SOFTWARE TESTING ASSIGNMENT

Module-4(Automation core testing)

1. Which components have you used in Load Runner?

- Virtual User Generator (VUGen): Used for recording and creating virtual user scripts that simulate user actions on an application.
- Controller: Used for managing and controlling load testing scenarios, including defining test scenarios, scheduling tests, and monitoring test results.
- Analysis: Used for analyzing and interpreting the results of load tests, including generating reports and graphs for performance analysis.

2. How can you set the number of Vusers in Load Runner?

- Launch the LoadRunner Controller.
- Create a new scenario or open an existing one.
- In the "Scenario" section, go to the "Vuser Groups" tab.
- Select the Vuser group for which you want to set the number of Vusers.
- In the "Number of Vusers" field, enter the desired number of Vusers that you want to simulate during the load test.
- You can also set additional options like ramp-up time, duration, and iterations for the Vusers in the same tab.
- Save the scenario.
- Start the load test by clicking the "Run" button in the Controller.

3. What is Correlation?

Correlation is a technique used in software testing to capture and replace dynamic values in server responses with parameterized values in subsequent requests. It helps ensure that test scripts accurately replicate user interactions with web applications by handling changing data, such as session IDs or user tokens. Correlation is commonly used in performance testing and web application testing to achieve realistic and accurate test results.

4. What is the process for developing a Vuser Script?

- Recording: User interactions with the application are recorded to generate a Vuser script.
- Enhancing: The recorded script is enhanced with parameterization, correlation, and data manipulation.
- Validating: The script is validated for accuracy and correctness.
- Customizing: The script is customized to simulate different scenarios or load levels.
- Execution: The script is executed as part of a load testing scenario.
- Maintenance: The script may require periodic maintenance to keep it up-to-date with changes in the application or system under test.

5. How Load Runner interacts with the application?

- Recording: Load Runner records user interactions with the application during script recording process.
- Playback: Load Runner replays the recorded script, sending requests and receiving responses from the application.
- Parameterization: Load Runner allows for parameterization of dynamic values in the script to simulate realistic user behavior.
- Correlation: Load Runner automatically correlates dynamic values in the script for accurate replay.
- Data Manipulation: Load Runner allows for data manipulation in the script, such as extracting data from responses and modifying request data.
- Monitoring: Load Runner captures performance metrics of the application during script execution.
- Analysis: Load Runner provides built-in analysis tools to interpret captured performance metrics and identify performance bottlenecks.

6. How many VUsers are required for load testing?

The number of Vusers (virtual users) required for load testing depends on factors such as application complexity, performance goals, and available resources. Typically, load testing involves simulating realistic user loads, and the number of Vusers needed varies for each application. It is important to carefully analyze and plan for the appropriate number of Vusers to ensure meaningful and accurate load testing results.

7. What is the relationship between Response Time and Throughput?

- As response time increases, throughput tends to decrease, and vice versa.
- Longer response time can lead to lower throughput or capacity to handle concurrent requests.
- Shorter response time can lead to higher throughput or capacity to handle concurrent requests.
- Achieving a balance between response time and throughput is important for optimal application performance in load testing.

8. What is the difference between hits/second and requests/second?

hits/second	requests/second
Hits/second measures the rate at which cached content is served from the server's cache.	Requests/second measures the rate at which the server receives incoming requests from clients.
Hits/second is an indicator of the server's efficiency in serving cached content.	Requests/second reflects the server's ability to handle incoming requests and process them in a timely manner.
Hits/second can be significantly higher than requests/second in scenarios with effective caching mechanisms.	Requests/second is a critical performance metric that indicates the server's capacity to handle concurrent requests.
Hits/second provides insight into the server's ability to handle content-heavy websites or applications with significant caching.	Requests/second can be affected by various factors such as server configuration, hardware resources, network latency, etc.
Hits/second may not accurately represent the overall server workload, as it only considers cached content and does not account for non-cached content, redirects, errors, etc.	Requests/second provides insights into the server's scalability, performance, and ability to handle high levels of concurrent traffic.

Module-5(Selenium IDE)

1. What is Automation Testing?

Automation testing is the use of software tools and scripts to perform repetitive and complex testing tasks in a software application, without human intervention. It involves writing test scripts, creating test cases, and using automation tools to execute tests, validate results, and identify defects or issues in the software system. Automation testing helps to improve efficiency, accuracy, and repeatability of testing processes, and enables faster and more reliable software releases.

2. Which Are The Browsers Supported By Selenium Ide?

- Mozilla Firefox
- Google Chrome
- Microsoft Edge (Chromium-based)
- Safari (Limited support)
- Opera (Limited support)

3. What are the benefits of Automation Testing?

- Increased efficiency and time savings.
- Improved test coverage and accuracy.
- Faster feedback on software quality.
- Cost-effective in the long run.
- Scalability for large and complex applications.
- Effective for regression testing.
- Reusability of test scripts.
- Better traceability and documentation of test results.
- Enhanced software quality and reliability.

4. What are the advantages of Selenium?

- Open source and free to use.
- Supports multiple programming languages.
- Cross-platform compatibility for testing on different operating systems.
- Large community and active support.
- Extensible with various frameworks and tools for enhanced testing capabilities.

5. Why testers should opt for Selenium and not QTP?

Testers should choose Selenium over QTP (UFT) due to the advantages of open-source nature, support for multiple programming languages, cross-platform compatibility, flexibility, customization, and a large active community for support.