International Journal for Research in Engineering Application & Management (IJREAM)
ISSN : 2454-9150 Vol-04, Issue-12, Mar 2019

# SELENIUM: An Automated Software Testing Tool

[*]Ms. Megha Desai, [#]Prof. Uttara Gogate

[*]PG Scholar, Alamuri Ratnamala Institute of Engineering & Technology, Mumbai, India.

meghadesai.8@gmail.com

[#]Associate Professor, Shivajirao S. Jondhale College of Engineering, Mumbai, India.

uttara.gogate16@gmail.com

**Abstract: Software Development Life Cycle ( SDLC) is a systematic way of developing quality product or application and Software Testing is one of the most important phase in SDLC. The objective of the testing process is to compare the expected result with actual result or to find out the variance between expected result and actual result of an application. The work of tester simplify by automating the execution of test scripts Automation tools. These days, the number of software system has been implemented as web applications. These web applications are very complex and very difficult to test manually. Hence, the automation testing is used. Automation testing reduces the human involvement and repeatable tasks with the help of automation tools. This paper focuses on the Selenium automation tool which is used to test web applications by executing test scripts.**

***Keywords — Automation testing, Selenium Grid, Selenium IDE, Selenium WebDriver, Software Testing, Web applications***

## I. INTRODUCTION

Software testing process is used to evaluate all the functionalities of software application to find out either the developed software application met the specified client requirement or not and to find the defects to make sure that the application is defect free to produce the good quality product. Two types of testing: 1. Manual Testing and 2. Automation Testing.

1. Manual testing finds the defects or bugs in a software application. The variance between expected and actual results is called as defects or bugs. In manual testing, tester executes all the test cases manually without utilizing any automation tool. Test lead prepares a test plan document which describes how testing team is moving to accomplish their task. Test plan document is the detailed and systematic way to test software application and verifies that all the functionalities along with functional and non-functional requirements of the application are running correctly. Manual testing is time consuming because it involves complete test cases.

2. Automation testing is the process of testing the application utilizing an automated tool to find out the defects. In this process of testing, testers execute the test script and then generate the test results automatically with the help of specific automation tool. When some existing bug fixes or some new functionality are added, at that time to manually test all the functionalities of the application every time is very difficult task for the testers. Thus, it's better to test the application every time using Automation tool is very efficient and effective. Automation testing is effective in terms of resources, cost and time.

These days, many software testing tools are accessible in the market for testing purpose. These software testing tools are categorized as Test management tools like TET, Functional testing tools like Test complete and Selenium, Load testing tools like Jmeter. Test Environment Toolkit (TET) is an open-source and unsupported command-line product which is extensively used in various test applications. TestComplete offer functional automation testing tool. Using TestComplete, testers can create automated tests for operating systems, Web and iOS applications. Test can be recorded, manually created or scripted and used for automated playback and error logging with the help of TestComplete. Apache JMeter is used for performance testing and web dynamic applications testing on dynamic and static resources as it reproduce a massive load on server to check the server strength or to examine the overall execution of the application under various load types.

Selenium is set of libraries and tools that automate web browser actions. Selenium produce tools which interact with the browser and automate the browser actions like navigate, select, input, click, etc using scripts. Selenium is not a tool but library of tools. It is open-source and free. It is extensively used open-source tool for test automation of web application.

 DOI : 10.18231/2454-9150.2019.0162 

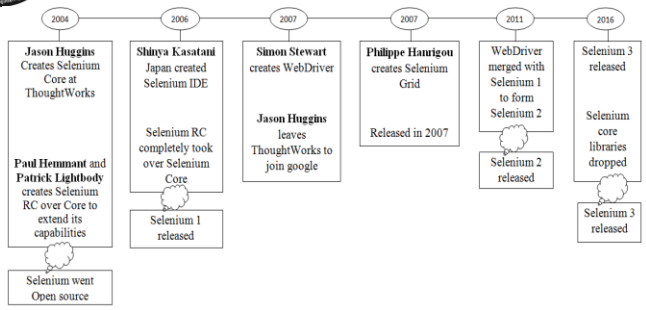


**Figure 1. History of Selenium**

In 2004, Jason Huggins created Selenium core. Selenium was made open source and released at the end of 2004. In 2006, the Selenium RC completely took over Selenium Core & released as Selenium 1. In 2006, Shinya Kasatani created Selenium IDE. In 2007, Simon Stewart created Selenium Webdriver. In 2007, Philippe Hanrigou created Selenium Grid. In 2011, the two tools merged to form Selenium 2.0 and were released on July, 2011. Selenium 3 released on October, 2016.

**Example: Login Page Testing**

When you want to login at that time first you enter Username and the Password and then you click login button. For automate testing of this login page first we download Selenium library ⟨any programming language⟩ and then we create test scripts. Finally we call that test scripts and test the given login page. The steps to be followed:

1. Identify web elements ⟨using identifiers like id, path⟩
2. Add actions ⟨using preferred programming language⟩
3. Run test and validate.

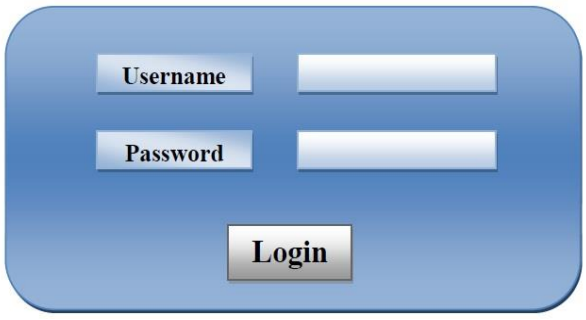


**Figure 2. Login Page**

During the time Selenium was being created, there was another popular web testing tool QTP developed by Mercury Interactive. In one of the email Jason Huggins jokingly said, "You need Selenium supplements to cure Mercury poisoning". The name caught on from there. Both Selenium and Mercury are chemical elements.

## II. LITERATURE SURVEY

To make sure the success of any testing project the identification of the right automation tool is very critical. Before selecting any particular automation tool the detailed analysis of that tool must be conducted. The efforts put in the tool's evaluation process helps the successful execution of the project. Testing tool selection is dependent on application and its technology stack, available skill sets or abilities in the organization, the detailed testing requirements and lastly the license cost of the tool. Now days, different automation tools are accessible in the market for automating both desktop and web applications. But Selenium is most accepted automated testing tool as compare to other automated tools. Some of these automated tools are:

### *A. HP Quick Test Pro (QTP) / HP Unified Functional Testing (UFT):*

UFT is also familiar as Quick Test Professional (QTP). It is used to test the various applications. UFT has its own Integrated Development Environment (IDE). Hence, tester can write the scripts efficiently. UFT is supported by Windows only. It uses VBScript Language for the programming. UFT is a functional automation testing tool. It is best suited for regression testing application.i.e.to performs impact analysis. [1]

**Limitations over Selenium:**

1. UFT costs more for licensing and maintenance, but Selenium is a totally free, open source tool and can be downloaded easily.
2. Selenium testing to be takes out in some platform but UFT can be used only on windows platform.
3. Browser support of Selenium is very high as compare to UFT. UFT supports mainly IE, Firefox, and Chrome browsers. Selenium supports Firefox, chrome, IE, Safari, etc.
4. UFT was designed to test one application at a time on a single machine where Selenium can run code on one machine and test the application on remote machine.

### *B. TestComplete:*

TestComplete is functional automation testing tool and it is developed by smartbear which records the tester who is performing manual test and also allows tester to play back and also maintained as an automated tests. TestComplete provides testers the capability to create an automated test for the applications like Web, Microsoft Windows, iOS and Android. TestComplete is utilized to produce and automate various types of test. [2]

**Limitations over Selenium:**

1. TestComplete costs more for licensing and maintenance, but Selenium is a totally free, open source tool and can be downloaded easily.

 DOI : 10.18231/2454-9150.2019.0162 

2. Selenium testing to be takes out in any platform like Linux, Mac, Windows but TestComplete can be used only on Windows platform.

### *C. SoapUI:*

SoapUI is an API testing tool. It is free, open-source and cross-platform. SoapUI perform both functional and non-functional testing. Non-functional testing like security and performance test is performed by SoapUI. It permits to rapidly and easily create and execute an automated regression test, functional test and load tests. [2]

**Limitations over Selenium:**

1. Selenium testing to be carried out in any platform like Mac, Windows, Linux but SoapUI can be used only on Windows platform.

2. In SoapUI, programming language used to edit testing scripts or for the creation of testing scripts are Groovy or JavaScript only where Selenium tool used Java, C#, Ruby, Python, PHP, JavaScript as Script-language.

## III. SELENIUM TOOL

Selenium tool is a group of various software tools and each software tool gives different perspective to support the test automation. Quality Analyst team members focus on one or two testing tools which meet the needs of their project. The help for performing one's tests on different browser platforms is the key feature of Selenium. Selenium tool is open-source and licensing cost is not involved. This is major advantages of Selenium tool over the other testing tools. Selenium is categorized into four components:

### A. Selenium IDE:

Selenium Integrated Development Environment i.e. Selenium IDE is prototyping tool use for creating test scripts. Selenium IDE records the user actions. These user actions are executed and then exports as a re-usable script in multiple programming languages and those re-usable scripts can be executed later. Selenium IDE is a Firefox and Chrome plug-in and it also provides an easy-to-use interface to the end user.

### B. Selenium RC:

Selenium RC is also called as Selenium 1. Selenium RC was main Selenium project before the Selenium WebDriver. In maintenance mode, still Selenium 1 is actively supported and provide some features which may not be present in Selenium 2 like support for some languages like Java, Javascript, Ruby, PHP, Python, Perl and C# and support for almost every browser. It is used to execute scripts which are written in any languages using javascript. Selenium 1 find unacceptable now and not actively supported.

Selenium RC components are:

- Selenium RC Server launches and terminates browsers, execute the Selenese commands. These commands are proceed from test program. Selenium RC server serves as HTTP proxy, also interrupting and confirming HTTP messages. These HTTP messages are passed between browser and AUT i.e. Application under Test.
- Selenium Client libraries provides the interface between Selenium RC Server and each programming language.

### C. WebDriver:

Combination of WebDriver API is one of the new features in Selenium 2. To support the dynamic web pages of an application, the Selenium WebDriver tool was developed. In dynamic web pages, content of the page may alter without reloading the web page. API is used to send commands directly to the browser. Selenium WebDriver gives programming interface to direct some restrictions in Selenium-RC API. It is a successor to the Selenium RC. Selenium RC and Selenium WebDriver are merged to create Selenium 2.

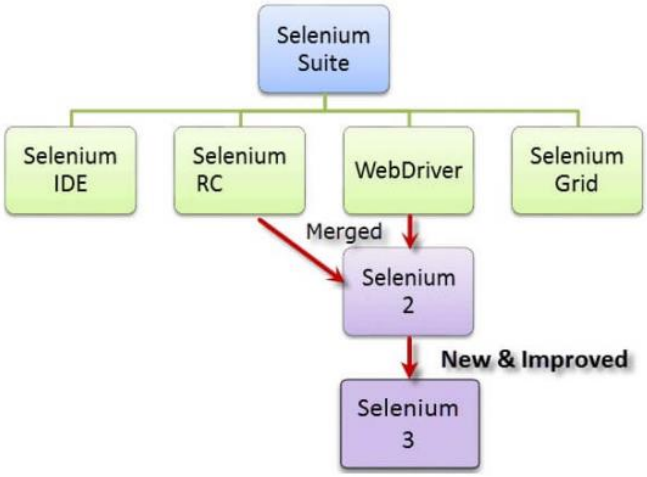


**Figure 3. Components of Selenium**

Selenium WebDriver produce direct calls to web browser using each browser's support for the automation. When the web browser was loaded, Selenium RC 'injected' the javascript functions into web browser and used javascript to handle an Application under Test. Instead of using this technique Selenium WebDriver manage the browser directly using the browser's built-in support for the automation.

### D. Selenium Grid:

Using Selenium Grid, we can run multi-tests at the very same time on different computer system which are using different web browsers and the different operating systems. Selenium Grid holds up distributed test executions and also it permit for executing the tests in distributed test execution background. Selenium Grid speeds up the test execution. Selenium Grid can be time saver for the long executing test suite and the large test suites. These test suites perform large amount of data validation.

For example, there is network of virtual machines and each

 DOI : 10.18231/2454-9150.2019.0162 

virtual machine supporting a various web browser that an application to be tested must support. Machine 1 has Internet Explorer 8, machine 2 has Internet Explorer 9, machine 3 has latest Chrome and machine 4 has the latest Firefox. When the test suite runs, the Selenium Grid receives each test-browser combination and assigns each test to run against its required browser. Hence, it is very flexible.

## IV. ARCHITECTURE

The following figure shows the architecture of Selenium web tool which consists of two basic components:

1. Selenium Server and

2. Selenium Client.

Selenium Server consists of:

1. Server Component

2. WebDriver API

3. Selenium Grid

Server Component: Selenium server consists of a server component. Server component is used to accept requests from Selenium Client's Remote Web Driver class.

WebDriver API: Selenium server is also consists of Web Driver API. WebDriver API is used to execute tests for the browsers on a server machine.

Selenium Grid: The last component of Selenium server is the Selenium Grid. Selenium Grid is executed by the Selenium Server in the command-line for grid characteristics which consist of nodes and the central hub for many situations and preferred browser capabilities. Selenium Grid tool runs the parallel tests through different machines and web browsers at same time which results the reduced execution time.

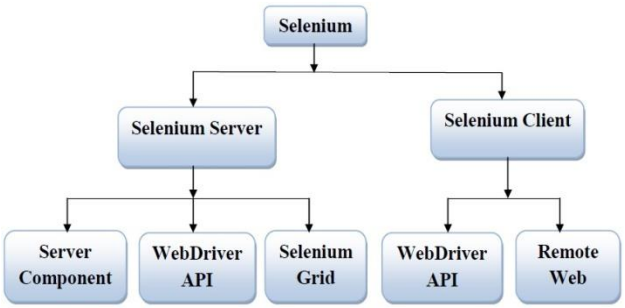


**Figure 4. Architecture of Selenium [3]**

Selenium Client consists of:

1. WebDriver API

2. Remote Web Driver

WebDriver API: The Selenium client consists of WebDriver API. WebDriver API is used to create test scripts to build an interaction with web pages and other application elements.

Remote Web Driver: It also includes the remote Web Driver class. Remote Web Driver class is used to communicate with remote Selenium server. [3]

## V. COMPARISON MATRIX

The given table shows comparison between the various tools based on the key features of software automation.

**Table 1. Comparison between various automation testing tools**

| Comparison Matrix | | | |
|---|---|---|---|
| **Selenium** | **HP QTP (UFT)** | **TestComplete** | **SoapUI** |
| **License cost** | | | |
| Its an open source testing tool hence it does not require license and is free | It is commercial tool by Micro Focus and hence it requires a license and is expensive | License and maintenance fees | Open source, Commercial licensed version available |
| **Supported programming languages** | | | |
| Java, C #, Ruby, Python, Perl, PHP and JavaScript | VB Script | JavaScript, Python, VBScript, JScript, Delphi, C++, and C # | JavaScript or Groovy |
| **Cross Platform** | | | |
| Selenium supports more number of OS like Windows, Linux, Unix and Mac. | QTP supports only Windows. | Windows 7 and later | Window XP and later |
| **Supported Browsers** | | | |
| Google Chrome, Mozilla Firefox, Internet Explorer, Edge, Opera, Safari, etc. | Specific versions of Google Chrome, Mozilla Firefox and Internet Explorer. | IE, Firefox, Opera, Chrome | IE, Firefox, Chrome |
| **Test Execution** | | | |
| execute tests in parallel. | perform one test per machine | perform parallel testing with TestComplete | Executed either in sequence or parallel |
| **Product Support** | | | |
| Open Source Community | Dedicate HP support along with support forums | Smartbear support with support forums | Smartbear support with support forums |

 DOI : 10.18231/2454-9150.2019.0162 

## VI. PERFORMANCE ANALYSIS

Various automated tools are available in the market for testing purpose. These automated testing tools can be used in the different areas of testing. Selection of the tool depends on the type of application that we want to test like UI testing tools, web testing tools, performance testing tools, etc.

Table 2. Performance Analysis of various automation tools

| Sr No. | Tool | Testing |
|---|---|---|
| 1 | Mercury Interactive WinRunner | Regression testing |
| 2 | Watir | Automating web browsers |
| 3 | Soap UI | Web service Integration Testing |
| 4 | Seque Software's Silk Test | Regression testing |
| 5 | Sahi | Web Applications |
| 6 | QTP | Functional/ regression Testing |
| 7 | IBM Rational SQA Robot | Functional Testing |
| 8 | Apache's JMeter | Performance and load testing |
| 9 | Ranorex | Testing desktop and mobile applications |
| 10 | Coded UI | Testing user interfaces |
| 11 | Mercury Interactive Load Runner | Performance and load testing |
| 12 | Tellurium | Testing web applications |
| 13 | Selenium | Functional testing |

## VII. FEATURES

The popularity of Selenium is due to many of its attractive features as enlisted below-

**1. Open-Source:** Selenium is free of charge and portable tool. This tool can be downloaded freely. It is community based and the product support is freely available.

**2. Supports languages:** Selenium supports a various languages like Ruby, Groovy, Java, Perl, Java Script, Python, C# etc. Selenium has its own or individual script and doesn't bind it to those languages.

**3. Supports Operating Systems:** Selenium support many Operating Systems like Mac, Windows, UNIX, Linux, etc. Test cases can be created over any platform and then executed on another platform. Test cases can be created using Linux operating system and run that test cases on a windows operating system.

**4. Support across browsers:** Selenium provides support across multiple browsers like Internet Explorer, Chrome, Firefox, Opera, Safari, etc. A selenium package supports the following browsers:

a. Selenium IDE can be used with Mozilla Firefox browser as a plug-in.

b. Selenium WebDriver and Selenium RC supports various web browsers like Internet Explorer.

**5. Support for programming languages and frameworks:** Selenium can be merging with various frameworks and programming languages. Selenium can merge with TestNG testing framework for experimenting web applications and reporting purpose. For compilation of source code, Selenium can merge with ANT framework.

**6. Tests across devices:** Selenium tests executed for Mobile web applications on Android, Blackberry and Windows operating system.

**7. Constant updates:** Selenium product support is community based. An active community support enable constant updates and upgrades.

**8. Loaded Selenium Suits:** Selenium is not only one testing tool. Selenium is a loaded package of different testing tools. Hence, it is referred as a Suite. Each tool provides different testing needs and requirements of test environments.

**9. Ease of implementation:** Selenium provide easy to use interface which successfully create and execute the tests. Tests execute directly across the browsers. While the tests are being executed, user can watch it.

**10. Add-ons and Reusability:** Selenium tool uses the scripts and that scripts can be tested over the multiple web browsers. Concurrently, it is feasible to execute multi-tests with Selenium tool. Selenium covers all functional testing features by add-on tools implementation.

## VIII. LIMITATIONS

Apart from the above mentioned features there are certain limitations of Selenium enlisted as follows-

1. For any desktop applications Selenium doesn't support.

2. Since Selenium is open-source tool. We have to depend on community forums to resolve any technical issues.

3. For tests scripts creation in Selenium WebDriver, we should know at least one supported programming language.

4. Using Selenium, we can't execute automation tests on web services like REST (Representational State Transfer) or SOAP (Simple Object Access Protocol).

## IX. CONCLUSION

Two types of testing methods - Manual testing and Automation testing are discussed here along with the need

 DOI : 10.18231/2454-9150.2019.0162 

and importance of Automation testing. A very popular open source Automation testing software tool used for web application testing - Selenium is discussed in detail. Comparison of Selenium with other similar software tools like - HP QTP (UFT), TestComplete, SoapUI is provided in the paper based on various key features of software automation like - License cost, support platform, programming language support etc. Selenium found to be a better choice as it provides more flexibility to the tester to test applications under various complex scenarios. After reviewing a number of test tools available in market for different type of testing it can be concluded that Selenium is best for functionality testing of web applications.

## REFERENCES

[1] Prachi Kunte, "Automation Testing of Web based application with Selenium and HP UFT(QTP)", *IRJET*, Vol. 4, Issue:6, June-2017.

[2] Meenu, "Comparative Study of Automated Testing Tools: Selenium SoapUI, HP Unified Functional Testing and Test Complete", *JETIR*, ISSN-2349-5162, September 2015, Volume 2, Issue 9.

[3] Jyoti Devi, "A Study on Functioning of Selenium Automation testing Structure", *IJARCSSE*, Volume 7, Issue 5, May 2017

[4] Jagannatha S, "Comparative study on Automation Testing using Selenium Testing Framework and QTP", *IJCSMC*, Vol. 3, Issue. 10, October 2014, pg.258-267

[5] Kadambari Metha, "Towards Developing a Selenium Based GUI Automation Test Pro: a Comparative Study", *IJERT*, ISSN:2278-0181, Vol. 3 Issue2, February-2014

[6] Niranjanamurthy M, "Research Study on Web Application Testing using Selenium Testing Framework", *IJCSMC*, Vol. 3, Issue. 10, October 2014, pg. 121-126

[7] Harpreet Kaur, "Comparative Study of Automated Testing Tools: Selenium, Quick Test Professional and Testcomplete", *IJERA*, ISSN: 2248-9622, Vol.3, Issue 5, Sep-Oct 2013, pp.1739-1743.

 DOI : 10.18231/2454-9150.2019.0162 