

АНАЛІЗ СУЧАСНИХ ІНСТРУМЕНТІВ ДЛЯ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ WEB-ДОДАТКІВ

Вінницький національний технічний університет

Анотація

У даній роботі досліджено сучасні інструменти для автоматизованого тестування web-додатків.

Ключові слова: тестування програмного забезпечення, автоматизація

Abstract

In this article considered the modern tools of automation testing.

Keyword: software testing, automation.

Вступ

Тестування відбувається на протязі всього життєвого циклу ПЗ, починаючи від проектування і закінчуючи етапом експлуатації. Метою тестування є якраз можливість переконатися у відповідності програм заявленим вимогам. Тому забезпечення якості є одним з головних етапів при створенні програмного продукту. [1]

Автоматизоване тестування має досить значні переваги над ручним тестуванням. Швидке виконання, виключення людського фактора та інші переваги – саме це відрізняє ручне та автоматизоване тестування. Існує велика кількість інструментів для автоматизації тестування, кожен з яких має різні сильні і слабкі сторони.

Аналіз інструментів і фреймворків для автоматизації тестування програмного забезпечення вебдодатків

Puppeteer - бібліотека для Node.js, яка дозволяє управляти браузером Chromium без користувальницького інтерфейсу. Інструмент це досить новий. Бібліотека запускає екземпляр Chrome / Chromium і надає набір високорівневих API. Puppeteer використовується для виконання безлічі різних завдань:

- автоматизація збору даних з веб-сайтів;
- створення скріншотів і PDF-файлів;
- тестування розширень Chrome;
- автоматизація тестування веб-інтерфейсів;
- діагностика проблем продуктивності за допомогою таких методів, як захоплення тимчасової шкали трасування веб-сайту. [2]

У порівнянні з Selenium бібліотека Puppeteer не володіє кросбраузерністю, але часто виграє в швидкості, так як не має проміжної ланки у вигляді Selenium server - команди йдуть безпосередньо в браузер.

Selenium - це інструмент автоматизації з відкритим кодом, який зараз користується попитом та широко використовується на ринку. Він може автоматизувати роботу в декількох ОС, таких як Windows, Mac та Linux, а також у таких браузерах, як Firefox, Chrome, IE.

TestComplete - це інструмент автоматизації для десктопних, мобільних та веб-додатків. TestComplete може створювати та запускати функціональні тести інтерфейсу користувача завдяки

можливостям запису, використовуючи сценарії на таких мовах програмування як Python, JavaScript, VBScript тощо. Завдяки підтримці широкого спектру програм, таких як .Net, та гібридних додатків для iOS та Android, а також можливостей регресії, паралельного та крос-браузерного тестування, є можливість масштабувати тести на реальних тестових середовищах для повного охоплення. [3]

Jest - це платформа для тестування JavaScript, орієнтована на простоту. Цей фреймворк працює з проектами, що використовують: Babel, TypeScript, Node, React, Angular, Vue та інші. Jest розроблений для забезпечення коректності будь-якої кодової бази JavaScript. Це дозволяє писати тести за допомогою доступного, звичного та багатофункціонального API, який швидко дає результати. [4]

Testsigma - один із засобів автоматизованого тестування, доступний сьогодні, який найкраще підходить для сучасного ринку Agile та DevOps. Це інструмент автоматизації тестів, керований штучним інтелектом, який використовує просту англійську мову для автоматизації навіть складних тестів і відповідає потребам постійної доставки. Testsigma забезпечує екосистему автоматизації тестів з усіма елементами, необхідними для постійного тестування, і дозволяє автоматизувати веб, мобільні програми та служби API, а також підтримує тисячі комбінованих пристроїв / ОС / браузерів у хмарі. [5]

Висновки

Для автоматизованого тестування не існує універсального інструменту. Мови програмування і технології, які використовуються для розробки програмного забезпечення продовжують розвиватися так само, як інструменти автоматизованого тестування. На основі проаналізованих інструментів можна зауважити, що більшість засобів автоматизації тестування є платними, крім того, потрібні додаткові затрати праці на адаптацію для тестування. [6]. Тому ми можемо зробити висновок, що обраний Puppeteer є найбільш ефективним рішенням для автоматизованого тестування веб-додатків .

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ремінний О. А. Патерни автоматизованого функціонального тестування користувацьких інтерфейсів / О. А. Ремінний // Інформаційні технології та комп'ютерна інженерія. – 2013. – № 3. – С. 10-15.
2. Puppeteer: бібліотека Node.js для управління безголовим Chrome [Електронний ресурс] – Режим доступу: <http://helpex.ru/programmirovanie/puppeteer-biblioteka-node-js-dlja-upravlenija>, вільний.
3. Automated UI testing that covers you from device cloud to packaged apps [Електронний ресурс] – Режим доступу: <https://smartbear.com/product/testcomplete/overview/>, вільний.
4. Jest is a delightful JavaScript Testing Framework with a focus on simplicity. [Електронний ресурс] – Режим доступу: <https://jestjs.io/>, вільний.
5. Unified, cloud-based automation testing tool for DevOps . [Електронний ресурс] – Режим доступу: <https://testsigma.com/>, вільний.
6. Лайза Кріспін, Джанет Грегори. Гнучке тестування. Вільямс, 2010.- 251 с.

Катерина Арменівна Паламарчук — студентка групи ІАКІТ-176, факультет комп'ютерних систем та автоматизації, Вінницький національний технічний університет, Вінниця, e-mail: sinhrofazatronova@gmail.com

Науковий керівник: **Роман Наумович Квєтний** — завідувач кафедри АІТ, Вінницький національний технічний університет, м. Вінниця

Palamarchuk Kateryna A. — Faculty of computer systems and automation, Vinnytsia national technical University, Vinnytsia, e-mail: sinhrofazatronova@gmail.com

Supervisor: **Kvetny Roman N.** head of the Automation and intelligent information technology

department —, Vinnytsia National Technical University, Vinnytsia