

Звіт

До лабораторної роботи №1

З предмету

«Інструментальні засоби розробки програмного забезпечення»

Виконав

Студент групи ІПС-21

Лобач Роман Вадимович

Київ-2025

Тема та мета:

Тема роботи – розробка модульних тестів для перевірки коректності введених користувачем даних у модулі DataInputService. Мета роботи – забезпечити надійну валідацію полів користувача (ім'я, пароль, тег, електронна пошта) та відпрацювати навички написання юніт-тестів із Catch2.

Короткий теоретичний вступ:

Модуль DataInputService реалізує перевірку таких даних:

- Ім'я користувача: перевірка довжини, допустимих символів, UTF-8.
- Пароль: перевірка довжини, пробілів, контрольних символів, допустимих знаків.
- Теги: перевірка довжини, допустимих символів, відсутності подвійних підкреслень.
- Електронна пошта: перевірка формату, локальної частини, домену, IP-літер, плюс-аліасів.

Юніт-тести дозволяють автоматично перевіряти правильність роботи кожної функції і запобігти помилкам при зміні коду.

Посилання на репозиторій GitHub:

<https://github.com/Roma3214185/dev-tools-lab-1>

Основні етапи роботи:

1. Створено гілку my-tests-branch.
2. Написано тестові секції для всіх полів (name, password, tag, email) із Catch2.
3. Виконано кілька комітів із описом змін:
 - Додано перевірку імені користувача
 - Додано перевірку пароля
 - Додано перевірку тегів
 - Додано перевірку електронної пошти

4. Створено Pull Request для об'єднання гілки з основною.

Скріншот основних етапів роботи:

```
RomanDataInputService % git log --oneline
79315d2 (HEAD -> my-tests-branch, origin/my-tests-branch) Refactor tests
dc6b26f Add test for email and chained validation
456fc0b Add tests for name and tag boundaries
c42c3f9 Implement detailed tag and password checking
bf4a707 Implement detailed name and email checking
f3624ec Refactor DataInputService.h using new structs
77c1555 Add structs for input validation and configuration
f4b912f (main) Added tests for name
eefda31 Added tests for email
68bd6ea Added tests for tag
88fb072 Added test for password
205ba5c (origin/main) Initial project version
```

Опис процесу написання юніт-тестів:

- Використано фреймворк Catch2.
- Для кожного поля створено окремий TEST_CASE з різними SECTION, які перевіряють:
 - Межі довжини (мінімальна/максимальна)
 - Наявність контрольних і недопустимих символів
 - Підтримку Unicode і UTF-8
 - Позитивні та негативні кейси
- Для комплексної перевірки введених даних використано функцію validateUserInput, яка послідовно викликає всі валідатори.

- Для перевірки кількох варіантів паролів застосовано параметризовані тести через GENERATE.

Висновки про набуті навички:

- Ознайомився з принципами написання юніт-тестів у C++ за допомогою Catch2.
- Навчився перевіряти edge-case сценарії для текстових даних, включно з UTF-8.
- Отримав практичний досвід у роботі з Git: створення гілок, коміти, Pull Request.
- Зрозумів важливість комплексного тестування для забезпечення стабільності модулів.