MODULE -4 AUTOMATION CORE TESTING

**Which components have you used in Load Runner?**

* COMPONENTS OF LOAD RUNNER :

1. Virtual user generator (VUGEN): It records V user scripts that emulate the step of real users using the application.
2. Controller: it is an administrative centre for creating, maintaining, and executing scenarios .starts and stops load tests, and perform other administrative tasks.
3. Analyser: uses the load test results to create graphs and reports that are used to correlate system information and identify both bottlenecks and performance issues.

**How can you set the number of V Users in Load Runner?**

1. Open Load Runner Controller.
2. Tap on Tools option given on the Toolbar.
3. Click on Vuser Generator.
4. In the Number of Vusers field, enter the desired number of Vusers.
5. Click OK.

**What is Correlation?**

* **Troubleshooting:** Correlation can be used to identify the root cause of a problem by finding variables that are correlated with the problem. For example, if there is a correlation between the number of website visitors and the number of server errors, then the website traffic may be causing the server errors.
* **Predictive analytics:** Correlation can be used to predict future events by identifying variables that are correlated with those events. For example, if there is a correlation between the number of online searches for a product and the number of sales of that product, then the number of online searches can be used to predict future sales.
* **Decision making:** Correlation can be used to make decisions by identifying variables that are correlated with the desired outcome. For example, if there is a correlation between the number of employees who receive training and the number of employees who are promoted, then the company may decide to increase the amount of training that it provides to its employees.

**What is the process for developing a Vuser Script?**

1. Identify the tasks that need to be performed. The first step is to identify the tasks that need to be performed by the virtual user. This can be done by analysing the user requirements for the web application.
2. Record the Vuser script. Once the tasks have been identified, the Vuser script can be recorded. This can be done by using the VuGen tool. VuGen allows you to record the actions that you perform on a web application.
3. Edit the Vuser script. After the Vuser script has been recorded, it may need to be edited. This may involve adding or removing steps, or changing the parameters of steps.
4. Set up the Vuser environment. Before the Vuser script can be executed, the Vuser environment needs to be set up. This includes configuring the Vuser machine, installing the LoadRunner software, and configuring the LoadRunner license.
5. Run the Vuser script. Once the Vuser environment has been set up, the Vuser script can be run. This can be done in stand-alone mode or in a LoadRunner scenario.

**How Load Runner interacts with the application?**

* LoadRunner interacts with an application using a protocol. A protocol is a set of rules that define how two systems communicate with each other. LoadRunner supports a wide variety of protocols, including HTTP, HTTPS, FTP, SMTP, and JMS.
* When LoadRunner starts a test, it creates a virtual user (vuser). The vuser is a software program that runs on the LoadRunner controller. The vuser reads a script that defines the actions that it should perform. The script can contain instructions to send requests to the application, receive responses from the application, and perform other actions.
* When the vuser sends a request to the application, it uses the appropriate protocol to do so. The application responds to the request using the same protocol. LoadRunner captures the request and response data. This data can be used to analyse the performance of the application.
* LoadRunner can also use a variety of other techniques to interact with an application. For example, LoadRunner can use a screen scraping technique to interact with a web application that does not support a standard protocol.

**How many VUsers are required for load testing?**

The number of virtual users (Vusers) required for load testing depends on a number of factors, including:

* The expected number of users on your website or application during peak traffic times
* The type of load testing you are performing (e.g., performance testing, stress testing)
* The resources available to you (e.g., hardware, software)
* A good starting point is to estimate the expected number of users on your website or application during peak traffic times. You can then use this number to calculate the number of Vusers required for your load test. For example, if you expect 1000 users on your website during peak traffic times, you would need to run a load test with at least 1000 Vusers.

**What is the relationship between Response Time and Throughput?**

* Response time and throughput are two important metrics used to measure the performance of a system. Response time is the amount of time it takes for a system to respond to a request, while throughput is the number of requests that a system can handle per unit of time.
* Response time and throughput are inversely related. This means that as response time decreases, throughput increases. Conversely, as response time increases, throughput decreases.

There are a number of factors that can affect response time and throughput, including:

* The number of users accessing the system
* The amount of data being transferred
* The complexity of the system
* The performance of the hardware and software
* It is important to monitor response time and throughput to ensure that the system is performing as expected. If response time is too high, users may experience delays or errors. If throughput is too low, the system may not be able to handle the load.

**What is the difference between hits/second and requests/second?**

* The terms "hits" and "requests" are often used interchangeably, but there is a subtle difference between the two. A "hit" is a single unit of data that is transferred between a web server and a web client. A "request" is a single action that is performed by a web client, such as loading a web page.
* In most cases, a single request will result in multiple hits. For example, loading a web page will result in hits for the HTML, CSS, JavaScript, and images that make up the page. However, it is possible for a single hit to result in multiple requests. For example, if a web page contains a video, the video will be streamed in multiple chunks, each of which will result in a separate request.
* As a result of this difference, the terms "hits per second" and "requests per second" can measure different things. "Hits per second" measures the number of units of data that are transferred between a web server and a web client, while "requests per second" measures the number of actions that are performed by a web client.
* In general, "requests per second" is a more accurate measure of the load on a web server. This is because a single request can result in multiple hits, and the number of hits can be affected by factors such as the size of the resources that are being transferred.
* However, "hits per second" can be a useful metric for measuring the overall volume of traffic to a website. This is because it is easier to measure the number of hits than the number of requests.
* Ultimately, the best metric to use depends on the specific needs of the situation. If you are trying to measure the load on a web server, then "requests per second" is the best metric to use. If you are trying to measure the overall volume of traffic to a website, then "hits per second" is the best metric to use.

**To test the Performance testing on “Tops Technologies Website”:-**

1. **To Record all the top level menu**

* **ERROR IN SCRIPT**

**Create a Normal Script of above website with correlate using HP Default website.**

* **LINK IS NOT WORKING**

**What is Automation testing?**

* Automation testing is the process of testing software and other tech products to ensure it meets strict requirements. Essentially, it's a test to double-check that the equipment or software does exactly what it was designed to do. It tests for bugs, defects, and any other issues that can arise with product development.

**Which are the Browsers Supported by Selenium Ide?**

Selenium IDE supports the following browsers:

1. Google Chrome
2. Mozilla Firefox
3. Microsoft Edge (legacy version)

* Selenium IDE is a free and open-source integrated development environment (IDE) for Selenium WebDriver. It allows users to record and playback user interactions with a web application, including clicks, text inputs, and navigation. Selenium IDE can be used to create automated tests for web applications.
* To use Selenium IDE, you will need to install it as a browser extension. Once it is installed, you can start recording your test by clicking on the "Record" button. As you interact with the web application, Selenium IDE will record your actions. When you are finished recording, you can save your test and then playback the test by clicking on the "Play" button.

**What are the benefits of Automation Testing?**

There are many benefits to automation testing, including:

* **Increased test coverage:** Automated tests can be run much faster than manual tests, which allows testers to test more scenarios and combinations of data. This can lead to increased test coverage, which can help to ensure that the software is more thoroughly tested and that fewer bugs are missed.
* **Improved accuracy:** Automated tests can be run more consistently than manual tests, which can help to improve the accuracy of the testing process. This is because automated tests are not subject to human error, such as fatigue or boredom.
* **Reduced time to market:** Automated tests can help to reduce the time it takes to bring new software to market. This is because automated tests can be run in parallel with development, which can help to catch bugs early and prevent them from delaying the release of the software.
* **Improved quality:** Automated tests can help to improve the quality of software by ensuring that it is thoroughly tested and that any bugs are found and fixed early. This can lead to a better user experience and fewer customer support issues.
* **Reduced costs:** Automated tests can help to reduce the costs associated with testing, such as the cost of manual labour, the cost of test data, and the cost of test environments.
* **Increased productivity:** Automated tests can free up testers to focus on other tasks, such as designing new tests, creating test data, and analysing test results. This can lead to increased productivity and a more efficient testing process.

**What are the advantages of Selenium?**

* Selenium is a popular open-source automation testing tool for web applications. It is free to use and can be run on a variety of operating systems and browsers. Selenium also supports a variety of programming languages, making it a flexible tool for automation testing.

Here are some of the advantages of Selenium:

* **Open-source and free:** Selenium is an open-source tool, which means that it is free to use and modify. This makes it a cost-effective option for automation testing.
* **Cross-platform:** Selenium can be run on a variety of operating systems, including Windows, Linux, and macOS. This makes it a versatile tool for testing web applications that are used by a global audience.
* **Browser support:** Selenium supports a variety of web browsers, including Chrome, Firefox, Internet Explorer, and Safari. This makes it possible to test web applications across a wide range of browsers.
* **Flexible programming languages:** Selenium can be used with a variety of programming languages, including Java, Python, C#, and Ruby. This makes it a flexible tool for automation testing teams with different skill sets.
* **Community support:** Selenium has a large and active community of users and developers. This community provides support and resources for users of all skill levels.
* Overall, Selenium is a powerful and versatile automation testing tool that is free to use and has a large community of users and developers. These factors make Selenium a popular choice for automation testing teams of all sizes.

**Why tester should OPT for Selenium and not QTP?**

|  |  |
| --- | --- |
| **SELENIUM** | **QTP** |
| Open source, free to use, and free of charge. | Commercial. |
| Highly extensible | Limited add-ons |
| Can run tests across different browsers | Can only run tests in Firefox , Internet Explorer and Chrome |
| Supports various operating systems | Can only be used in Windows |
| Supports mobile devices | Supports mobile devise using 3rd party software |
| Can execute tests while the browser is minimized | Needs to have the application under test to be visible on the desktop |
| Can execute tests in parallel. | Can only execute in parallel but using Quality Center which is again a paid product. |

**To validate the tops technologies website Contact us page and enter your friend detail at last “Guest Call Back”** [**https://www.tops-int.com/contact-us/**](https://www.tops-int.com/contact-us/)

**Need Revision**