



## Мета роботи

- ❖ Метою роботи є дослідження сучасних архітектур центрів сертифікації ключів, їх порівняльний аналіз з подальшою розробкою сховища ключів з підтримкою заміни та перевірки цілісності.
- ❖ В результаті розробки отримана програма «CertOnTheGo», яка може використовуватися користувачами зберігання та оновлення сертифікатів з можливістю контролю їх цілісності.

Рисунок А.3 – Мета роботи

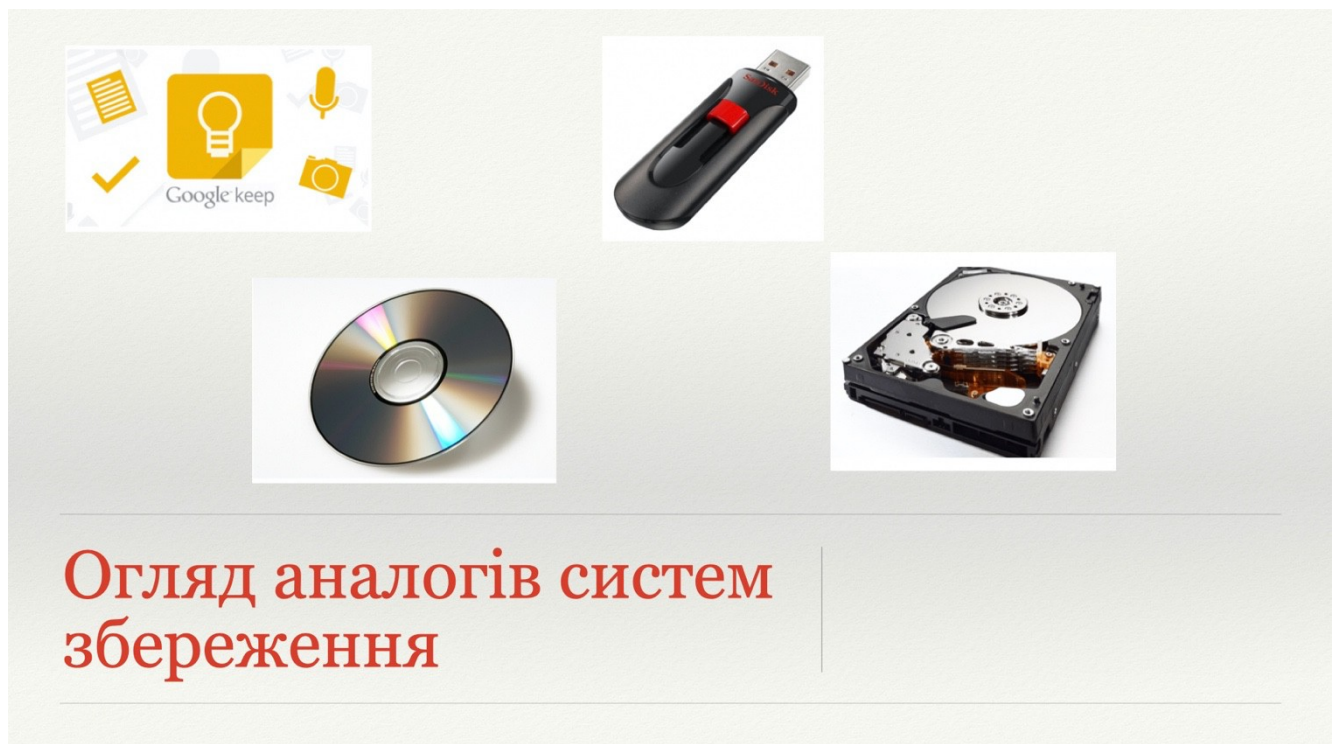
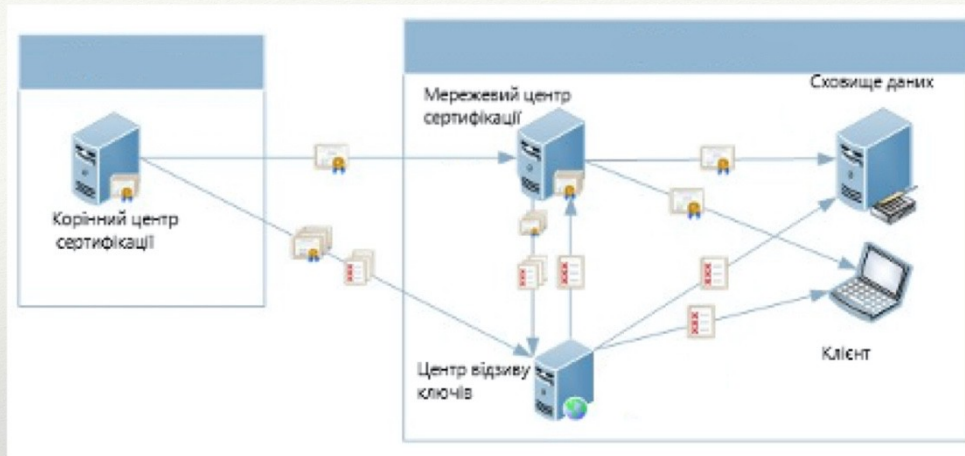


Рисунок А.4 – Огляд аналогів систем збереження



## Огляд аналогічних систем центрів сертифікації

На прикладі системи Active Directory від Microsoft видно, як даний вид систем може розгортатися в корпоративній мережі.

Рисунок А.5 – Огляд аналогічних систем центрів сертифікації

## Переваги аналогічних середовищ зберігання

- ❖ Доступ з будь-якої точки світу (Cloud services)
- ❖ Надійність, можливість резервного копіювання
- ❖ Незмінність інформації (Optical disk)

Рисунок А.6 – Переваги аналогічних середовищ зберігання

## Недоліки аналогічних середовищ зберігання

- ❖ Можливість перекодування із втратами (Cloud services)
- ❖ Можливість виходу з ладу сховища
- ❖ Неможливість відправити напряму клієнту або вбудувати напряму в цільові мережі

Рисунок А.7 – Недоліки аналогічних систем зберігання

## Функціональні вимоги

- ❖ Реєстрація нового користувача (один набір даних для всіх центрів системи)
- ❖ Одночасна підтримка прямого доступу за протоколом REST та інтерфейсу користувача
- ❖ Функціонал перегляду сертифікатів:
  - ❖ Перегляд сертифікатів користувача
  - ❖ Деталі конкретного сертифікату (включаючи публічний ключ)
  - ❖ Завантаження сертифікату у форматі .cer
- ❖ Функціонал керування сертифікатами:
  - ❖ Додавання сертифікату
  - ❖ Видалення сертифікату
  - ❖ Заміна сертифікату без втрати посилання

Рисунок А.8 – Функціональні вимоги

## Нефункціональні вимоги

- ❖ Система має підтримувати масштабування в залежності від навантаження
- ❖ Система має підтримувати резервну копію бази даних
- ❖ Система повинна перевіряти цілісність сертифікатів у базі даних щоденно
- ❖ Система має використовувати стійкий до атак механізм шифрування
- ❖ Обмеження:
  - ❖ Система розгортається лише у середовищі, що підтримує .Net Core 3.1 або Docker

Рисунок А.9 – Нефункціональні вимоги

## Варіанти використання

Система передбачає чотири варіанти користувача

- ❖ Зовнішній користувач. Такий користувач не має бути зареєстрованим та має право лише на отримання одного сертифікату по прямому посиланню.
- ❖ Тримач сертифікату. Такий користувач має зареєструватися в системі, після чого йому буде доступний функціонал завантаження сертифікату, видалення та заміни.
- ❖ Довірений користувач. Такий користувач має ті самі права, що й тримач сертифікату, але додатково може керувати корінними сертифікатами.
- ❖ Проміжний центр сертифікації. Для корінних центрів проміжні є користувачами з правами зовнішніх користувачів. Вони можуть тільки завантажити сертифікат із системи.

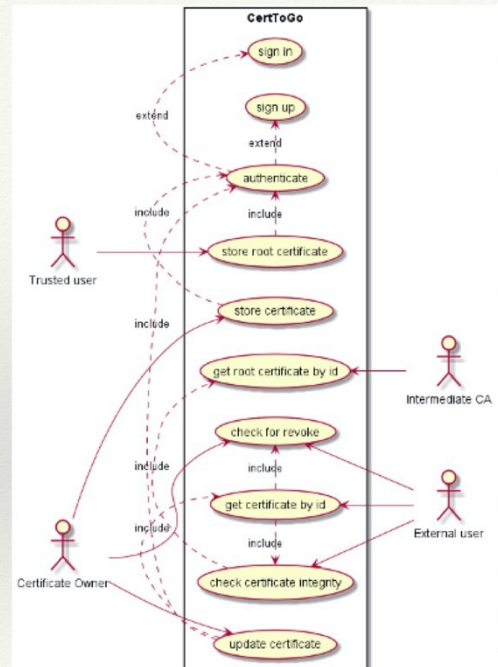


Рисунок А.10 – Варіанти використання системи

## Діаграма компонентів

У якості клієнта системи тут представлений компонент <<Web site>>, але це може бути абсолютно будь-який клієнт, що буде виконувати REST-запити

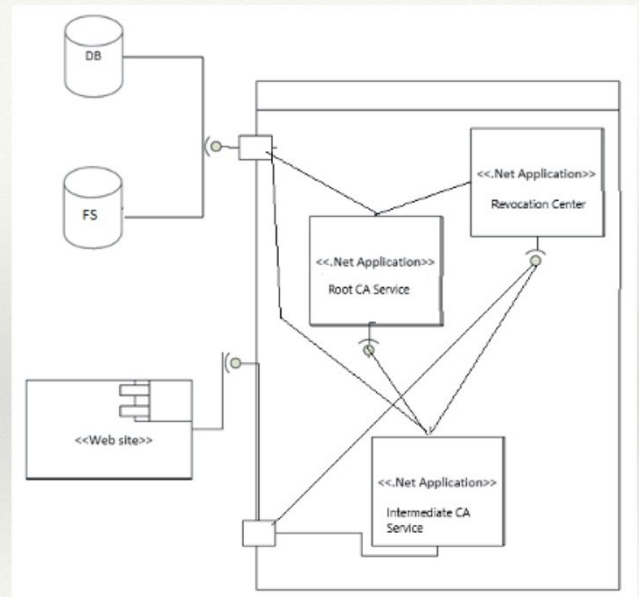


Рисунок А.11 – Діаграма компонентів

RootCa.WebApi Home Upload file Certificates

---

956239674 for Vladyslav Biletskyi, Issuer=Test issuer, 08/05/2018 - 06/05/2024  
 Replace Remove

---

67045623123 for Vladyslav Biletskyi, Issuer=Test issuer, 10/05/2017 - 08/05/2022  
 Replace Remove

## Форма перегляду сертифікатів користувача

Кожен рядок представляє посилання для перегляду повної інформації та скачування у форматі .cer

Рисунок А.12 – Форма перегляду сертифікатів користувача

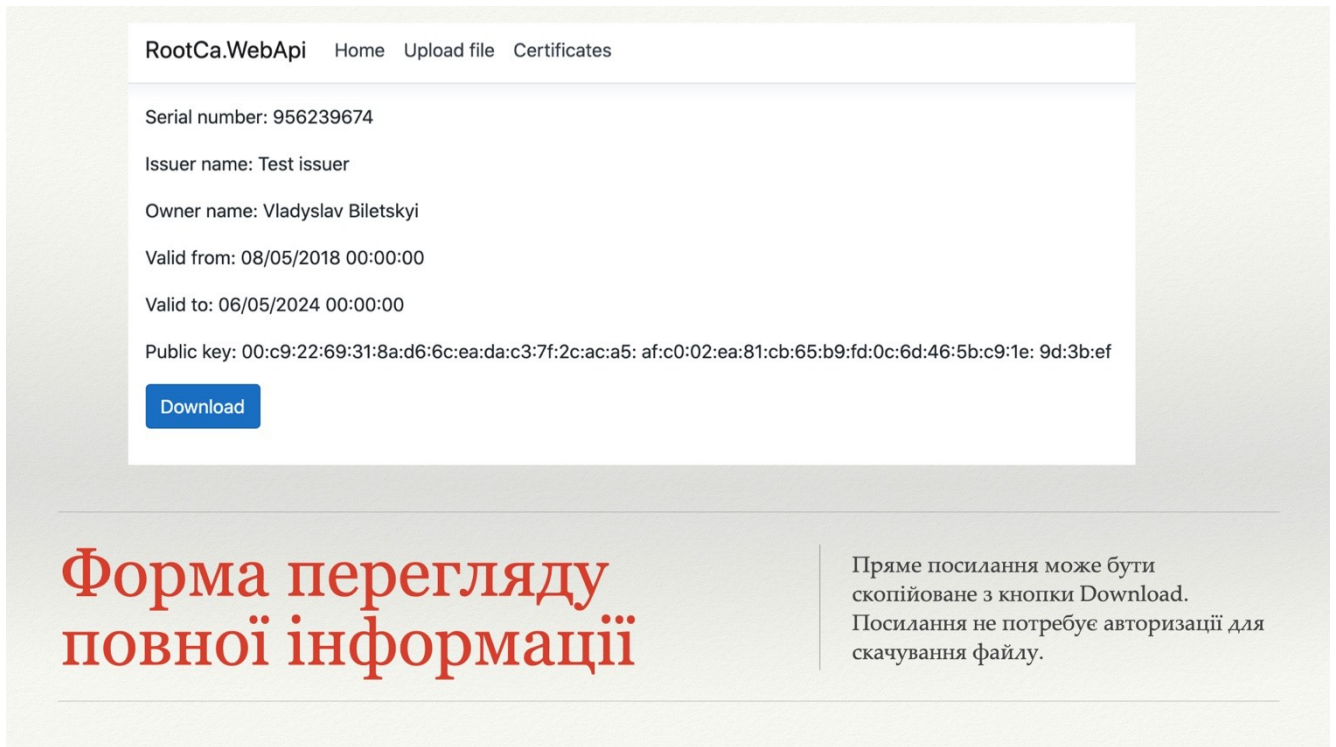


Рисунок А.13 – Форма перегляду повної інформації

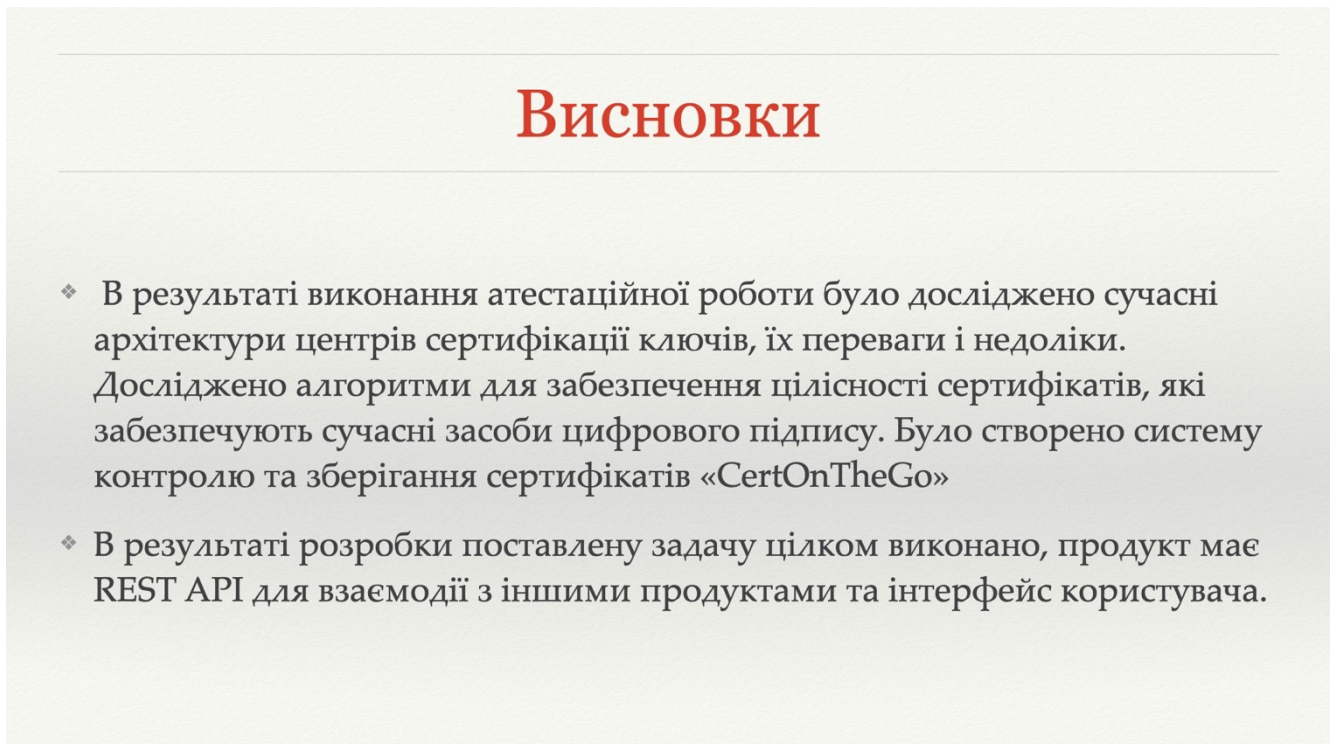


Рисунок А.14 – Висновки

**ДОДАТОК Б**  
Електронні матеріали