MODULE 4 : SOFTWARE TESTING

# Q1 : Which components have you used in Load Runner?

ANS : **LoadRunner** is a software testing tool from [Micro Focus](https://en.wikipedia.org/wiki/Micro_Focus). It is used to test [applications](https://en.wikipedia.org/wiki/Application_software), measuring system behaviour and performance under load. LoadRunner can simulate thousands of users concurrently using application software, recording and later analyzing the performance of key components of the application.

LoadRunner simulates user activity by generating messages between application components or by simulating interactions with the user interface such as keypresses or mouse movements. The messages and interactions to be generated are stored in scripts. LoadRunner can generate the scripts by recording them, such as logging [HTTP](https://en.wikipedia.org/wiki/HTTP) requests between a client web browser and an application's web server.[[1]](https://en.wikipedia.org/wiki/LoadRunner#cite_note-1)

**Load Generator** generates the load against the application by following scripts

**VuGen** (Virtual User Generator) for generating and editing scripts

**Controller** controls, launches and sequences instances of Load Generator - specifying which script to use, for how long etc. During runs the Controller receives real-time monitoring data and displays status.

**Agent process** manages connection between Controller and Load Generator instances.

**Analysis** assembles logs from various load generators and formats reports for visualization of run result data and monitoring data.

Q2 : How can you set the number of Vusers in Load Runner?

ANS : **N = Z \* (R + T)**

where N – number of VUsers,

             Z – Transactions per Second (TPS)

             R – Response Time in seconds

             T – Think Time in seconds

If you get the following data from the stakeholders i.e. TPS, Response Time and Think Time, number of VUsers can be calculated easily.

E.g.      TPS is 100, R is 3 sec and T is 2 sec then N will be

**N           = 100 \* (3+2)**

**= 100 \* 5**

**= 500**

Peak load will be 500 VUsers.

# Q3 : What is Correlation?

ANS : Correlation is the process of extracting some value from the response of one step into the request of another step. It captures and stores the dynamic response from the server and passes it on to subsequent requests.

# Q4 : What is the process for developing a Vuser Script ?

ANS : Step 1- Record the Vuser Script.

Step 2- Playback and improve the recorded vuser script.

Step 3- Define and test the different run-time parameters.

Step 4- Use the script in a LoadRunner scenario.

# Q5 : How Load Runner interacts with the application ?

ANS : LoadRunner is a software testing tool from Micro Focus. It is used to test applications, measuring system behaviour and performance under load. LoadRunner can simulate thousands of users concurrently using application software, recording and later analyzing the performance of key components of the application.

# Q6 : What is the relationship between Response Time and Throughput ?

ANS: Response time and throughput are related. The response time for an average transaction tends to decrease as you increase overall throughput. However, you can decrease the response time for a specific query, at the expense of overall throughput, by allocating a disproportionate amount of resources to that query.

# Q7 : What is the difference between hits/second and requests/second?

ANS : Hits per second means the number of hits the server receives in one second from the vuser. Request per second is the number of request the vuser will request from the server.

The exact meaning of those terms is context-sensitive. Normally, I'd consider "request" and "hit" to be synonymous. However, when talking about a caching webserver, a "hit" might refer to a cache hit (as opposed to a "miss")

# Q8 : What is Automation Testing?

ANS : **Automation Testing** is a software testing technique that performs using special automated testing software tools to execute a test case suite. On the contrary, Manual Testing is performed by a human sitting in front of a computer carefully executing the test steps.

The automation testing software can also enter test data into the System Under Test, compare expected and actual results and generate detailed test reports. Software Test Automation demands considerable investments of money and resources.

# Q9 : What are the benefits of Automation Testing?

ANS : Saves Time and Money: Testing applications is a continuous process. ...

Increases Test Coverage: ...

Improves Accuracy: ...

Helps Achieve Continuous Testing: ...

Enables Reusability : ...

Offers Faster Feedback : ...

Improves Team's Morale : ...

Promises Information Security :

# Q10 : What are the advantages of Selenium?

ANS: Selenium is the most popularly used freeware and open source automation tool. The benefits of Selenium for Test Automation are immense. Importantly, it enables record and playback for testing web applications and can run multiple scripts across various browsers. The benefits of Selenium Test Automation hold relevance across diverse business segments.

**1.Open-Source:**

As mentioned earlier, the biggest strength of Selenium is that it is a freeware and a portable tool. It has no upfront direct costs involved. The tool can be freely downloaded and the community-based support for it is freely available.

**2.Language support:**

Selenium supports a range of languages, including Java, Perl, Python, C#, Ruby, Groovy, JavaScript, and more. It has its own script, but it is not limited by that language. It can work with various languages – whatever the developers/testers are comfortable with.

**3. Supports Operating Systems:**

Selenium can operate and support across multiple Operating Systems (OS) like Windows, Mac, Linux, and UNIX. With Selenium suite of solutions, a tailored testing suite can be created over any platform and then executed on another one. For instance, you can create test cases using Windows OS and run it with ease on a Linux–based system.

**4.Support across browsers:**

Selenium provides support across multiple browsers, namely, Internet Explorer, Chrome, Firefox, Opera, and Safari. This becomes highly resourceful while executing tests and testing it across various browsers simultaneously.

**5.Support for programming languages and framework**

Selenium integrates with programming languages and various frameworks. For instance, it can integrate with ANT or Maven type of framework for source code compilation. Further, it can integrate with the TestNG framework for testing applications and reporting purposes. It can integrate with Jenkins or Hudson for Continuous Integration (CI) and can even integrate with other open-source tools to support other features.

**6.Tests across devices**

Selenium Test Automation can be implemented for mobile web application automation on Android, iPhone, and Blackberry. This can help in generating necessary results and address issues on a continuous basis.

**7.Constant updates**

Selenium support is community–based, which enable constant updates and upgrades. These upgrades are readily available and do not require specific training. This makes Selenium resourceful and cost-effective as well.

**8.Loaded Selenium suites**

Selenium is not just a singular tool or utility, it a loaded package of various testing tools and so, is referred to as a Suite. Each tool is designed to cater to different testing needs and requirements of test environments.

Additionally, Selenium comes with capabilities to support Selenium IDE, Selenium Grid, and Selenium Remote Control (RC).

**9.Ease of implementation**

Selenium offers a user-friendly interface that helps create and execute tests easily and effectively. Its open-source features help users to script their own extensions that make them easy to develop, customized actions and even manipulate at an advanced level.

Tests run directly across browsers and the users can watch while the tests are being executed. Additionally, Selenium’s reporting capabilities are one of the reasons for choosing it, as it allows testers to extract the results and take follow-up actions.

**10.Reusability and Add-ons**

Selenium Test Automation framework uses scripts that can be tested directly across multiple browsers. Concurrently, it is possible to execute multiple tests with Selenium, as it covers almost all aspects of functional testing by implementing add-on tools that broaden the scope of testing.

There is another school of thought that talks about some gaps in Selenium Automation Testing. For instance, Test Automation experts also say that Selenium is not an all-inclusive tool for automating the testing of web applications, as it needs third-party frameworks and language support to get absolutely functional and show the needed results.

# Q11: Why testers should opt for Selenium and not QTP ?

ANS: With Selenium grid, you have the ability to run as many tests as you want. Further, you can achieve this when you need and in parallel. You will just need a browser so that you can run the test. With QTP testing tool, there is no resource like selenium grid and hence selenium takes preference here