін.

6) Система має локально зберігати варіант блокчейну на пристрої користувача та синхронізуватися з мережею.

Слайд 11 – Слід також визначити основні нефункціональні вимоги:

Продуктивність - формування нового блоку до 30 секунд.

Доступність у використанні - інтерфейн простий та зрозумілий, для рядового користувача освоїтись повинно займати до 1 робочого дня.

Безпека – можливість підробити дані має бути вкрай низькою, для цього зловмисникам потрібно мати більше 50% апаратної потужності мережі. Приватний ключ має бути унікальним для кожного акаунта.

Локалізація – система має бути локалізована на англійську мову.

Технічні вимоги – система має працювати на операційній системі Windows 10.

Слайд 12 - Технологій для реалізації:

Розроблюване ПЗ має бути розроблено на операційній системі Windows 10, тому що дана ОС є найпоширенішою для персональних комп’ютерів. А відносно портативного сегмента, то вході аналізу програмних обмежень даних систем, було вирішено відмовитись від них.

Оскільки операційною системою є Windows, то найбільш зручним та оптимізованим рішенням є використання .NET Framework та мови програмування c#.

Відповідно до цього рішення, для UI складової ПЗ, є найбільш доцільним використання WPF, бо також є прямою розробкою Microsoft для рішень на ОС Windows.

Теж саме можна сказати про Entity framework.

Для реалізації P2P буде використовувати розширення класичної бібліотеки .NET Framework System.Net.PeerToPeer.

А для алгоритму хешування буде використовуватися SHA3-KECCAK, як найновіший, найнадійніший, сертифікований алгоритм.

Слайд 13 – Визначивши основний вектор розроблюваного ПЗ можна перейти до розробки архітектурних рішень:

Архітектура програми має містити в собі наступні модулі:

- Peer-To-Peer API
- Local blockchain store API
- Blockchain Service
- UI interface

Схематичне відображення цих модулів та їх взаємодія показано на наступному слайді у вигляді діаграми розгортання

Слайд 14 – На даному слайді, відповідно, можна побачити діаграму розгортання

Слайд 15 – Також, слід сформувати основні можливі маніпуляції користувача з системою. Це схематично показано на діаграмі варіантів використання.

Слайд 16-20 – На наступних слайдах показано, за допомогою діаграм діяльностей, як мають проходити ті чи інші процеси в середині системи токенізації.

Слайд 21 – Локальне сховище

Сховище генерується автоматично за допомогою Entity Framework.

Доступ до сховища здійснюється за допомогою цього ж фрейм ворку, та спеціально створеного в ході розробки API.

Сховище містить в собі список усіх блоків.

Вміст блоків зображено безпосередньо на слайді.

Слайд 22-24 – На основі розроблених архітектурних рішень було розроблено систему токенізації, що відповідає поставленим вимогам та виконує поставлені задачі. Для демонстрації роботи системи було розроблено додаток, що має на меті візуальне відображення стану блокчейн мережі. Вигляд даного додатка показано на наступних слайдах.

((Далі опис кожного елемента))

Слайд 25 – після виконання розробки системи бло проведено ряд тестів.

На даномі слайді зображено результат виконання модульних тестів окреми частин системи. А саме: Тестування P2P API, Local store API та Blockchain API.

Також було проведено функціональне та не функціональне тестування.

Згідно до раніше представлених функціональних вимог система має надавати змогу зареєструватися або увійти, додати новий токен до блокчейну, та переглянути усі наявні блоки у блокчейні. Усі поставлені вимоги система задовольняє.

Продуктивність – генерація блоку, та додавання його до глобального блокчейну відбувається менш ніж за 30 секунд.

Доступність у використанні –інтерфейс є зрозумілим та простим.

Надійність – система стабільно працює при середньому рівні завантаженості на неї. Вході тестування на стресостійкість система гарно себе показала. При завантажені 40 блоків з яких близько половини були файли середнім розміром у 500 кб, час очікування повністю новоствореного аканта на завантаження склало менш ніж 1 секунда на одному пристрої.

Безпека – можливість підробити дані існує лиш в випадку переваги більш ніж 50 відсотків у апаратних можливостях конкретного користувача, що є звичним для схожих систем токенізації.

Локалізація – система повністю локалізована на англійській мову.

Технічні вимоги – система працює на операційній системі Windows 10 та потребує менш ніж 8 гб оперативної пам’яті.

Інтерфейс – повністю відповідає стилістичним вимог, що були поставлені раніше.

Слайд 26 – Приклади використання

Розроблену систему можна використовувати наприклад в системах, що мають на меті медичний облік історій хвороб пацієнтів. Або звичайне збереження документів, наприклад про закінчення вищої освіти чи проходження якогось курсу. Також, можна зберігати невеликі програми, що можна використовувати як спрощену альтернативу смарт-контрактів Ethereum.

Усе це можна досягти не змінюючи саму систему, а лиш змінюючи найвищий шар взаємодії з системою та інтерфейс користувача.

Слайд (27 – 28) – У рамках тестування можливостей подальших модифікацій, було розроблено ПЗ для обліку історій хворих медичних закладів.

Час на розробку даного ПЗ склав 2 робочих дні, що є досить швидким результатом.

Вигляд даної програми можна побачити на наступних слайдах.

Слайд 29 – Можливості вдосконалення системи токенізації

При розробці ПЗ були виявлені недоліки в архітектурі взаємодій між системою токенізації та системою Peer-To-Peer передачі даних. Ці проблеми слід виправити для більшої надійності, та покращенню гнучкості системи для подальших модифікацій.

Також слід провести оптимізацію збережень даних при першому запуску застосунка, тому що, час входу буде значно збільшуватися з кількістю інформації, що знаходиться в блокчейні.

Слайд 30 – Висновки

В ході дипломної роботи було:

- Проведено аналіз предметної області;
- Проаналізовано проблеми;
- Виявлені недоліки та переваги сучасних систем;
- Проведений аналіз методів рішення поставлених проблем;
- Сформовані вимоги до програмної системи
- Виконано реалізацію системи
- Виконано тестування розробленої системи

Слайд 31 – Дякую за увагу