

# ВИКОРИСТАННЯ ТЕХНОЛОГІЇ WEBDRIVER ДЛЯ РОЗРОБКИ ТЕСТОВОГО АВТОМАТИЗОВАНОГО ФРЕЙМВОРКУ

<sup>1</sup>Вінницький національний технічний університет

## **Анотація**

*Наведено короткі відомості про сучасні тенденції в розвитку технологій для написання тестових автоматизованих фреймворків, описано переваги технології WebDriver, її способи використання та можливості щодо написання та автоматизації тестових сценаріїв. Також наведено приклад архітектури з використання технології.*

**Ключові слова:** WebDriver, IT, NodeJS, TAF, JavaScript, Chrome Devtools, JSON.

## **Abstract**

*The article provides brief information about current trends in the development of technologies for writing test automated frameworks, describes the advantages of WebDriver technology, its ways of using it, and opportunities for writing and automating test scripts. An example of an architecture using the technology is also given.*

**Keywords:** WebDriver, IT, NodeJS, TAF, JavaScript, Chrome Devtools, JSON.

## **Вступ**

Тестові автоматизовані фреймворки дозволяють тестувальникам автоматизувати роботу, спрощуючи розробку і виконання тестів. Типова платформа автоматизації надає середовище для виконання планів тестування та створення відтворюваних вихідних даних. Це спеціалізовані інструменти, які допомагають у повсякденних завданнях автоматизації тестування. Будь то програма запуску тестів, засіб запису дій або засіб веб-тестування, вона призначена для того, щоб зняти всю важку роботу по створенню тестових сценаріїв і залишити розробнику більше часу для перевірки якості.

Автоматизація тестування [1] – це перевірений і економічно ефективний підхід до вдосконалення розробки програмного забезпечення. Таким чином, вибір найкращої платформи автоматизації тестування може виявитися вирішальним для результатів тестування і термінів контролю якості.

Webdriver – це прогресивна платформа автоматизації, створена для автоматизації сучасних веб і мобільних додатків. Вона спрощує взаємодію з додатком і надає набір плагінів, які допомагають створювати масштабований, надійний і стабільний набір тестів.

*Метою дослідження є вивчення переваг та особливостей технології Webdriver при розробці тестових автоматизованих фреймворків.*

*Об'єктом дослідження є процес розробки тестових автоматизованих фреймворків з використанням технології Webdriver.*

*Предмет дослідження – технологія Webdriver.*

## **Основна частина**

WebDriver [2] є популярним і вважається одним з кращих фреймворків автоматизації тестування, які працюють як в протоколі Webdriver, так і в протоколі Chrome Devtools. Він заснований на Node.js і має відкритий вихідний код за своєю природою. З Webdriver можна виконувати тестування за допомогою Selenium як в локальній системі, так й інтегрувати їх з хмарними рішеннями, наприклад, такими як LambdaTest.

LambdaTest – це хмарне рішення для кросбраузерного тестування, яке допомагає розробникам

програмного забезпечення тестувати якість і продуктивність своїх веб-додатків в більш ніж 2000 онлайн-браузерах і комбінаціях операційних систем.

Незважаючи на те, що WebdriverIO є новим доповненням до спільноти з відкритим вихідним кодом, він зібрав велику кількість користувачів GitHub і може похвалитися вражаючою зіркою GitHub і кількістю форків.

Webdriver є широко використовувана платформа автоматизації тестування на Javascript. Він володіє різними функціями, такими як підтримка безлічі звітів і служб, тестових фреймворків і тестових програм WDIO CLI.

Головні особливості Webdriver:

- Велика підтримка: WebdriverIO підтримує протокол WebDriver, а Chrome Devtools для проєктів Chromium і Puppeteer.

- Підтримка великих додатків: ця платформа тестування браузера підтримує будь-який мобільний або веб-додаток, розроблене з використанням сучасних веб-технологій і бібліотек, таких як Vue і React.

- Вбудований інтелектуальний селектор: інтелектуальний селектор, вбудований в фреймворк, може легко вибрати конкретний селектор, що належить певній бібліотеці або фреймворку розробки. Наприклад, компоненти React можна вибрати за допомогою команди react\$.

- Інтеграція з Google lighthouse: Webdriver можна легко інтегрувати з Google lighthouse за допомогою плагіна @wdio / devtools-service. За допомогою цієї інтеграції також можна вимірювати показники продуктивності зовнішнього інтерфейсу і додатків PWA.

- Проста крива навчання: оскільки Webdriver надає користувачам знайому інтерактивну поверхню, керувати ним стає дуже просто.

Розглянемо архітектуру Webdriver [3]. На рис.1 показано, що відбувається, коли запускається тестовий сценарій Webdriver.

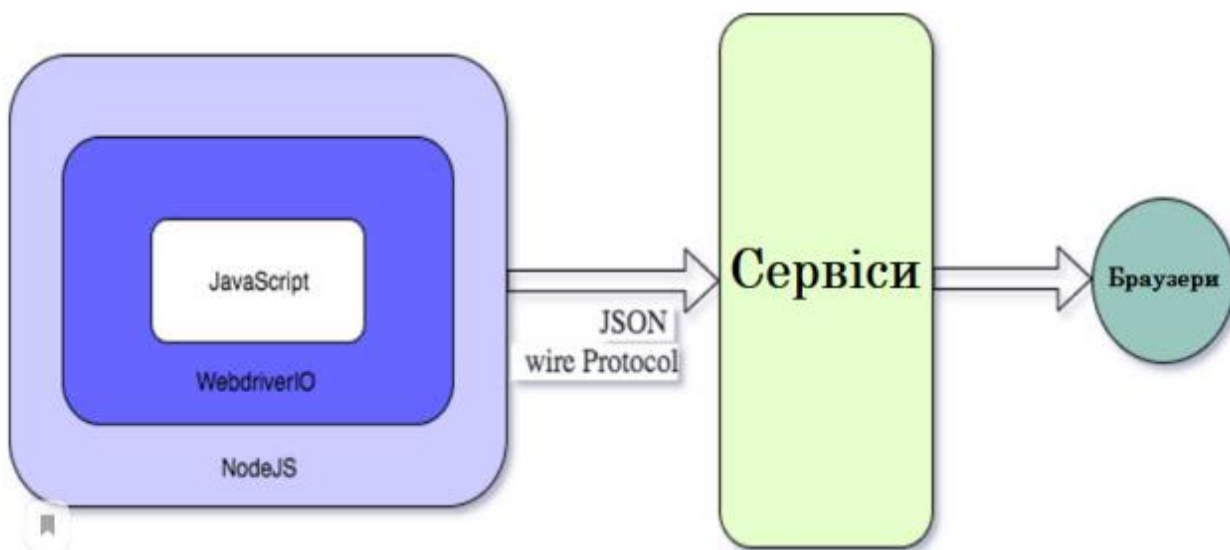


Рисунок 1 – Архітектура тестового автоматизованого фреймворку з використанням технології WebDriver

- NodeJS – це проєкт з відкритим кодом, який допомагає запускати середовище виконання Javascript.

- Webdriver – побудований на вершині NodeJS, який взаємодіє з NodeJS.

- JavaScript – сценарій написаний користувачем за допомогою бібліотеки WebdriverIO.

Цей потік виконується, коли користувач запускає тестовий скрипт Webdriver: JavaScript, написаний користувачем, надсилає запит Webdriver через вузол JS до служб, який має форму http-команди з використанням дротового протоколу JSON. Далі служби пересилають запит браузерам для виконання дій користувача.

### Висновок

Дослідження показало і довело, що технологія WebDriver може спростити написання та

автоматизацію тестових UI сценаріїв. Дана технологія не може повністю замінити популярний підхід до розробки написання UI автоматизованих тестів на базі технології Selenium, але є надзвичайно корисним при розробці автоматизованих тестових фреймворків, особливо у парі із Selenium, оскільки технологія є досить гнучкою та підтримує чимало інших корисних інструментів автоматизації.

#### СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. 13 Best Test Automation Frameworks: The 2021 List [Електронний ресурс]. – Режим доступу до ресурсу: <https://www.lambdatest.com/blog/best-test-automation-frameworks-2021/>.
2. Selenium WebDriverIO Tutorial [Електронний ресурс] – Режим доступу до ресурсу: <https://www.lambdatest.com/blog/webdriverio-tutorial-with-examples-for-selenium-testing/>.
3. Next-gen browser and mobile automation test framework for Node.js. [Електронний ресурс] – Режим доступу до ресурсу: <https://webdriver.io/>.

**Симон Андрій Дмитрович** – студент групи 2ПІ-18б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: [simchik3216@gmail.com](mailto:simchik3216@gmail.com)

**Романюк Оксана Володимирівна** – к.т.н., доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, e-mail: [romaniukoksanav@gmail.com](mailto:romaniukoksanav@gmail.com)

*Symon Andrii* – student of group 2PE-18b, Faculty for Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: [simchik3216@gmail.com](mailto:simchik3216@gmail.com)

*Oksana Romaniuk* – Candidate of Technical Sciences, Associate Professor of the Software Chair, Vinnytsia National Technical University, Vinnytsia, e-mail: [romaniukoksanav@gmail.com](mailto:romaniukoksanav@gmail.com)