https://dx.doi.org/10.22161/ijaers.74.15 *ISSN: 2349-6495(P) | 2456-1908(O)*

Selenium as a Free Tool to Test for Java Web Application

João Paulo Renovato dos Santos[1], Katia Petrolina da Silva[2], Bruno Pereira Gonçalves[3], Jaqueline Silva de Souza Pinheiro[4], Jean Mark Lobo de Oliveira[5], David Barbosa de Alencar[6]

[1,2,3,4,5]Academic department, University Center FAMETRO, Manaus-AM, Brazil
[6]Research department, Institute of Technology and Education Galileo of Amazon – ITEGAM, Manaus-AM, Brazil

Abstract**—** *The objective of this article was to present the Selenium tool in conjunction with the Java Web program language and the need to use automated tests to ensure a system (software) of quality and reliability, thereby preventing occasional failures, extra expenses, loss of reputation commercial, among other annoying. Nowadays it is still very common for companies to carry out the testing of their systems manually and this can cause numerous problems for companies and customers, who may come to deposit personal information on the website, especially e-commerce systems that in addition of personal information, they also tend to take advantage of addresses and the various forms of online financial transactions. The tests performed in an automated way are faster than manual tests and this tends to decrease the delivery delay of the projects, besides making it easier for the tests to be carried out during the development of the system, thereby reducing the problems that may occur before version which will be made available to the customer or end user.*

Keywords**— Selenium; Java Web; Software; Tests; Automated Tests; Manual Tests.**

I. INTRODUCTION

Software is part of our routine, basically everything today needs a system, from a simple algorithm to a complex system. With all this growth, the user tends to be more careful with the software used. If a system or app lacks quality, it can generate financial losses, time and can even hurt the business reputation. They need to provide the correct data, maintaining its stability, robustness and usability. This becomes complicated when companies need to create complex and interconnected systems in a short time and at a low cost. The tests often do not occur in the correct way, which can cause failures, bugs in the production environment, or even fail to meet the expectations of stakeholders in the project. Thus, bringing a long delay in the final delivery of the project.

There are several mistakes in software development, one of which is just to run the software and look for errors, although it is not only that, it involves other activities such as: planning, analysis, modeling and implementation in a test environment. Following the strategy of being involved since the beginning of the project and development and of knowing the right moment to carry out a manual or automated test, there will be several benefits in terms of saving time and money, regarding gains in the quality of the project.

If a user accesses a shopping website and it is not secure, the customer will be frustrated and may withdraw from the purchase or buy from a competitor. The solution is to show the steps for the developer to follow and produce the system that meets the design quality standards and user requirements. Making the site fluid, without errors or flaws and making a site safe and easy to make purchases..

II. BIBLIOGRAPHIC REFERENCE

2.1 Java market on the web

With regard to the current market, there is no doubt that Java WEB is the one that contains the greatest possibilities of performance. And it is not for nothing that Java has this fame, the projects developed in it bring greater independence, thereby avoiding the need for another supplier, that is, vendor lock-in. Better said, the company becomes more autonomous from other software, such as: the database, the developer of its virtual machine (Virtual Machine) and the servlet container. In addition, Java

https://dx.doi.org/10.22161/ijaers.74.15 ISSN: 2349-6495(P) | 2456-1908(O)

allows development to be done on one operating system and implementation (deploy) to be performed on another operating system.

Even though it is popular in the WEB area, for the development of Java software it is also essential to have a certain knowledge of servlet APIs and JSP, regardless of whether the development team wants to make use of: VRaptor, JSF, Struts, in addition to other Frameworks. In addition, in order to understand the problems that may occur in your application, knowledge of session, cookies and HTTP is essential.

2.2 Software tests

Software tests are methods associated with the development of a project (software), which has the function of locating errors in the software, reporting errors and analyzing whether they have already been solved, with that the final product will have a higher quality.

2.2.1 The importance

The importance of software testing is in preserving the quality of the software. Ensuring that the software has all the requirements requested by the customer. However, quality control is only one of a number of areas that software testing covers.

They are utilities that help to reduce financial expenses and bring greater reliability to customers, so it is common for companies to use software testing to safeguard reputation, as companies that develop low quality systems tend to be frowned upon in the Marketplace. With software testing, future expenses can be avoided by analyzing Croatian errors in the system. Because when a defect is found in the test phase, it can be up to 100 times cheaper to correct compared to the same error in the production phase.

2.3 Selenium

Selenium is a framework made available for free that focuses on testing web applications by the browser (browser) automatically, compressing tests for web purposes and its compliance with browsers allows it to work on different systems. It operates with a kit of resources and libraries used in the automation of browsers for the purpose of administrative repetition actions on websites and for testing actions. The set of tools that Selenium makes it possible to follow numerous tests focusing on various types of utilities in WEB applications, in this way versatile actions such as the location of elements of the customer interface and the comparison of these elements with the expected results. possible to have real conjectures of the application procedures. Selenium has support for other programming languages besides Java, such as: CSharp, JavaScript, Php, Perl, Python and Ruby. In addition to supporting your actions for different browsers on different platforms.

III. MATERIALS AND METHODS

In this article we use the Selenium test tool for Java Web, to do the acceptance tests of the application. In each update launched of a web application there is a range of features that need to be tested numerous times. These tests take a lot of time and generate a lot of rework. That is why it is necessary to use a testing tool like Selenium, to streamline the testing process.

Fig. 1: Selenium logo.

Source: Selenium dev official website

Selenium is a free and open source tool that performs automated testing of an application. No longer has necessary to perform repetitive tested done by humans, this exhaustive work will be done by Selenium.

Fig. 2: Process of using Seleium with the Navigator.

Source: own authorship.

Selenium simulates user behavior using a browser. When we use it, we need two tools: Selenium IDE and Selenium WebDriver.

The Selenium IDE allows a quick creation of test scripts, it allows to record the user's behavior in the application, we can know the page accessed, texts typed in the inputs, click on buttons and etc.

Selenium WebDriver provides the API to perform tests with greater productivity in an agile way. This is the natural choice when we want to do automated tests for web development.

It makes calls directly in the browser using each browser's native engine. That is why the tests with it are quite realistic, because instead of using a JavaScript engine itself, it is used by the browser itself. It supports most browsers on the market, such as: Opera, Mozilla, Chrome, Safari, among others.

https://dx.doi.org/10.22161/ijaers.74.15 *ISSN: 2349-6495(P) | 2456-1908(O)*

Use Selenium Web Driver

We will see use in a Java Web application.

3.1 Initial setting

To perform tests on Selenium, first of all it is necessary to add the dependency of the Selenium Web Driver to your application. Using Maven, which is a dependency manager most used in Java applications, just add the artifact to your pom.xm. As in the image below:

```
<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>3.4.0</version>
    <scope>test</scope>
</dependency>
```

Fig. 3: Added dependency on the application using Maven.

Source: Authors, 2020.

By placing this library in your project, Selenium is ready for use. There are several implementations of interfaces for use, we can use one of them for our applications and start the tests.

A common use of Selenium is to fill out a form and submit it on a web page. Let's take an example to understand it better.

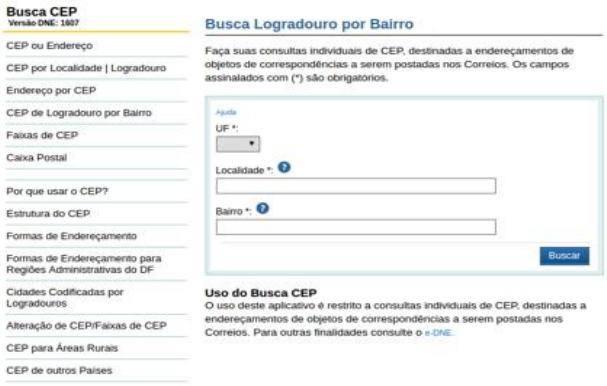

Fig. 4: Form where the test will be performed.

Source: post office website.

We will need to fill three fields, the UF, Localities and Bairro. After that click on the search button.

3.2 How to fill data with Selenium

First of all, we need to know the HTML codes, because we are conducting tests for WEB. And before getting to know Selenium, you have to have a web development base, because we need to know the TAGS of a web document.

We want to fill a select, our UF. Two text fields, which are our Locality and Neighborhood inputs. After that click on submit to submit the form.

UF: tag select (creates a combo with options) with the name = "UF" attribute

Locality: input tag with name = "Locality attribute

Neighborhood: input tag with name = "Neighborhood" attribute

"Search" button: input tag with the attribute type = "Submit" (Creates a button to submit the form).

3.3 We will perform the following steps:

1. Enter the post office page; 2. Fill in the fields; 3. Click the search button; 4. View the search result

```
@Test
public void preencheFormularioCorreiosBuscaLogradouroPorBairro() {
  WebDriver driver = new ChromeDriver();
  // Visita a página do Correios
  driver.get("http://www.buscacep.correios.com.br/sistemas/buscacep/buscaLogBairro.cfm");

  // Escolhe o valor de UF
  Select selectUF = new Select(driver.findElement(By.name("UF")));
  selectUF.selectByVisibleText("RJ");
  // Preenche a Localidade com o valor "Rio de Janeiro"
  WebElement inputLocalidade = driver.findElement(By.name("Localidade"));
  inputLocalidade.sendKeys("Rio de Janeiro");
  // Preenche o campo Bairro com o valor "Copacabana"
  WebElement inputBairro = driver.findElement(By.name("Bairro"));
  inputBairro.sendKeys("Copacabana");

  // clica no botão Buscar
  WebElement buttonBuscar = driver.findElement(By.cssSelector("input[type='submit'"));
  buttonBuscar.click();
}
```

Fig. 5: Test script to fill out fields on the post office website.

Source: The Authors, 2020.

When running the test, the Browser is automatically opened, the fields are filled and the search is performed.

With automated tests it is not necessary to keep filling fields and submitting forms manually. Just create an automated script to do all of that. The test case we performed was a basic test. But more advanced scripts can be created, to perform performance tests and systems invasion.

Fig. 6: Selenium test cycle.

Source: selenium. dev official website.

This is the test cycle of Selenium; it is possible to perform complete tests of your application. It is not just used to test interfaces.

https://dx.doi.org/10.22161/ijaers.74.15 *ISSN: 2349-6495(P) | 2456-1908(O)*

Tests of business rule, usability, database, system functionalities etc. can be performed.

With it is possible to carry out tests on applications written in other programming languages, in addition to Java. It is a complete tool for performing automated software testing.

IV. RESULTS AND DISCUSSION

An exploratory research was carried out to identify what users think of the software they work with on a daily basis, and together with this research, we visited the UEA development pole, where we interviewed some developers about the tests carried out at the pole of its developed programs.

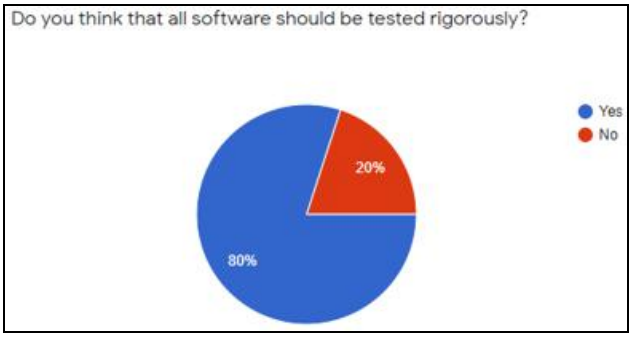

Fig. 7: There is no Software tester.

Source: The Authors, 2020.

We see that 80% of companies do not have a software tester. This means that many companies do not bother to test their company.

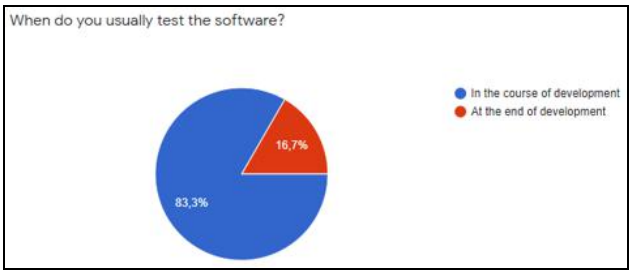

Fig. 8: Software tested in Development.

Source: The Authors, 2020.

We see that 83.3% test the software in the course of development. This generates tests during the entire development process, but it is necessary to make a final test to evaluate the possible errors that may be present in the software.

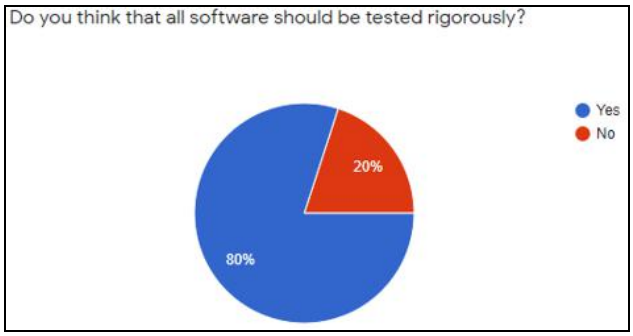

Fig. 9: Tested software.

Source: The Authors, 2020.

We see that 80% think that the software must be tested rigorously. Tests are essential to ensure the quality of the software, and to ensure that the user does not suffer from possible system failures.

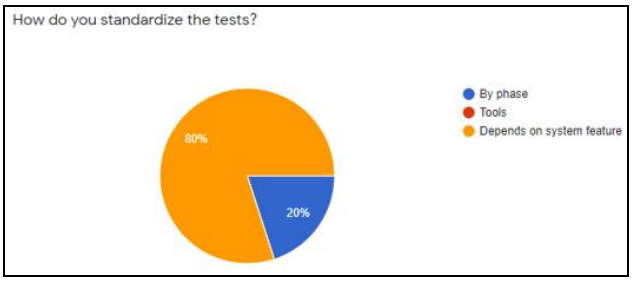

Fig. 10: Software standardization.

Source: The Authors, 2020.

We see that 80% standardize the tests depending on the characteristic of the system. Depending on the system the test will be done in a way, you need to evaluate the language, the framework you are using to perform your tests.

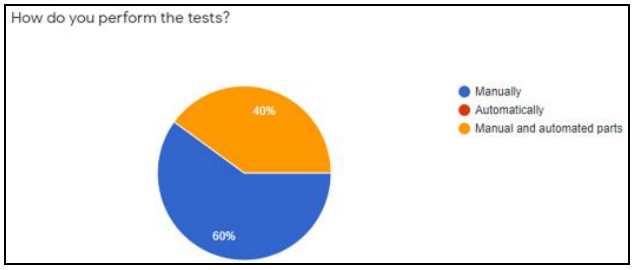

Fig. 11: How the Software test is done.

Source: The Authors, 2020.

We see that 60% perform the tests manually. This is the old way of doing tests, today the tests are automated, avoiding the repetitive work of the tester, who can miss countless failures.

https://dx.doi.org/10.22161/ijaers.74.15 *ISSN: 2349-6495(P) | 2456-1908(O)*

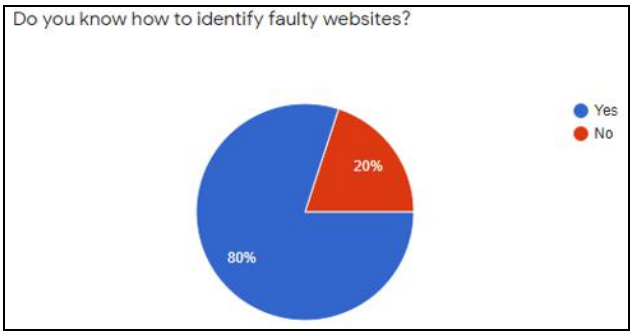

Fig. 12: Identifying faults.

Source: The Authors, 2020.

Most users of the survey, accounting for 80%, said they can identify flaws in the websites they access.

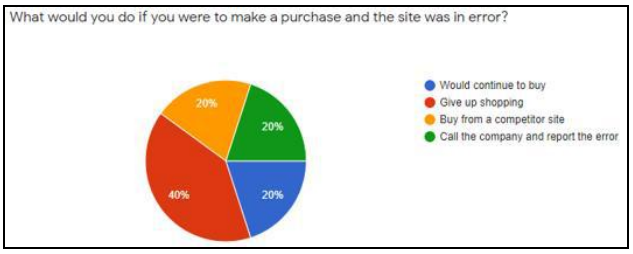

Fig. 13: Software purchase.

Source: The Authors, 2020.

If there was a failure on the site 40% of users answered that they would buy on a competitor site. This is worrying for those who have an e-commerce, with possible failures will result in the loss of future customers.

V. CONCLUSION

It can be concluded from this article that there is an increase in the quality of software, due to the users who are becoming more careful. The lack of quality and stability of a system can result in great losses. With this, the need arises to guarantee that the final user experience is satisfactory and for that, it is necessary tools that can cover the project.

Java for Web is one of the best options for the current market, as its structure allows the system not to depend on third parties, thereby avoiding the vendor lock-in, however the entire project is, regardless of the facilities that the programming language offers, is subject to failures that affect the final product. With that said tools that can identify these errors are also necessary for the quality of the software, in this situation comes the Selenium framework. The saying whose objective is both tests and repetition tasks on sites with support for different platforms.

ACKNOWLEDGEMENTS

I thank God first for keeping me on the right track during this research project with health and strength to reach the end.

We are grateful to our family for the support they have always given us throughout our lives.

We would like to thank our advisor for the encouragement and dedication of his limited time to my research project.

We also thank the University Center FAMETRO and all the teachers of the course for the high quality of the education offered.

REFERENCES

[1] ABINADER, Jorge Abílio; LINS, Rafael Duarte. Web Services em Java. 1. ed. Rio de Janeiro: Brasport, 2006.

[2] BEIZER, B.; Software Testing Techniques, 2. Ed. New York: Van Nostrand Eeinhold, 1990.

[3] CARVALHO, Lucas G. Introdução ao Selenium WebDriver. EDITORA GLOBO. 2018. Disponível em: <https://medium.com/editora-globo/introdução-ao-selenium-webdriver-8ac2f9a8d9d0>.Acesso em: 28 mar. 2020.

[4] ENFRENTANDO O JAVA NA WEB. CAELUM. Disponível em: <https://www.caelum.com.br/apostila-java-web/enfrentando-o-java-na-web/>. Acesso em: 28 jan. 2020.

[5] FERRAMENTAS QUALIDADE DE SOFTWARE SELENIUM TESTES TESTES DE ACEITAÇÃO. CODE A TEST. Disponível em: <http://www.codeatest.com/testes-aceitacao-automatizados-selenium/>. Acesso em: 12 fev. 2020.

[6] MALDONADO, José Carlos; VINCENZI, Auri Marcelo Rizzo; BARBOSA, Ellen Francine; SOUZA, Simone do Rocio Senger de; DELAMARO, Márcio Eduardo. Aspectos teóricos e empíricos de teste de cobertura de software. Curitiba: Champagnat, 1998.

[7] NETO, Arilo Cláudio Dias. Introdução a Teste de Software. Engenharia de Software Magazine: v.90, p. 54 -59, 2007.

[8] NOURIE, Dana. Java Technologies for Web Applications. ORACLE. 2006. Disponível em: <https://www.oracle.com/technical-resources/articles/javase/webapps-1.html>. Acesso em: 20 jan. 2020.

[9] PRESSMAN, Roger S. Software Engineering: A Practitioner's Approach. 4. ed. New York: McGraw-Hil, 1997.

[10] ROCHA, A.R.C.; MALDONADO, J.C.; WEBER, K.C. Qualidade de software: teoria e prática. São Paulo: Prenttice Hall, 2001.

[11] VIEGAS, Júlio. TESTE DE SOFTWARE: INTRODUÇÃO, CONCEITOS BÁSICOS E TIPOS DE TESTES. ONEDAYTESTING. Disponível em: <https://blog.onedaytesting.com.br/teste-de-software/>. Acesso em: 8 mar. 2020.