

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



Факультет інформаційних технологій
Кафедра системного аналізу та управління

Звіт
з практичної роботи №2
«Аналіз програмного забезпечення»

Виконав:
Студент групи 121-22-3
Марочка В. І.
Перевірили:
доц. Мінєєв О.С.
ас. Шевченко Ю.О.

Дніпро
2025

Тема: Створення і налаштування профілю у системі Git.

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

Хід роботи

Для створення облікового запису у системі GitHub, необхідно виконати декілька простих дій:

- 1) Перейдіть на сайт <https://github.com/>
- 2) Натисніть кнопку “Sign up” у верхньому правому куті сторінки.
- 3) Заповніть реєстраційну форму, вказавши ваше ім’я користувача, адресу електронної пошти та пароль.
- 4) Натисніть “Create account” та дотримуйтесь інструкцій на екрані для завершення реєстрації.

Sign up for GitHub

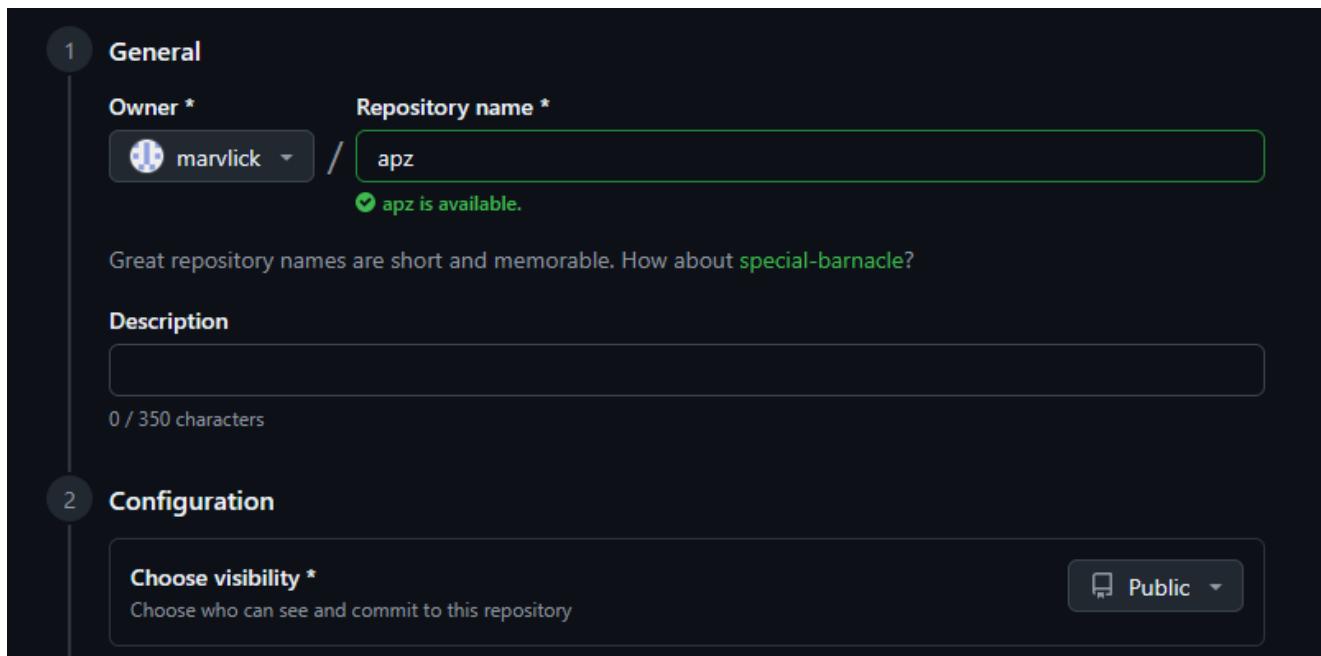
The screenshot shows the GitHub sign-up form. It includes social login options for Google and Apple, followed by an "or" separator. The main form starts with an "Email*" field, which is a placeholder for "Email". Below it is a "Password*" field, also a placeholder for "Password". A note states: "Password should be at least 15 characters OR at least 8 characters including a number and a lowercase letter." The next field is "Username*", with a placeholder for "Username". A note specifies: "Username may only contain alphanumeric characters or single hyphens, and cannot begin or end with a hyphen." The final section is "Your Country/Region*", with "Ukraine" selected from a dropdown menu. A note below the dropdown says: "For compliance reasons, we're required to collect country information to send you occasional updates and announcements." At the bottom, there's an "Email preferences" section with a checkbox for "Receive occasional product updates and announcements". The final button is a large black "Create account >" button.

Для того, щоб створити новий репозиторій на GitHub, виконайте наступні дії:

- 1) Увійдіть у ваш обліковий запис на GitHub - “Sign up” у верхньому правому куті сторінки.
- 2) Натисніть на кнопку “New” у верхньому правому куті сторінки, щоб створити новий репозиторій.
- 3) Введіть назву вашого репозиторію і, за бажанням, його опис.
- 4) Виберіть, щоб ваш репозиторій був репозиторій публічним (Public).

Публічний репозиторій і всі файли в ньому зможе побачити будь-яка людина в інтернеті, в той час, як приватний буде доступний лише тим людям, кому ви безпосередньо надасте доступ.

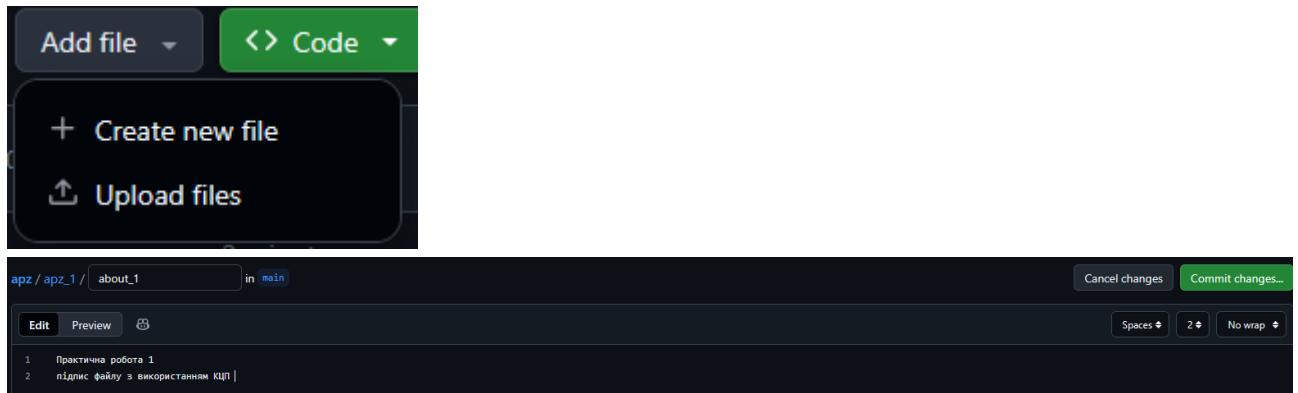
- 5) Натисніть кнопку “Create repository”



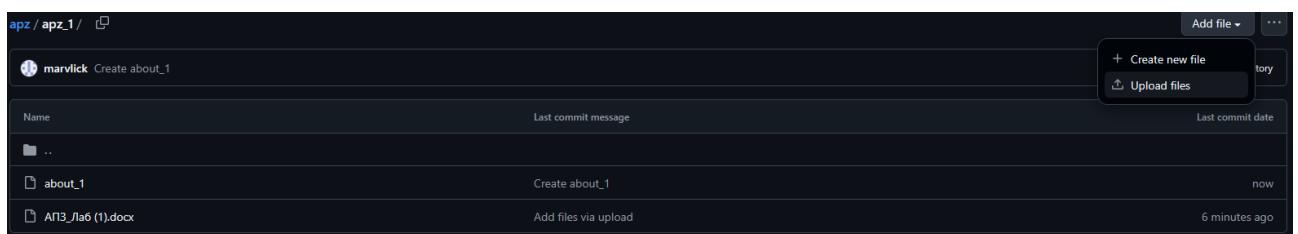
The screenshot shows the GitHub repository creation interface. It's divided into two main sections: 'General' (step 1) and 'Configuration' (step 2).
Step 1: General
Owner: marvlick
Repository name: apz
Description: (empty field)
Note: Great repository names are short and memorable. How about [special-barnacle](#)?
Step 2: Configuration
Choose visibility: Public

Далі для кожної лабораторної робото зробіть окрему папку і розмістить в ній звіт для перевірки викладачем:

1) Створити новий файл, наприклад текстовий документ (рис. 2.3), який потім при необхідності можна видалити чи замінити на звіт з практичної роботи.



2) В кожну окрему папку додаємо файл звіту, скориставшись «Upload files» у верхньому куті екрану.



Контрольні питання

1. Що таке GIT?

GIT — це система контролю версій, яка відстежує зміни в коді та дозволяє працювати в команді.

2. Що таке репозиторій у GIT?

Репозиторій — це місце, де зберігається код і вся історія його змін.

3. Які переваги використання GIT?

- Історія змін
- Паралельна робота в гілках
- Просте повернення до попередніх версій
- Спільна робота через GitHub/GitLab
- Швидкість і надійність

4. Яка мова використовується в GIT?

GIT не має «мови програмування» — це інструмент.
Команди виконуються через CLI (термінал).

5. Як можна створити репозиторій у Git?

У папці проєкту:

```
git init
```

або клонувати вже існуючий:

```
git clone URL
```

6. Яка команда використовується для видалення гілки?

Локально:

```
git branch -d branch_name
```

Примусово:

```
git branch -D branch_name
```

7. Що таке контроль версій GIT?

Це система, яка зберігає історію змін файлів, дозволяє повернутися назад, створювати гілки та об'єднувати їх.

8. Як можна виправити несправний комміт?

Найчастіше використовують:

```
git commit --amend
```

(виправити останній комміт)

або

```
git revert <hash>
```

(створити комміт, що скасовує зміни)

9. Як дізнатися, чи гілку вже об'єднано в master?

```
git branch --merged master
```

Якщо твоя гілка в списку — вона вже злита.