

**Автор:**

Гирич Артем Олександрович,  
студент ІІКНм групи

**Науковий керівник:**

Малежик Петро Михайлович  
Доктор педагогічних наук, доцент

## АРХІТЕКТУРА МУЛЬТИ-ПРОЕКТНОГО ФРЕЙМВОРКА ДЛЯ ТЕСТУВАННЯ UI

**Анотація.** Метою дослідження є розробка архітектури системи для автоматизованого тестування користувацьких інтерфейсів (UI) у декількох схожих проєктах за допомогою фреймворка Playwright. Завдання дослідження полягає у створенні багатофункціонального та уніфікованого рішення для тестування, що дозволить скоротити час розробки та підтримки тестів. Об'єктом дослідження є автоматизація тестування UI. Предметом дослідження є методологія побудови мульти-проєктного фреймворка для тестування. Використані методи включають аналіз вимог до автоматизації, розробку архітектурного рішення та інтеграцію з інструментами тестування. Результатом роботи є побудова гнучкої архітектури, що забезпечує ефективне тестування декількох проєктів в єдиній тестовій системі.

**Ключові слова:** автоматизація тестування, тестування UI, Playwright, мульти-проєктний фреймворк.

**Вступ.** Автоматизація тестування користувацьких інтерфейсів є важливою частиною процесу забезпечення якості програмного забезпечення. Зі зростанням кількості схожих проєктів, зокрема веб-додатків, виникає необхідність створення універсального рішення, яке дозволить уникнути дублювання роботи та забезпечує швидке розширення тестового покриття. Playwright є сучасним інструментом для автоматизації тестування UI, який підтримує різні браузері, платформи та сценарії. Використання Playwright для побудови мульти-проєктного фреймворка відкриває можливості для уніфікації процесу тестування, скорочення витрат на підтримку та підвищення продуктивності QA-команд. Розробка такого фреймворка стає актуальною задачею, враховуючи потребу в централізованому управлінні тестами для різних проєктів, які мають схожі, але не ідентичні компоненти UI.

**Постановка задачі.** Основна проблема полягає у відсутності єдиного підходу до автоматизації тестування UI для декількох схожих проєктів. Створення окремих фреймворків для кожного проєкту призводить до значних витрат часу і ресурсів.

Метою є розробка архітектури, яка дозволить:

- Використовувати єдиний набір інструментів і підходів для тестування UI різних проєктів.
- Забезпечити масштабованість та повторне використання тестів.
- Мінімізувати складність внесення змін та інтеграції нових проєктів у тестову екосистему.

**Мета роботи.** Метою дослідження є створення архітектури мульти-проєктного фреймворка для тестування UI за допомогою Playwright на основі JavaScript. Такий фреймворк забезпечує ефективне управління тестами, дозволить швидко адаптувати тести до змін у UI, а також спростить інтеграцію з існуючими інструментами CI/CD.

**Методи дослідження.**

- Аналіз вимог до тестування UI.
- Проєктування гнучкої архітектури для тестів.
- Реалізація модульної структури для підтримки схожих компонентів UI.
- Інтеграція з існуючими процесами CI/CD (за необхідності).

**Результати.**

Розроблений фреймворк дозволяє автоматизувати тестування UI для декількох проектів в єдиному середовищі. Реалізація модульного підходу забезпечує простоту розширення функціональності, повторне використання коду та зниження витрат на підтримку.

**Висновки.** Результатом роботи стала архітектура, яка забезпечує ефективне тестування UI для багатьох схожих проектів за допомогою фреймворка Playwright. Використання мульти-проектного підходу спрощує процес автоматизації, знижує витрати на розробку та дозволяє QA-командам сфокусуватися на перевірці ключових функціональностей замість підтримки тестової інфраструктури.

### Список використаних джерел

1. Open-Source. Playwright official documentation.  
<https://playwright.dev/> (date of access: 19.12.2024).