# STQA Mini Project No. 2

**Group members -:**

1) Gibraan Jafar rollno - 421037 div - A

2) Divyank Shah rollno - 422048 div - B

3) Ankit Sonje rollno - 422060 div - B

4) Sneha Ukarande rollno - 422070 div - B

## Title

Create a small web-based application by selecting relevant system environment/platform and programming languages. Narrate concise Test Plan consisting features to be tested and bug taxonomy. Narrate scripts in order to perform regression tests. Identify the bugs using Selenium WebDriver and IDEand generate test reports encompassing exploratory testing.

## Problem Definition:

Perform Web testing and identify the bugs using Selenium WebDriver and IDEand generate test reports encompassing exploratory testing.

## Prerequisite:

Knowledge of Core Java

## Software Requirements:

Eclipse photon R latest Version, JAVA 1.8, selenium-server-standalone-3.13.0 Chromedriver.exe

## Hardware Requirement:

PIV, 2GB RAM, 500 GB HDD, Lenovo A13-4089Model.

## Learning Objectives:

We are going to learn how Identify the bugs using Selenium WebDriver and IDEand generate test reports encompassing exploratory testing.

## Outcomes:

You are able to Web Testing using Automation Tool like Selenium Web driver and IDE

## TheoryConcepts: What is Selenium?

Selenium is a free (open source) automated testing suite for web applications across different browsers and platforms.

Selenium is a suite of software tools to automate Web Browsers.

* It is an Open source suite of tools mainly used for Functional and Regression Test Automation.

Selenium is a free (open source) automated testing suite for web applications across different browsers and platforms.

It is quite similar to HP Quick Test Pro (QTP now UFT) only that Selenium focuses on automating web- based applications. Testing done using Selenium tool is usually referred as Selenium Testing.

## Selenium supports various Operating environments.

* + MS Windows
  + Linux
  + Macintosh etc…

## Selenium supports various Browsers.

* + Mozilla Firefox
  + IE
  + Google Chrome
  + Safari
  + Opera etc…

## Note: Selenium IDE supports Mozilla Firefox only.

* **Selenium supports various programming environments to write programs (Test scripts)**
  + Java
  + C#
  + Python
  + Perl
  + Ruby
  + PHP

## History of the Selenium Project

Selenium first came to life in 2004.

* In 2006, Selenium WebDriver was launched at Google.
* In 2008, the whole Selenium team decided to merge Selenium WebDriver with Selenium RC in order to form more powerful tool called Selenium 2.0

## Selenium 1

(Selenium IDE + Selenium RC + Selenium Grid)

## Selenium 2

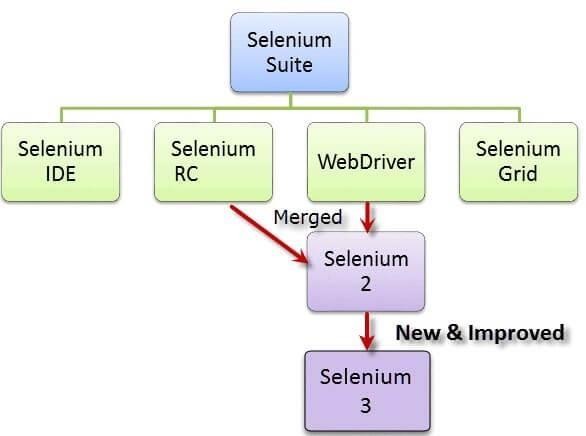
(Selenium IDE + Selenium RC + Selenium WebDriver + Selenium Grid)

## Selenium’s Tools Suite

Selenium is not just a single tool but a suite of software's, each catering to different testing needs of an organization.

## It has four components.

* **Selenium Integrated Development Environment (IDE)**
* **Selenium Remote Control (RC)**
* **WebDriver**
* **Selenium Grid**

****

**Brief Introduction Selenium IDE**

It is a Firefox browser plug in, used to create and execute Test cases.

## Selenium IDE Features:

* Create Test Cases, Test suites (We can Record test cases or type Test steps using element locators and Selenese commands)
* Edit Test Cases
* Execute Test cases, Test suites
* Debug Test Cases.
* Enhance Test Cases
* Export Test cases to other formats (java, ruby etc…) Note: **selenium IDE Test case default format is .html**

## Drawbacks of Selenium IDE

* It supports Mozilla Firefox browser only.
* It doesn’t support Programming logic/features to enhance Test cases.
* It doesn’t support Data Driven Testing.
* It is not suitable for complex test case design.
* No centralized maintenance of Objects/Elements

1. **Selenium RC (\* Out dated) -**Currently, Selenium RC is still being developed but only in maintenance mode.

## Selenium WebDriver

* + It is a Programming interface to create and execute Test cases.

Selenium IDE has IDE but doesn’t have Programming interface

* + Selenium WebDriver has Programming interface but doesn’t have IDE
  + It communicates Directly to the browser.
  + No need of Separate Server such as RC Server
  + UFT/QTP has both IDE as well as Programming interface
  + Faster Execution than IDE & RC

## Selenium WebDriver supports various programming environments to write programs.

* + Java,
  + C#
  + Perl
  + Python
  + Ruby
  + PHP
* Using Element/Object locators/properties and Webdriver Methods we can create and execute Test cases.
* Selenium Webdriver supports various browsers to create and execute test case/test script/test Note: **Browser driver varies from one browser to another.**

## Selenium WebDriver supports various operating environments

* + MS Windows
  + Linux Macintosh etc…

## Drawback of Selenium WebDriver

* It doesn’t generate detailed Test Reports.
* No centralized maintenance of Object/elements
* It require Programming Knowledge
* cannot support the readily new browser
* Installation is More Complicated than Selenium IDE
* No built-in mechanism for logging runtime message

## Selenium Grid

* Selenium Grid is used to execute tests across multiple browsers, operating environments and machines in parallel.
* Selenium Grid 2 supports Selenium RC Tests as well as Selenium WebDriver Tests.

1. Selenium WebDriver to create Test cases using element locators and Webdriver methods.
2. Java Programming to enhance test cases.
3. TestNG Framework to group test cases, execute test batches and generate detailed test reports.

## Features:

* + Enables **simultaneous running of tests** in **multiple browsers and environments.**
  + **Saves time** enormously.
  + Utilizes the **hub-and-nodes** concept. The hub acts as a central source of Selenium commands to each node connected to it.

## Note on Browser and Environment Support

* + Because of their architectural differences, Selenium IDE, Selenium RC, and WebDriver support different sets of browsers and operating environments.

|  |  |  |
| --- | --- | --- |
|  | **Selenium IDE** | **WebDriver** |
| **Browser Support** | Mozilla Firefox | Internet Explorer versions 6 to 11, both 32 and 64-bit  Microsoft Edge version 12.10240 & above ( partial support some |

|  |  |  |
| --- | --- | --- |
|  | **Selenium IDE** | **WebDriver** |
|  |  | functionalities under development) Firefox 3.0 and above  Google Chrome 12.0. and above Opera 11.5 and above  Android - 2.3 and above for phones and tablets (devices & emulators)  iOS 3+ for phones (devices & emulators) and 3.2+ for tablets (devices & emulators)  HtmlUnit 2.9 and above |
| **Operating System** | Windows,Mac OS X, Linux | All operating systems where the browsers above can run. |

* + **Note:** Selenium WebDriver is termed as the successor of Selenium RC which has been deprecated & officially announced by SeleniumHQ.

## How to Choose the Right Selenium Tool for Your Need

|  |  |  |
| --- | --- | --- |
| **Tool** | **Why Choose?** | |
|  |  | * To learn about concepts on automated testing and Selenium, including: * Selenese commands such as type, open, clickAndWait, assert, verify, etc. * Locators such as id, name, xpath, css selector, etc. * Executing customized JavaScript code using runScript * Exporting test cases in various formats. * To create tests with little or no prior knowledge in programming. * To create simple test cases and test suites that you can export later to RC or WebDriver. |
|  |  |
| **Selenium IDE** |
|  |
| * To test a web application against Firefox only. |
|  | |
| **Selenium RC** | * To design a test using a more expressive language than Selenese * To run your test against different browsers (except HtmlUnit) on different operating systems. * To deploy your tests across multiple environments using Selenium Grid. | |

|  |  |  |
| --- | --- | --- |
| **Tool** | **Why Choose?** | |
|  | * To test your application against a new browser that supports JavaScript. * To test web applications with complex AJAX-based scenarios. | |
|  |  | * To use a certain programming language in designing your test case. * To test applications that are rich in AJAX-based functionalities. * To execute tests on the HtmlUnit browser. |
|  |  |
| **WebDriver** |
|  |
| * To create customized test results. |
|  | |
| **Selenium Grid** | * To run your Selenium RC scripts in multiple browsers and operating systems simultaneously. * To run a huge test suite, that needs to complete in the soonest time possible. | |

**Advantages of Selenium**

1. It is an Open source Software.
2. It supports various Operating environments (Windows, Linux, Mac etc…)
3. It supports various browsers (IE, Mozilla Firefox, Chrome, safari, Opera etc…)
4. It supports various programming environments (Java, Perl, Python, Ruby and PHP)
5. It supports parallel Test execution.
6. It uses less Hardware resources.

## Disadvantages of Selenium

1. It supports Web based Applications only.
2. No reliable support from anybody.
3. No centralized maintenance of Elements/objects
4. Difficult to setup environment.
5. Difficult to use.
6. Limited support for Image based testing.
7. New features may not work properly.
8. No other tool integration for test management & No built in Reporting facility.

## SeleniumVersus UFT

|  |  |
| --- | --- |
| Selenium | UFT / QTP |
| 1) Open Source | Vendor tool, License is required. |
| 2) Supports various OS Environments. | MS Windows only. |
| 3) Supports various Programming Environments | VBScript only. |
| 4) No Object Repositories | Local and Shared object Repositories. |
| 5) No built-in Reporting feature. | Built-in reporting feature. |
| 6) Selenium WebDriver has no IDE and Selenium IDE has no Programming Interface. | UFT has both IDE and Programming Interface. |
| 7) Uses less Hardware resources. | Uses more Hardware resources |
| 8) Difficult to setup environment and use. | Easy to setup and use. |
| 9) Limited support for Image Testing | Rich support for Image Testing |
| 10) No Reliable support | Support from HP |
| 11) No other tool integration for Test management. | UFT can be integrated with ALM/QC for Test Management. |
| 12) New features may not work properly. | New features will properly. |
| 13) No Add ins for supporting Application Environments. | Add ins are required for supporting Application environments. |
| 14) Supports Web Applications only | Supports Desktop and Web Applications. |
| 15) No Authorized Certification | Authorized Certification program. |

**What is TestNG?**

TestNG is a powerful testing framework, an enhanced version of JUnit which was in use for a long time before TestNG came into existence. NG stands for 'Next Generation'.

TestNG framework provides the following features −

* + Annotations help us organize the tests easily.
  + Flexible test configuration.
  + Test cases can be grouped more easily.
  + Parallelization of tests can be achieved using TestNG.
  + Support for data-driven testing.
  + Inbuilt reporting.

Input -

package com.sel.demo;

import org.openqa.selenium.By;

import org.openqa.selenium.By.ById;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class demo {

public static void main(String[] args) throws InterruptedException {

System.setProperty("webdriver.chrome.driver" , "C:\\Users\\HP\\Desktop\\Selenium\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver=new ChromeDriver();

driver.get("http://www.gmail.com/");

driver.manage().window().maximize();

driver.findElement(By.id("identifierId")).sendKeys("divyank.shah@viit.ac.in");

Thread.sleep(2000);

driver.findElement(By.className("CwaK9")).click();

Thread.sleep(2000);

String at =driver.getTitle();

String et="gmail";

driver.close();

if(at.equalsIgnoreCase(et))

{

System.out.println("Test successfull");

}

else

{

System.out.println("Test failed");

}

}

}

Output -

