

Звіт
До лабораторної роботи №2
З предмету
«Інструментальні засоби розробки програмного забезпечення»

Виконав
Студент групи ІПС-21
Лобач Роман Вадимович

Київ-2025

Тема: Документація коду (Doxygen/JavaDoc/...) і автоматична публікація через GitHub Actions та GitHub Pages

Мета роботи:

Навчитися документувати програмний код стандартними форматами (Doxygen/JavaDoc), автоматично генерувати HTML-документацію в CI/CD та публікувати її на GitHub Pages після кожного оновлення репозиторію.

Хід виконання роботи

1. Підготовка

- Для роботи використано код із попередньої лабораторної.
- Створено нову гілку: **feature/docs-ci**.
- У файлі **README.md** додано розділ *Documentation* із коротким описом та посиланням на майбутню сторінку GitHub Pages.

2. Документування коду

- До основних класів та методів було додано коментарі у форматі **Doxygen**.
- Використані стандартні теги:
 - **@brief** — короткий опис класу чи методу.
 - **@param** — опис параметрів.
 - **@return** — опис значення, що повертається.
 - **@throws** — перелік можливих виключень.
 - **@example** — приклади використання.
- Коментарі додані у вигляді **/** ... */** перед класами та функціями.

3. Генерація документації

- Створено конфігураційний файл **Doxyfile**.
- Локально перевірено генерацію HTML-версії документації командою **doxygen Doxyfile**.

- Після виконання генерації в каталозі *docs/html* створено повноцінний сайт документації.

4. CI/CD через GitHub Actions

- У директорії `.github/workflows/` створено файл **docs.yml**, який виконує:
 - встановлення Doxygen;
 - генерацію документації;
 - деплой готової документації на GitHub Pages.
- Workflow виконується автоматично після кожного push у гілку main.

5. Публікація документації

- У налаштуваннях репозиторію активовано GitHub Pages (Source → GitHub Actions).
- Після першого автоматичного запуску CI/CD було перевірено, що сторінка документації доступна за вказаним URL.
- Після кожного нового коміту документація оновлюється автоматично.

Скріншот git історії

```
roma@MacBook-Air-Roman DataInputService % git log --oneline
3d1c870 (HEAD -> my-tests-branch, origin/my-tests-branch) fix Doxygen.yml
199c344 Add doxygen.yml
b998102 docs: Add Doxygen comments to DataInputService class
44678cc docs: Add Doxygen comments to UserInput class
ee31310 docs: Add Doxygen comments to Config class
b22fde6 docs: Add Doxygen comments to ValidationResult class
8ef14a4 Add Doxyfile
305480c Add report
```

Результати роботи

- **Посилання на репозиторій GitHub:** [<https://github.com/Roma3214185/dev-tools-lab-1>]
- **Посилання на GitHub Pages з документацією:** [<https://roma3214185.github.io/dev-tools-lab-1/?t=1>]
- Основні результати:
 - Додано коментарі у стандартному форматі документації.
 - Створено Doxufіle та налаштовано локальну генерацію документації.
 - Налаштовано GitHub Actions для автоматичної генерації та публікації.
 - Документація оновлюється автоматично після кожного push

Висновок

У ході виконання лабораторної роботи я навчився документувати код за допомогою Doxygen, створювати HTML-документацію та налаштовувати автоматичне оновлення документації через GitHub Actions. Також отримав практичний досвід роботи з GitHub Pages та CI/CD. Набуті навички дозволяють створювати професійну, підтримувану і доступну документацію для будь-якого програмного проєкту.