

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»



**Факультет інформаційних технологій  
Кафедра системного аналізу та управління**

**Звіт  
з практичних робіти з дисципліни  
«Аналіз програмного забезпечення»**

Виконав: ст. гр. 122–22- 1  
Косов Н.О.

Прийняла: асистент кафедри САУ,  
Шевченко Ю.О.

Дніпро  
2025  
Лабораторна №1

**Тема:** Підписання персонального документа за допомогою кваліфікованого електронного підпису (КЕП).

**Мета:** Набування навичок підписання особистої документації з використанням кваліфікованого електронного підпису.

### Хід роботи

1. Створення текстового файлу з інформацією про себе.

Я студент групи 122-22-1 Косов Нікіта, я люблю подорожувати і граю на гітарі.

### Рис. 1 Створений файл

2. Підпис файлу за допомогою Дія.підпис та його перевірка.

# Перевірити підпис



Файл успішно перевірено

Ви можете зберегти підпис

↓ Завантажити все архівом

## Файл з підписом

Косов122-22-1 (1).pdf

176.3 КБ

## Файл без підпису

Косов122-22-1 (1).verified.pdf

176.3 КБ

## Протокол створення та перевірки кваліфікованого підписання

Косов122-22-1 (1).verified\_Validation\_Report.pdf

176.3 КБ

Рис. 2 Перевірка підпису

Висновки: у ході виконання цієї практичної роботи я здобув навички підписання особистої документації за допомогою кваліфікованого

електронного підпису. Було створено персональний PDF-документ, який успішно підписано за допомогою сервісу «Дія».

## Лабораторна №2

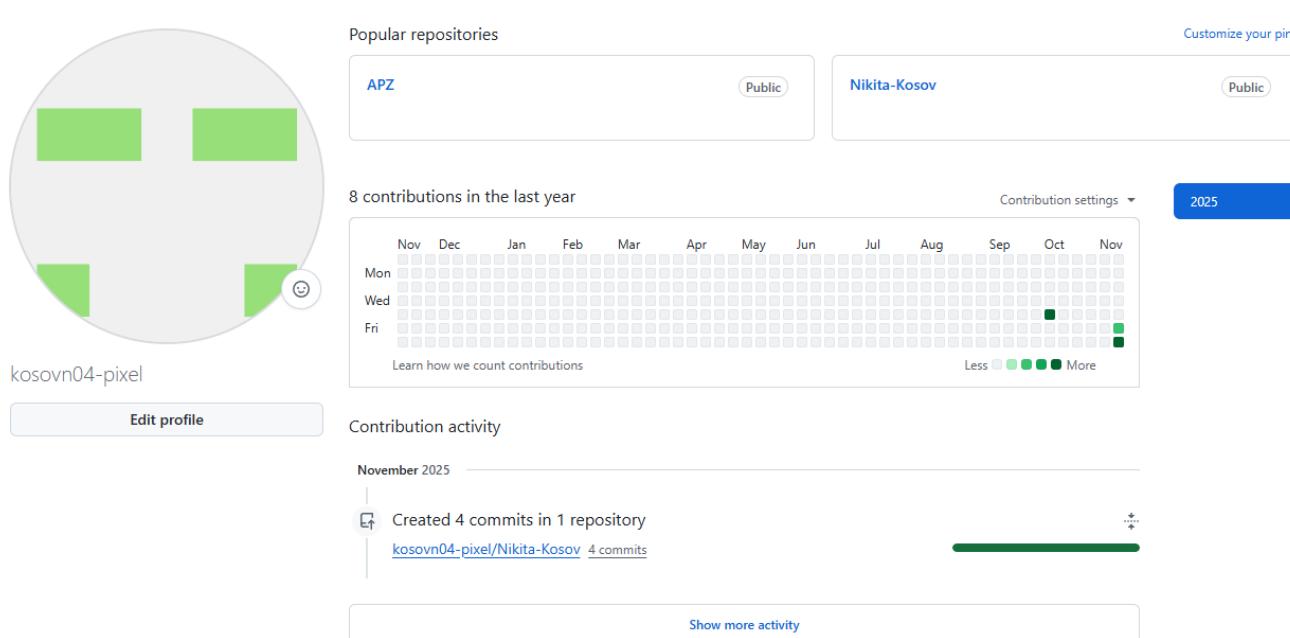
**Тема:** Створення і налаштування профілю у системі Git.

**Мета:** Набування навичок при реєстрації та налаштуванню облікового запису (account) на хостінгу GitHub.

**Завдання.** Створити власний репозиторій в GitHub. В подальшому усі результати своїх практичних робіт необхідно завантажувати у цей репозиторій. В репозиторії створити для кожної практичної роботи окрему папку і розмістити звіт. Якщо ви ще не маєте обліковий запис, будь ласка, в професійному світі, запис повинен виглядати, як поєднання першої літери ім'я та прізвище в повному виді. Наприклад Микола Єфремов буде «myefremov», а Тарас Шевченко «tshevchenko»

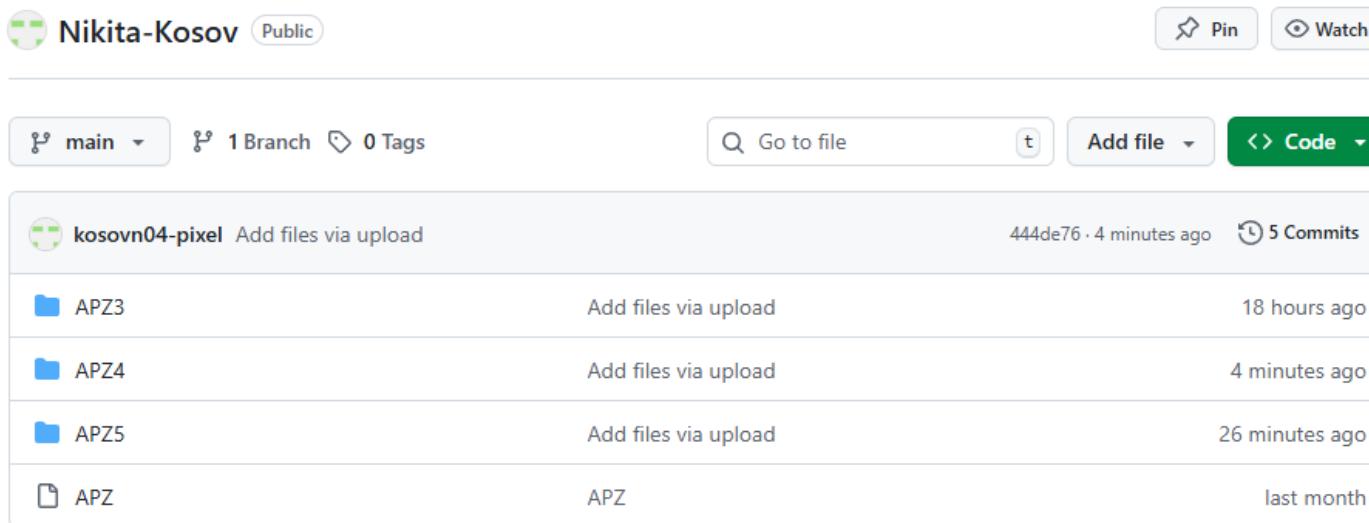
### Хід роботи

1. Створення акаунту github.



**Рис. 3 Створений акаунт**

## 2. Створення репозиторію.



**Рис. 4 Створений репозиторій**

Посилання на github репозиторій з лабораторними роботами:  
<https://github.com/kosovn04-pixel/Nikita-Kosov>

**Висновок:** у ході виконання практичної роботи я зареєстрував та налаштував обліковий запис на хостингу GitHub. Також було створено публічний репозиторій.

## **Лабораторна №3**

**Тема:** Написання тест-кейсів (Test Case).

**Мета:** Набування навичок у написанні тест-кейсів різних пристройів.

**Очікувані результати навчання:** уміння підписувати особисту документацію з використанням єдиного цифрового підпису за допомогою різних сервісів і додатків.

### **Короткі теоретичні відомості**

**Тест-кейс** – це професійна документація тестувальника, це послідовність дій, спрямована на перевірку будь-якого функціоналу, що описує як прийти до очікуваного результату.

**Тема кейса** – описова назва тесту, яка спрощує його пошук та розуміння його змісту. У темі тест-кейсі не повинно бути залежностей від інших тест-кейсів і також нечітких формулювань та зайвої деталізації.

### **Завдання.**

Придумайте об'єкт тестування. Це повинен буде будь-який об'єкт, який

складається мінімум з 5 частин. (годинник, скейт, велосипед, монітор і т.п.)

Напишіть не менше 20 тест кейсів до цього об'єкту, що зможуть оцінити його

якість з точки зору продукту. Створіть \*.pdf документ, де кратко опишіть ваш

об'єкт та потім опишіть ваші тест-кейси.

## Виконання

Об'єкт тестування: Розумний годинник "Aura 5"

Короткий опис: "Aura 5" — це розумний годинник (Smartwatch) останнього покоління. Він призначений для відстеження фізичної активності, моніторингу показників здоров'я, отримання сповіщень зі смартфона та виконання базових комунікаційних функцій.

Основні компоненти (частини):

1. Сенсорний AMOLED-дисплей: Головний елемент взаємодії.
2. Цифрове коліщатко (Digital Crown) та бічна кнопка: Елементи фізичного керування.
3. Оптичний датчик серцевого ритму (Пульсометр): Знаходиться на тильній стороні.
4. Модуль Bluetooth 5.2 та GPS: Для підключення до смартфона та відстеження місцезнаходження.
5. Акумулятор та модуль бездротової зарядки: Забезпечують автономну роботу.
6. Динамік та мікрофон: Для дзвінків та голосових команд.
7. Силіконовий ремінець: Для кріplення на руці.

Тест-кейси для Розумного годинника "Aura 5"

Нижче наведено 20 тест-кейсів, спрямованіх на оцінку якості продукту з точки зору користувача.

1. Функціональне тестування (Базові функції)

Test Case 1: Перше ввімкнення пристрою

- Назва: Перевірка першого запуску годинника (Out-of-Box Experience).
- Pre-condition: Годинник повністю заряджений, вийнятий з коробки, ще не вмикався.
- Кроки:

1. Натиснути й утримувати бічну кнопку протягом 3 секунд.
- Expected Result: Дисплей вмикається, відображається логотип виробника, запускається помічник початкового налаштування (вибір мови, підключення).
- Post-condition: Годинник увімкнено та знаходиться на екрані вітання.

#### Test Case 2: Відображення поточного часу

- Назва: Перевірка основної функції – відображення часу.
- Pre-condition: Годинник увімкнено, налаштовано та синхронізовано зі смартфоном.
- Кроки:
  1. Активувати екран (підняттям зап'ястя або дотиком).
  2. Перевірити час, що відображається на циферблаті.
- Expected Result: Час на годиннику точно відповідає поточному часу в часовому поясі (звірити зі смартфоном або світовим часом).
- Post-condition: Екран годинника гасне через встановлений тайм-аут.

#### Test Case 3: Зміна циферблата

- Назва: Перевірка кастомізації (zmіни) циферблата.
- Pre-condition: Годинник увімкнено, знаходиться на головному екрані (циферблаті).
- Кроки:
  1. Натиснути й утримувати палець на поточному циферблаті.
  2. З'явиться меню вибору. Свайпнути вліво або вправо, щоб обрати інший циферблат.
  3. Торкнутися обраного нового циферблата.
- Expected Result: Головний екран змінює свій вигляд відповідно до обраного циферблата.
- Post-condition: Новий циферблат встановлено як основний.

#### Test Case 4: Вимкнення живлення

- Назва: Перевірка коректного вимкнення пристрою.

- Pre-condition: Годинник увімкнено та працює.
- Кроки:
  1. Натиснути й утримувати бічну кнопку (5 секунд).
  2. У меню, що з'явилося, натиснути на іконку "Вимкнути".
  3. Підтвердити дію (якщо потрібно).
- Expected Result: На екрані з'являється анімація вимкнення, після чого дисплей повністю гасне. Годинник не реагує на дотики.
- Post-condition: Пристрій повністю вимкнено.

## 2. Тестування підключення (Connectivity)

### Test Case 5: Сполучення (Pairing) зі смартфоном

- Назва: Перевірка підключення годинника до смартфона через Bluetooth.
- Pre-condition: Годинник увімкнено (на екрані налаштування), смартфон (iOS/Android) має увімкнений Bluetooth та встановлений додаток "Aura Connect".
- Кроки:
  1. Запустити додаток "Aura Connect" на смартфоні.
  2. Обрати "Додати новий пристрій".
  3. Обрати "Aura 5" зі списку знайдених пристрій.
  4. Ввести код підтвердження (який з'явився на годиннику) у додаток на смартфоні.
- Expected Result: Годинник і смартфон успішно з'єднуються. На годиннику з'являється головний циферблат. У додатку відображається статус "Підключено".
- Post-condition: Пристрої синхронізовано.

### Test Case 6: Отримання сповіщення (дзвінок)

- Назва: Перевірка відображення вхідного дзвінка на годиннику.
- Pre-condition: Годинник підключено до смартфона, звук на смартфоні увімкнено.
- Кроки:

1. Здійснити вхідний дзвінок на смартфон з іншого номера.
- Expected Result: Годинник починає вібрувати та/або видавати звуковий сигнал. На дисплеї відображається ім'я/номер абонента та кнопки "Прийняти" / "Відхилити".
- Post-condition: Дзвінок завершено (прийнято або відхилено).

#### Test Case 7: Отримання сповіщення (месенджер)

- Назва: Перевірка відображення сповіщення з додатка (напр., Telegram).
- Pre-condition: Годинник підключено, у додатку "Aura Connect" надано дозвіл на сповіщення від Telegram.
- Кроки:
  1. Надіслати повідомлення в Telegram на акаунт, підключений до смартфона.
- Expected Result: Годинник вібрює, на дисплеї з'являється іконка Telegram, ім'я відправника та текст повідомлення.
- Post-condition: Сповіщення переглянуто або закрито.

#### Test Case 8 (Негативний): Втрата з'єднання Bluetooth

- Назва: Перевірка реакції на втрату зв'язку зі смартфоном.
- Pre-condition: Годинник підключено до смартфона.
- Кроки:
  1. Вимкнути Bluetooth на смартфоні (або віднести смартфон на >15 метрів).
- Expected Result: Годинник вібрює та/або показує на екрані іконку "Розірвано з'єднання" (наприклад, перекреслений телефон).
- Post-condition: Bluetooth на смартфоні знову увімкнено, годинник автоматично підключається.

#### Test Case 9: Робота GPS під час тренування

- Назва: Перевірка точності запису маршруту GPS.
- Pre-condition: Годинник заряджений, знаходиться на відкритій місцевості (не в приміщенні).

- Кроки:
  1. Відкрити додаток "Тренування".
  2. Обрати "Біг на вулиці".
  3. Дочекатися сигналу про успішне з'єднання з GPS (іконка GPS стає зеленою).
  4. Натиснути "Старт".
  5. Пробігти/пройти 500 метрів відомим маршрутом.
  6. Натиснути "Стоп" і "Зберегти".
- Expected Result: Годинник коректно фіксує початок тренування. Після збереження, у додатку на смартфоні відображається карта з пройденим маршрутом, який відповідає реальному шляху.
- Post-condition: Тренування збережено.

### 3. Тестування датчиків та здоров'я

#### Test Case 10: Вимірювання пульсу

- Назва: Перевірка роботи оптичного датчика серцевого ритму.
- Pre-condition: Годинник надійно закріплений на зап'ясті. Користувач у стані спокою.
- Кроки:
  1. Зайти в меню "Пульс".
  2. Натиснути "Виміряти".
  3. Сидіти нерухомо протягом 15 секунд.
- Expected Result: Годинник відображає значення пульсу (напр., "65 уд/хв"). Значення є реалістичним для стану спокою (не 0, не 300).
- Post-condition: Вимірювання завершено.

#### Test Case 11: Робота крокоміра

- Назва: Перевірка точності підрахунку кроків.
- Pre-condition: Годинник на руці, лічильник кроків на початку дня показує 0 (або відоме число).
- Кроки:

1. Пройти рівно 100 кроків (рахуючи вголос).
  2. Зупинитися.
  3. Перевірити показник крокоміра на годиннику.
- Expected Result: Лічильник кроків збільшився на 95-105 кроків (враховуючи похибку  $\pm 5\%$ ).
  - Post-condition: Користувач продовжує рух.

Test Case 12 (Негативний): Вимірювання пульсу "на столі"

- Назва: Перевірка, що датчик не вимірює пульс на неживих об'єктах.
  - Pre-condition: Годинник знято з руки.
  - Кроки:
    1. Покласти годинник на стіл датчиком донизу.
    2. Запустити вимірювання пульсу.
- Expected Result: Годинник показує помилку ("Неможливо виміряти", "Одягніть годинник щільніше") або "---". Він *не повинен* показувати цифрове значення пульсу.
  - Post-condition: Вимірювання скасовано.

#### 4. Тестування взаємодії (UI/UX)

Test Case 13: Навігація за допомогою коліщатка (Digital Crown)

- Назва: Перевірка прокрутки списків за допомогою коліщатка.
  - Pre-condition: Годинник увімкнено, відкрито меню "Налаштування" (довгий список).
  - Кроки:
    1. Прокрутити коліщатко "від себе" (вгору).
    2. Прокрутити коліщатко "на себе" (вниз).
- Expected Result: При прокручуванні вгору список рухається вниз. При прокручуванні вниз – вгору. Прокрутка плавна, з тактильним відгуком (вібрацією).
  - Post-condition: Вихід з меню "Налаштування".

Test Case 14: Використання бічної кнопки

- Назва: Перевірка функції бічної кнопки (повернення/меню).
- Pre-condition: Годинник увімкнено, відкрито додаток "Погода".
- Кроки:
  1. Натиснути бічну кнопку один раз.
- Expected Result: Годинник повертається на головний екран (циферблат) АБО відкриває меню додатків (залежно від налаштувань). Дія чітка і відповідає заявленій функції.
- Post-condition: Годинник на головному екрані.

#### Test Case 15: Реакція сенсорного екрана на "свайп"

- Назва: Перевірка жестів "свайп" для доступу до віджетів/сповіщень.
- Pre-condition: Годинник на головному екрані (циферблاتі).
- Кроки:
  1. Провести пальцем зверху вниз (свайп вниз).
  2. Провести пальцем знизу вгору (свайп вгору).
  3. Провести пальцем зліва направо (свайп вправо).
- Expected Result: Свайп вниз відкриває Центр сповіщень. Свайп вгору відкриває Панель швидких налаштувань. Свайп вправо відкриває віджети. Реакція швидка, без затримок.
- Post-condition: Всі меню закрито, годинник на головному екрані.

#### 5. Тестування "заліза" та автономності

##### Test Case 16: Процес бездротової зарядки

- Назва: Перевірка роботи бездротової зарядки.
- Pre-condition: Зарядний пристрій підключено до джерела живлення. Рівень заряду годинника 50%.
- Кроки:
  1. Помістити годинник на магнітний зарядний пристрій.
- Expected Result: Годинник притягується магнітом, екран вмикається, відображається анімація зарядки та поточний відсоток (напр., "Зарядка, 50%").

- Post-condition: Годинник знято з зарядки.

Test Case 17: Попередження про низький заряд

- Назва: Перевірка сповіщення про низький рівень заряду.
- Pre-condition: Годинник використовується, рівень заряду падає до 10%.
- Кроки:
  1. Користуватися годинником, доки заряд не досягне 10%.
- Expected Result: Годинник вібрує та показує сповіщення "Низький заряд. Залишилось 10%. Увімкнути режим енергозбереження?".
- Post-condition: Обрано "Ні" або сповіщення закрито.

Test Case 18: Водостійкість (миття рук)

- Назва: Перевірка базової водостійкості (IP68).
- Pre-condition: Годинник на руці, працює коректно.
- Кроки:
  1. Ретельно вимити руки під проточною водою (15-20 секунд), не знімаючи годинник.
  2. Витерти годинник.
- Expected Result: Годинник продовжує працювати коректно. Сенсорний екран реагує на дотики, звук з динаміка не спотворений (після висихання).
- Post-condition: Годинник працює.

Test Case 19: Заміна ремінця

- Назва: Перевірка механізму кріплення ремінця.
- Pre-condition: Годинник з встановленим ремінцем.
- Кроки:
  1. Натиснути кнопки фіксатора ремінця на тильній стороні.
  2. Витягнути одну частину ремінця.
  3. Вставити нову (або ту ж) частину ремінця назад.

- Expected Result: Ремінець легко від'єднується при натисканні кнопок. Новий ремінець легко вставляється і надійно фіксується (про що свідчить клацання).
- Post-condition: Ремінець надійно закріплений.

Test Case 20: Робота динаміка та мікрофона (дзвінок)

- Назва: Перевірка якості зв'язку при дзвінку з годинника.
- Pre-condition: Годинник підключено до смартфона.
- Кроки:
  1. Зайти в додаток "Телефон" на годиннику.
  2. Набрати номер (або обрати контакт).
  3. Натиснути кнопку виклику.
  4. Провести коротку розмову (5-10 с), говорячи в мікрофон годинника.
- Expected Result: Співрозмовник чітко чує голос (мікрофон працює). Користувач чітко чує співрозмовника з динаміка годинника.
- Post-condition: Дзвінок завершено.

## Контрольні питання

1. Навіщо потрібні тест-кейси?

для:

- **Систематизації:** Забезпечують структурований підхід до тестування, замість хаотичного "натискання кнопок".
- **Покриття:** Гарантують, що всі важливі функції (вимоги) були перевірені.
- **Повторюваності:** Дозволяють точно відтворити тест у майбутньому (наприклад, після виправлення помилки або в новій версії продукту).
- **Комунікації:** Служать чітким завданням для тестувальника і звітом для менеджера чи розробника.
- **Оцінки якості:** Дозволяють об'єктивно оцінити, чи відповідає продукт очікуванням.

2. Основні атрибути Test Case?

- **ID (Ідентифікатор):** Унікальний номер (напр., TC-001).

- **Title (Назва):** Короткий опис суті тесту (напр., "Перевірка входу з коректним паролем").
- **Pre-conditions (Передумови):** Стан системи, який має бути досягнутий до початку тесту (напр., "Користувач на сторінці логіну").
- **Steps (Кроки):** Послідовність дій, які потрібно виконати.
- **Expected Result (Очікуваний результат):** Те, як система повинна відреагувати на кроки.
- **Actual Result (Фактичний результат):** Те, як система відреагувала насправді (заповнюється тестувальником).
- **Status (Статус):** Результат виконання (Pass / Fail / Blocked).
- **(Необов'язково) Post-conditions (Післяумови):** Стан системи після завершення тесту (напр., "Користувач увійшов у систему").
- **(Необов'язково) Priority (Пріоритет):** Наскільки важливий цей тест (High / Medium / Low).

### 3. Типи тест-кейсів.

- **Функціональні (Functional):** Перевіряють, чи працює функція (напр., чи додається товар у кошик).
- **Нефункціональні (Non-functional):** Перевіряють, як працює система.
  - **Тест-кейси продуктивності (Performance):** Перевірка швидкості, стабільності, роботи під навантаженням.
  - **Тест-кейси юзабіліті (Usability):** Перевірка, наскільки зручно користуватися інтерфейсом.
  - **Тест-кейси сумісності (Compatibility):** Перевірка роботи на різних пристроях, браузерах, ОС.
- **Позитивні (Positive):** Використовують коректні дані та очікуваний сценарій (напр., вхід з правильним паролем).
- **Негативні (Negative):** Використовують некоректні дані або несподівані умови (напр., вхід з неправильним паролем).
- **Тест-кейси GUI (Graphical User Interface):** Перевірка візуальної частини (чи коректно відображаються кнопки, кольори, шрифти).

### 4. Що таке негативний тест-кейс?

*це тест, який перевіряє, як система поводиться при некоректних або неочікуваних діях користувача з метою переконатися, що*

*система правильно обробляє помилки і не ламається після певних таких дій*

5. Що повинен знати тестувальник?

- основи життєвого циклу розробки ПЗ;
- принципи тестування;
- види тестування;
- інструменти для тест-менеджменту (*Jira, TestRail тощо*);
- методи створення тест-кейсів і звітів про помилки;
- основи роботи з вимогами, логікою системи;
- базові знання інструментів (ПЗ/ЯП) для технічного тестування.

6. Скільки основних принципів тестування?

*сім основних принципів тестування:*

1. Тестування показує наявність дефектів, а не їх відсутність.
2. Повне тестування неможливе.
3. Раннє тестування економить час і кошти.
4. Дефекти мають тенденцію концентруватися в певних модулях.
5. Повторне тестування без змін може не знайти нових помилок.
6. Тестування залежить від контексту.
7. Відсутність помилок не означає якісний продукт.

## Лабораторна №4

**Тема:** AWS S3.

**Мета:** Набування навичок у створення і розміщення статичної веб-сторінки на AWS S3.

**Очікувані результати навчання:** уміння створити і розмістити сторінку з власними даними на ресурсі AWS S3.

Короткі теоретичні відомості

**Amazon S3** (Simple Storage Service) — це сервіс від компанії **Amazon Web Services** для зберігання об'єктів (файлів) у хмарі. Він дозволяє зберігати будь-який тип даних: документи, зображення, відео, резервні копії, веб-сторінки тощо.

AWS S3 — це гнучкий, потужний та простий сервіс для зберігання даних, який також можна використовувати для хостингу простих статичних сайтів.

Він дозволяє завантажити HTML-файл і отримати публічну URL-адресу для доступу через інтернет.

## Крок 1. Реєстрація в AWS

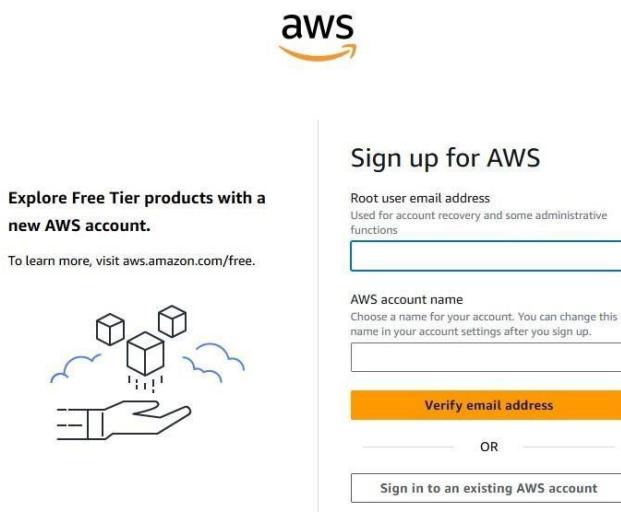


Рис.1. Введення даних для реєстрації

## Крок 2. Створення S3 бакету.

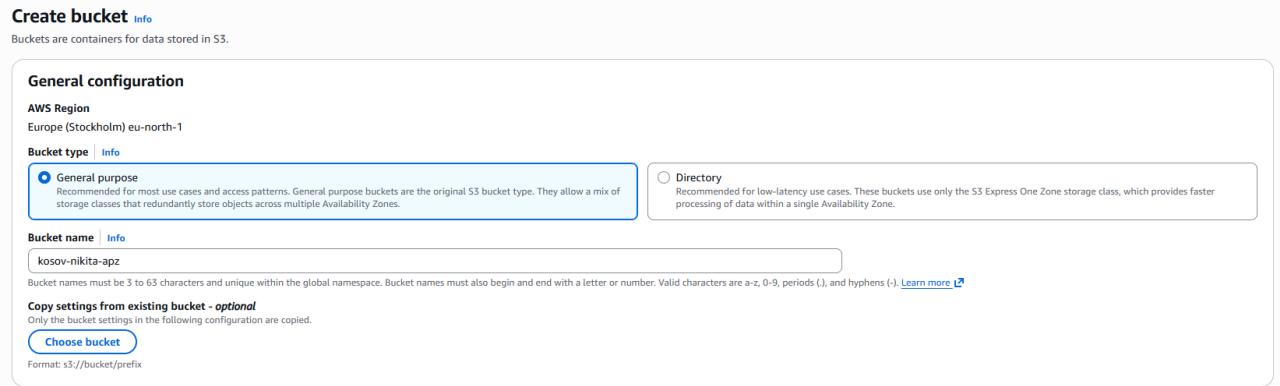


Рис.2. Створення S3 бакету

### Крок 3. Завантаження HTML-файлу.



Рис.3. Створення index.html з кодом

Object overview	
Owner	70e83d8b577e1e3fa81807555dbf88c1e31f523b7f0f8a007439a19cceeaacb8
AWS Region	Europe (Stockholm) eu-north-1
Last modified	November 15, 2025, 14:37:24 (UTC+02:00)
Size	228.0 B
Type	html
Key	<a href="#">index.html</a>
S3 URI	<a href="#">s3://kosov-nikita-apz/index.html</a>
Amazon Resource Name (ARN)	<a href="#">arn:aws:s3:::kosov-nikita-apz/index.html</a>
Entity tag (Etag)	<a href="#">36baeb86458fd7852b603334b427e5fc</a>
Object URL	<a href="#">https://kosov-nikita-apz.s3.eu-north-1.amazonaws.com/index.html</a>

Рис.4. Додали до S3 бакету index.html

### Крок 4. Налаштування хостингу та отримання адреси сторінки.

1. Поверніться в AWS S3 та відкрийте створений бакет. Оберіть Properties та прокрутіть до Static website hosting. Далі натисніть Edit і

увімкніть "Enable". У полі Index document вкажіть index.html і натисніть

Save changes:

The screenshot shows the 'Static website hosting' configuration for an S3 bucket. At the top, it says 'Static website hosting' and provides a link to learn more. A note recommends using AWS Amplify Hosting for static website hosting, with links to Amplify Hosting and existing apps. Below this, the 'S3 static website hosting' section is shown as 'Disabled'. An 'Edit static website hosting' button is available. The 'Edit static website hosting' page is then displayed, showing the configuration details: 'Static website hosting' is enabled, 'Hosting type' is set to 'Host a static website' (selected), and the 'Index document' field contains 'index.html'. There are also sections for 'Error document - optional' (set to 'error.html') and 'Redirection rules - optional'.

**Static website hosting**  
Use this bucket to host a website or redirect requests. [Learn more](#)

**We recommend using AWS Amplify Hosting for static website hosting**  
Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about [Amplify Hosting](#) or [View your existing Amplify apps](#)

**S3 static website hosting**  
Disabled

[Edit static website hosting](#) Info

**Static website hosting**  
Use this bucket to host a website or redirect requests. [Learn more](#)

**Static website hosting**  
 Disable  
 Enable

**Hosting type**  
 Host a static website  
Use the bucket endpoint as the web address. [Learn more](#)  
 Redirect requests for an object  
Redirect requests to another bucket or domain. [Learn more](#)

**Index document**  
Specify the home or default page of the website.  
index.html

**Error document - optional**  
This is returned when an error occurs.  
error.html

**Redirection rules - optional**  
Redirection rules, written in JSON, automatically redirect webpage requests for specific content. [Learn more](#)

Рис.5. Створення посилання

2. Тепер знову у вкладці Properties, знайдіть Static website hosting, там буде ваш URL, наприклад <http://kosov-nikita-apz.s3-website.eu-north-1.amazonaws.com>

## Static website hosting

Use this bucket to host a website or redirect requests. [Learn more ↗](#)

**We recommend using AWS Amplify Hosting for static website hosting**

Deploy a fast, secure, and reliable website quickly with AWS Amplify Hosting. Learn more about AWS Amplify Hosting.

### S3 static website hosting

Enabled

#### Hosting type

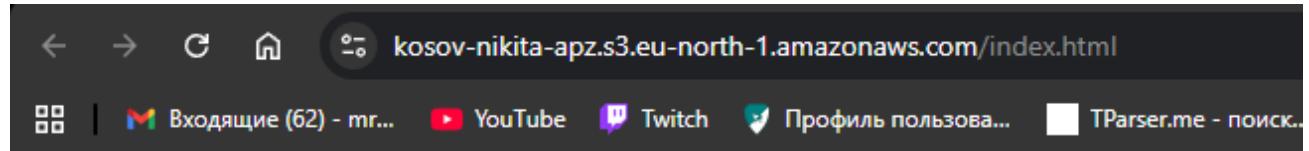
Bucket hosting

#### Bucket website endpoint

When you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint.

<http://kosov-nikita-apz.s3-website.eu-north-1.amazonaws.com> ↗

Рис.6. Створене посилання



## Косов Нікіта Олександрович

122-22-1

Рис.5. Результат виконання практичної роботи

### Створені посилання

Лінка на статичний сайт:

<http://kosov-nikita-apz.s3-website.eu-north-1.amazonaws.com>

Лінка на сам index.html:

<https://kosov-nikita-apz.s3.eu-north-1.amazonaws.com/index.html>

Висновок: У ході виконання роботи було опрацьовано практичний матеріал щодо сервісу AWS S3 та його використання для розміщення статичних веб-сайтів. Було засвоєно поняття «S3 бакет» (bucket) як основного контейнера для зберігання об'єктів та вивчено його ключові налаштування, досліджено процес конфігурації бакету для публічного доступу, зокрема вимкнення опції "Block all public access" та надання дозволів окремим файлам, також проаналізовано механізм активації "Static website hosting" у властивостях бакету, що включає вказання індексного документа (index.html).

## Лабораторна №5

Тема: AWS EC2.

Мета: набування навичок створення та розміщення віртуального сервера за допомогою AWS EC2. Хід роботи

### Хід роботи

#### Крок 1. Створюємо та запускаємо Instance.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The top navigation bar includes the AWS logo, search bar, and account information (Account ID: 1367-1330-0990, Europe (Stockholm)). The main header says 'Launch an instance' with a 'Take a walkthrough' button and a 'Do not show me this message again' checkbox.

The 'Name and tags' step is active, showing a 'Name' field containing 'AP2-My-PC'. Below it is a 'Software Image (AMI)' section for 'Amazon Linux 2023 AMI 2023.9.2...', a 'Virtual server type (instance type)' section for 't3.micro', and a 'Storage (volumes)' section indicating 1 volume(s) - 8 GiB.

The 'Application and OS Images (Amazon Machine Image)' step is shown below, featuring a search bar and a grid of recent AMIs: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian. A note at the bottom says 'Including AMIs from AWS, Marketplace and the Community'.

At the bottom right of the wizard, there are 'Cancel', 'Launch instance', and 'Preview code' buttons.

Створення key pair:

Free tier eligible



## Create key pair

### Key pair name

Key pairs allow you to connect to your instance securely.

apz-my-key

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

### Key pair type



RSA

RSA encrypted private and public key pair



ED25519

ED25519 encrypted private and public key pair (Not supported for Windows instances)

### Private key file format



.pem

For use with OpenSSH



.ppk

For use with PuTTY



When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more ↗](#)

Cancel

Create key pair

Configure Storage

Налаштування Configure Storage:

**▼ Configure storage** [Info](#)

1x  GiB  [▼](#) Root volume, 3000 IOPS, Not encrypted

[Add new volume](#)

The selected AMI contains instance store volumes, however the instance does not allow any instance store volumes. None of the instance store volumes will be accessible from the instance.

[ⓘ Click refresh to view backup information](#)  
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems

**Успішно ініційовано запуск екземпляра:**

 **Success**  
Successfully initiated launch of instance (i-...)

**Крок 2. Отримання зашифрованого паролю.**

## Get Windows password [Info](#)

Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

**Instance ID**

 (APZ-My-PC)

**Key pair associated with this instance**

apz-my-key

**Private key**

Either upload your private key file or copy and paste its contents into the field below.

[Upload private key file](#)

apz-my-key.pem

X

1.68 KB

[Private key contents](#)

**Крок 3. Підключаємося до створеного ПК.**

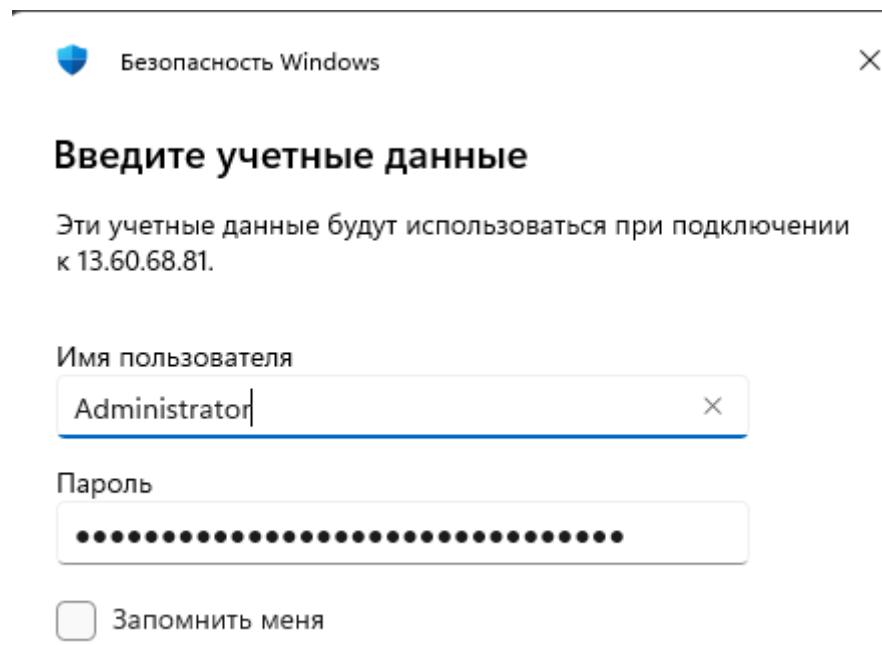


Рис.13 – Екран інстансу.

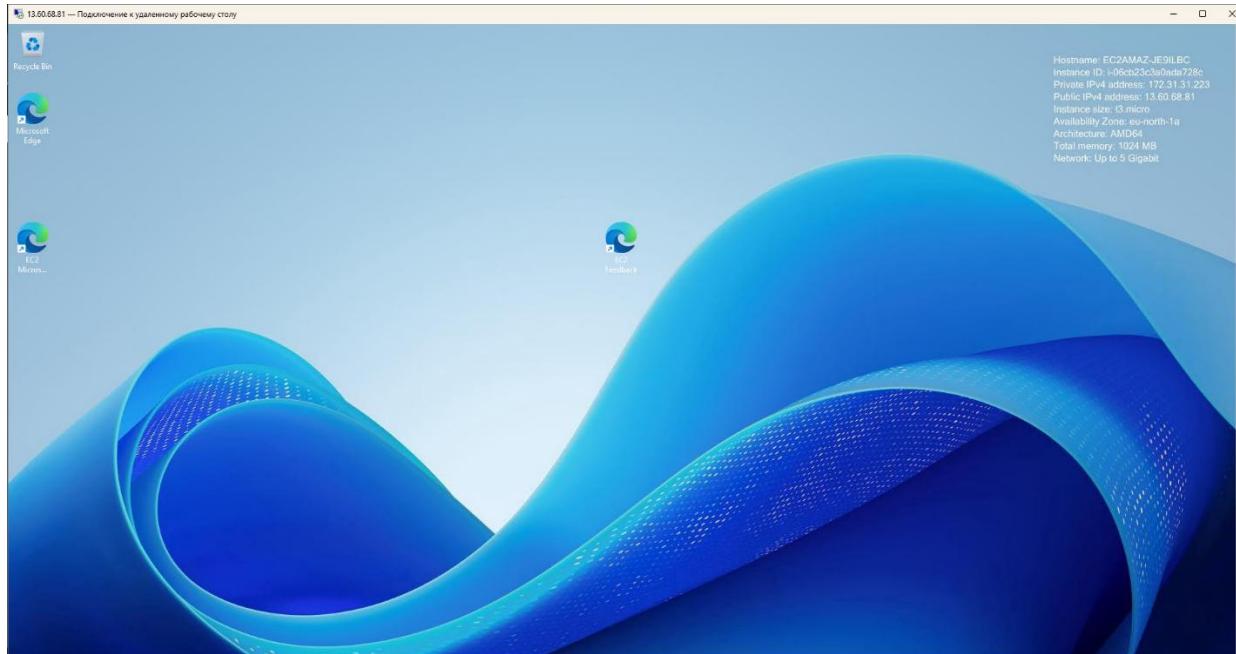


Рис – Екран інстансу.

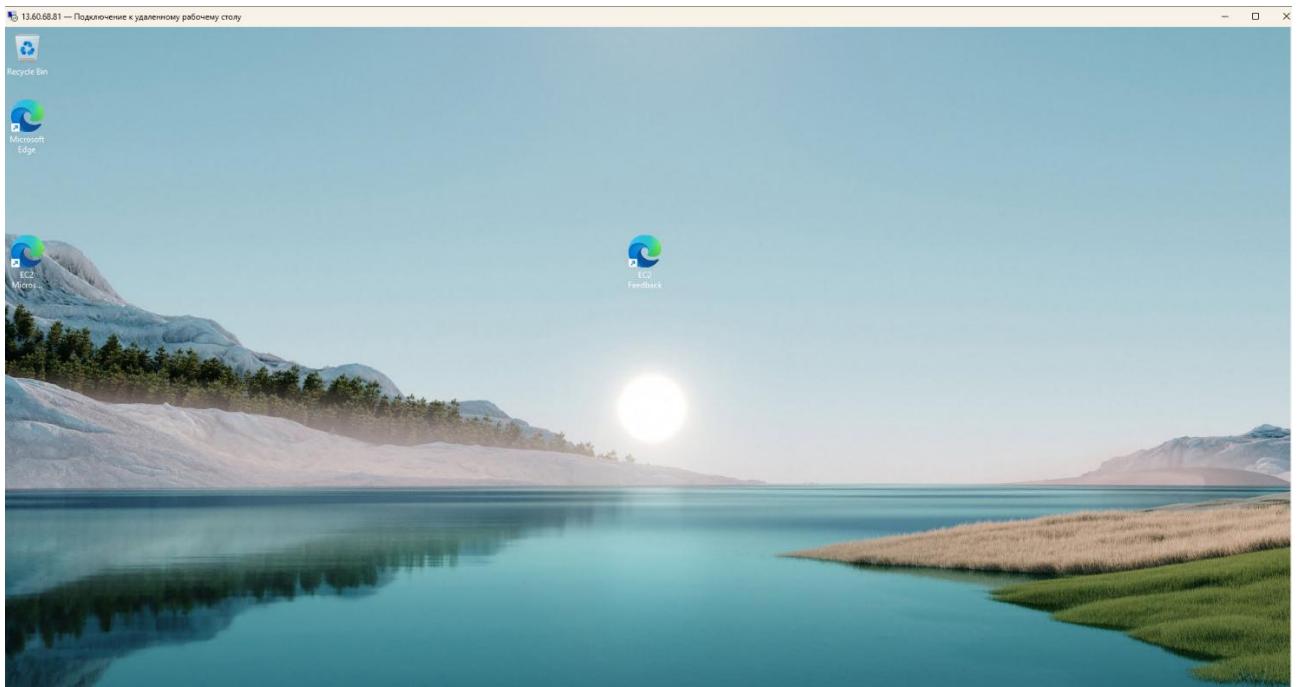


Рис – Кінцевий результат.

Дані для підключення:

IP: 13.60.68.81

User: Administrator

Password: !c7ugFMG;);zoovxF)-BkT87B!&mVU%o

Висновки: на цій практичній роботі я на навчився створювати, редагувати, віддалено підключатися до створеного instance через системну програму remote control app (Windows).