

A Comparative Study of UI Testing Framework

Elis Pelivani
South East European University,
Tetovo, Macedonia
Contemporary Sciences and
Technologies
elis.pelivani@hotmail.com

Adrian Besimi
South East European University,
Tetovo, Macedonia
Contemporary Sciences and
Technologies
a.besimi@seeu.edu.mk

Betim Cico
Epoka University,
Tirana, Albania
Computer Engineering Department
bcico@epoka.edu.al

Abstract—Software testing is gaining importance and most of the applications and delivery services have a threshold of test coverage and successful tests. There are different testing frameworks that can be used, but it is important that the companies choose based on their testing requirements, abilities, and capabilities. In this paper, we assess most of commonly popular software testing frameworks, test frames, and define their strength and weakness, Selenium IDE, Cypress, WebDriverIO and Robot Framework. For the research, there will be a real-life software with real test data. Test cases and test scenarios will be selected in order to have a better parameters measure and to have a realistic outcome for both the above points. We will do the evaluation and the comparison of automation testing tools and frameworks used to test the application.

Keywords—component; testing, automation, tool, framework

I. INTRODUCTION

This study is part of a PhD research that aims to introduce a general framework for test and code optimization. The first research is published MECO conference in 2021, and its focus was to dive deep in the advantages and disadvantages of the most powerful software testing frameworks and tools [1]. In this study, we are focusing mainly on the implementation and real case results for a real application using the most powerful and used software testing frameworks.

An essential factor for the efficiency and quality of software testing process are test metrics. Some of the test metrics are:

- Organization metric refers to the utility of a metric in planning and management processes.
- Product metrics are related to a particular product, such as a compiler for a programming language.
- Procedure metrics, in contrary with product metrics, are related to a specific test process.
- Static metrics are those that can be calculated without requiring the product to be executed.
- Project metrics should be used to monitor and control a particular project.
- Dynamic metrics necessitate the execution of code.

By using methodological frameworks, we can determine testing procedures or tools that are used in many case studies.

Software test tools comes in aid of developers to examine bugs, verify the functionality and assurance of the dependability of developed software. These tools are classified based on their application types and then compared in their attributes that are prices and features. Software testing methods and tools are getting popular day by day, they reduce time on test analyzation and creation process, and also helps developer to improve the quality of their application. Contributions to testing research are automated test data generation and new frameworks that are used for test execution which lower the testing cycles.

TABLE 1. COMPARISON CRITERIA

Features	Selenium	Cypress	WebDriverIO	Robot Framework
Test development platform	Cross-platform	Cross-platform	Cross-platform	
Cost	Free	\$399/Mon	Free	Free
Scripting languages	Java	JavaScript	JavaScript	Keyword driven
Programming skills	Advanced skills needed	Advanced skills needed	Advanced skills needed	No programming skills
Learning curves	Long	Medium	Medium	Fast
Installation and use difficulties	Take some time	Set up and run are easy	Set up and run are easy	Very easy to set up and run
Script creation time	Time-consuming	Fast	Fast	Fast
Continuous integrations	Jenkins, GitLab, Docker, Azure	Jenkins, TeamCity, Travis, Docker	Jenkins, GitLab, Travis	Jenkins, Docker, TeamCity
Application under test	Web apps	Web apps	Web apps, mobile	Web apps, rpa

Many criteria are considered for software development such as, scripting tools, pricing, mobile and service support, languages, meanwhile for the test management and

development we can list, test workflow test coverage, and test development platform [2].

In this research, compared to the other researches we have emphasized the most important criteria from the perspective of industry experts. In Table 1, we have listed the features that have an impact in choosing the right testing framework.

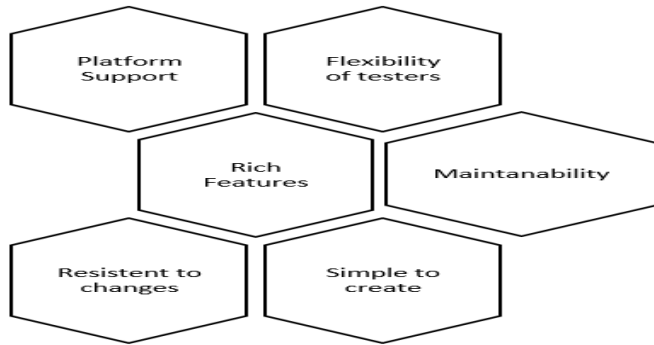


Figure 1. Key point to choose the best framework

In Figure 1, we have shown the most important features that should be considered before choosing a UI testing framework.

Table 2 shows the tools discussed and the requirements met by each tool. This analysis would allow business professionals to choose the best tool used to test software, whether large-scale or small-scale.

II. SIMULATIONS

A. Test frameworks and environment

Selenium is the most used automation-testing framework. The programming language used to develop Selenium is Java. It is integrated with Cucumber that is a framework that supports Behaviour Driven Development (BDD). This powerful combination allow us to create test cases specified in a logical language that customers can understand and at the same time very customized and maintainable implementation.

Cypress, because of its organization and architecture, is called as next generation Automation testing framework. This framework operates inside the browser. It is the browser, which is executing the test case. This framework is designed to get all data inside and outside the browser, which helps to deliver results that are more consistent. Furthermore, Cypress by providing native access to every object leads to fewer needs to work with object serialization or with over-the-wire protocols. Because of its architecture, Cypress is enabled to listen and modify how the browser act at run time by altering Network requests, responses on the fly and manipulating DOM. This framework is developing a new kind of testing with having control in both front and backend within the application. It supports only CSS selectors for catching the HTML elements. Framework as Mocha and Chai are used for building the test cases. Cypress is more oriented as developer-centric test automation framework comparing to the other tools used for testing by focusing its goal on making test-driven development (TDD) [3].

WebDriverIO is a framework based on selenium, which uses Node.js for its development. This framework can be easily setup by using its TestRunner client and run by using CI tools. As per the language used, is JavaScript, to run it is used Node, and provides a powerful framework for both mobile and web automation. Many plugins for the automation needs are provided by WebDriverIO framework [4].

Robot Framework is another framework used for test automation and is provided as an open-source automation framework. It can be integrated with virtually any other tool no matter what operating system is used to run the application. Robot Framework is designed as a modular architecture, which can be extended with bundled and self-made libraries. Selenium2Library is used in our case to implement the test cases. Keyword driven syntax makes it easy to implement the test cases, but the layered architectures slow a bit the execution [5, 6].

TABLE 2 THE SIMULATED TEST RESULTS

	Selenium	Cypress	WebDriIO	Robot Framework
System Model	HP EliteBook 850 g3	HP EliteBook 850 g3	HP EliteBook 850 g3	HP EliteBook 850 g3
RAM	16 GB	16 GB	16 GB	16 GB
Operating System	Windows 10 64-bit	Windows 10 64-bit	Windows 10 64-bit	Windows 10 64-bit
Tool version	3.141.59	6.3.0	6.12.1	3.2.2
Browser Used	Chrome 80.0	Chrome 80.0	Chrome 80.0	Chrome 80.0

B. Test cases and results

In this research, a set of test cases is developed and analyzed for all frameworks stated before, in order to compare them better and to come in a result.

No. of test cases = 10 and the number of steps ~ 120 steps.

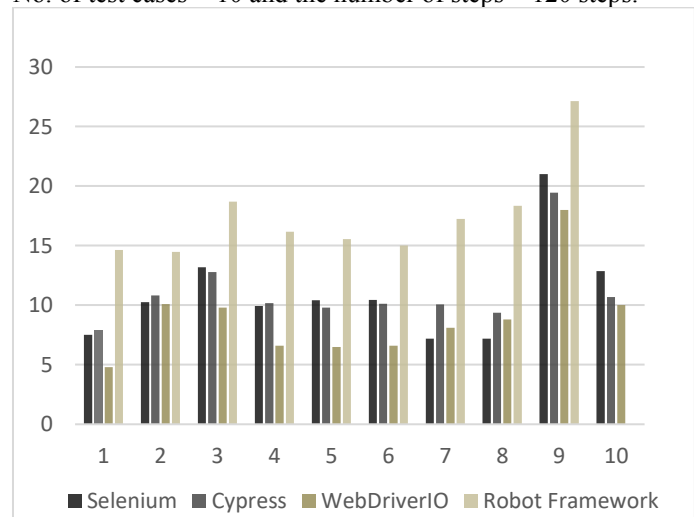


Figure 2. Average number of objects used on each tool

In this way, you can track everything that test case goes through. It makes finding the error easier. Cypress can show assumption of the failing reason. The Cypress Dashboard, Figure 8, is a service that makes it possible to record test results, normally when running Cypress tests from your CI provider [8].

```
[chrome 89.0.4389.90 windows #0-6] Spec: C:\Users\UI98447\Desktop\automation test\WebDriverIO\test\specs\onLeaveMode.js
[chrome 89.0.4389.90 windows #0-6] Running: chrome (v89.0.4389.90) on windows
[chrome 89.0.4389.90 windows #0-6] Session ID: bf2875c7a90b548614d2acc68652046a
[chrome 89.0.4389.90 windows #0-6]
[chrome 89.0.4389.90 windows #0-6] on leave mode
[chrome 89.0.4389.90 windows #0-6] ✓ on leave mode
[chrome 89.0.4389.90 windows #0-6]
[chrome 89.0.4389.90 windows #0-6] 1 passing (23.7s)
-----
[chrome 89.0.4389.90 windows #0-5] Spec: C:\Users\UI98447\Desktop\automation test\WebDriverIO\test\specs\inOfficeMode.js
[chrome 89.0.4389.90 windows #0-5] Running: chrome (v89.0.4389.90) on windows
[chrome 89.0.4389.90 windows #0-5] Session ID: bf2875c7a90b548614d2acc68652046a
[chrome 89.0.4389.90 windows #0-5]
[chrome 89.0.4389.90 windows #0-5] In office mode
[chrome 89.0.4389.90 windows #0-5] ✓ In office mode
[chrome 89.0.4389.90 windows #0-5]
[chrome 89.0.4389.90 windows #0-5] 1 passing (36.5s)
```

Figure 9. WebDriverIO Report

WebDriverIO's report is very specific for test execution time, which shows the milliseconds too. Robot Framework reporting service, generate a very detailed execution report, giving us the summary information, test statistics and test details, Figure 9. The report is a built-in feature and auto generated, therefore you cannot make any change in how information is displayed, Figure 10 [9, 10].

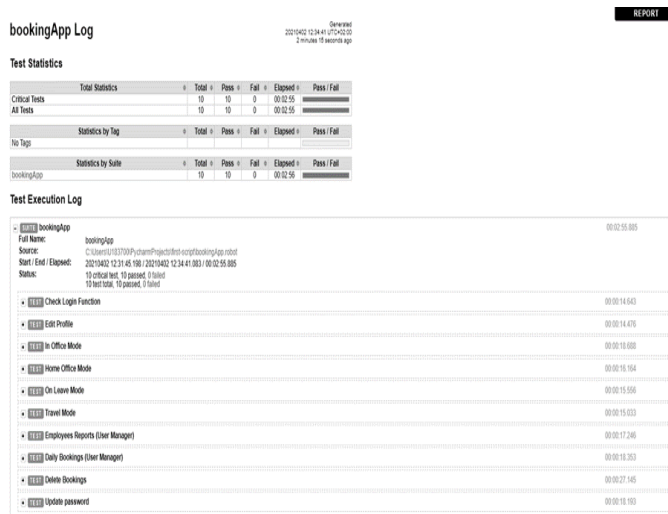


Figure 10. Robot framework Report

D. Discussion over our experiments

An interesting conclusion of the experiments and that is shown also in the figures, is the ration between the total numbers of the objects compared to the average number of objects used on each tool. Even though Selenium has the highest number of the objects, its average is similar to WebDriverIO, and smaller than the Robot Framework. Besides that, the total and average numbers of the steps used in Selenium, Cypress and WebDriverIO is significantly smaller than Robot Framework.

Although Selenium, Cypress, WebDriverIO, and Robot Framework are all frameworks designed for testing purposes, they differ considerably in terms of architecture and performance. Applications that have noncomplex front-end components can be used Selenium for test case creation and development. Because of its support for multiple languages makes Selenium a good choice as the test automation framework for development projects that are not in JavaScript. Cypress, it is natively tied into the web browser, so it integrates well with the client side and asynchronous design of these applications. Thus, test scripts run more reliably and much quicker than they would for the same application tested with Selenium for automation [11, 12]. On the other hand, Robot Framework offers possibilities to extend libraries implemented on Python or Java, and creating keywords, which can be an extension of the current keywords that are already implemented in framework by using a syntax that is equivalent with the one used for test case creation process [13, 14].

III. CONCLUSION

Software testing is an integral aspect of the software development by providing an error free product and without consequences. The main reason why software testers choose automated testing and automated tool is to reach the demand requirements and time factor. Furthermore, these automated tools give them the possibility to create better test cases, which are easy to maintain, experiment on different libraries, which can help on creating better scenarios, and minimize the expenses borne by the company during the research process. This research is based on four automated frameworks, Selenium, Cypress, WebDriverIO, and Robot Framework by highlighting their basic features and characteristics based on some scenarios, which are used for testing [15]. All these tools are effective to be used for testing, but depending on the situation, some may appear to be more efficient than others are. Furthermore, the previous paragraphs state all the specifics of some of the chosen testing frameworks and their test forms in tabular format, their licensing, the medium they work better, and the tools that help code reusability, among others. This analysis can be helpful for industry expertise to know the tool that is better use for a specific project and for analysts. In addition, more tools will be taken in consideration and more comparative requirements will be illustrated, making it easier for software testers to pick the perfect tools to assess apps with ease, saving more time and reducing costs while choosing the right testing tool [16, 17]. We may also assume that there is no ideal testing method, but for a specific testing reason, trade-offs can be made to choose the right tool, based on the scale of the project, the budgeted cost of testing, the implementation platform and even the language used to build the project.

REFERENCES

- [1] E. Pelivani and B. Cico, "A comparative study of automation testing tools for web applications," 2021 10th Mediterranean Conference on

- Embedded Computing (MECO), 2021, pp. 2-7, doi: 10.1109/MECO52532.2021.9460242.
- [2] Harpreet Kaur, Gagn Gupta. Comparative Study of Automated Testing Tools: Selenium, Quick Test Professional and Testcomplete. International Journal of Engineering Research and Applications, ISSN: 2248-9622, Vol. 4, No. 4, 09-10, 2013, pp.1740-1744.
- [3] Winkler, Dietmar & Hametner, Reinhard & Östreicher, Thomas & Biffl, Stefan. (2010). A framework for automated testing of automation systems. IEEE Conference on Emerging Technologies and Factory Automation (ETFA). 1 - 5. 10.1109/ETFA.2010.5641264.
- [4] Kim, Eun & Na, Jong & Ryoo, Seok. (2009). Implementing an Effective Test Automation Framework. 2. 537-541. 10.1109/COMPSAC.2009.188.
- [5] D. Zun, T. Qi and L. Chen, "Research on automated testing framework for multi-platform mobile applications," 2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS), October 2016, pp. 85-89, doi: 10.1109/CCIS.2016.7790229.
- [6] Guo, Shuang Zhou. "Research on Automatic Testing Framework." Applied Mechanics and Materials, vol. 635–637, Trans Tech Publications, Ltd., 09. 2014, pp. 1597–1601. Crossref, doi:10.4028/www.scientific.net/amm.635-637.1594.
- [7] Monier, Mohamed & El-mahdy, Mahmoud. Evaluation of automated web testing tools. International Journal of Computer Applications Technology and Research. 4. 407-411, October 2015.
- [8] Daniel Asfaw, Benefits of Automated Testing Over Manual Testing, International Journal of Innovative Research in information Security, ISSN: 2349-7017, Vol.2, November 2015, pp 7-16.
- [9] Medhat Kamal, M., Saad M. Darwish, and Ahmed Elfatraty. 2019. 'Enhancing the Automation of GUI Testing'. In ICSIE 2019, Cairo, 66–70.
- [10] Mobaraya, Fatini & Ali, Shahid. (2019). Technical Analysis of Selenium and Cypress as Functional Automation Framework for Modern Web Application Testing. 27-46. 10.5121/csit.2019.91803.
- [11] Selenium umbrella, Selenium Documentation: Test Automation for Web Applications, www.Seleniumhq.org, last update 11 November 2021.
- [12] Cypress team, CypressIO, <https://www.cypress.io/>, last update 2021
- [13] Satish Gojare, Rahul Joshi, Dhanashree Gaigaware, Analysis and Design of Selenium WebDriver Automation Testing Framework, Procedia Computer Science, Volume 50, 2015, Pages 341-346, ISSN 1877-0509,
- [14] Chandraprabha, Chandraprabha & Kumar, Ajeet & Saxena, Sajal. (2015). Data Driven Testing Framework using Selenium WebDriver. International Journal of Computer Applications. 119. 18-23. 10.5120/20845-3497.
- [15] Vila, Elior & Nedeltcheva, Galia & Todorova, Diana. (2017). Automation Testing Framework for Web Applications with Selenium WebDriver: Opportunities and Threats. 145-150. 10.1145/3133264.3133300.
- [16] Stresnjak, Stanislav & Hocenski, Zeljko. (2011). Usage of Robot Framework in Automation of Functional Test Regression.
- [17] Jyoti Shetty, Neha S Batni , "A Comprehensive Study on Automation using Robot Framework", International Journal of Science and Research (IJSR), https://www.ijsr.net/search_index_results_paperid.php?id=SR20710144623, Volume 9 Issue 7, July 2020, 1034 - 1037