Unit 2 Test Automation

2.1 Test Automation vs. Manual Testing:

Automation Testing:

Automation testing is a process of changing any manual test case into the test scripts by using automation testing tools, and scripting or programming language is called automation.

Automation testing is used to increase the efficiency, effectiveness, and coverage of Software testing.

Automation test engineer uses automation testing tools to automate the manual design test cases without any human interference.

And these testing tools can control the execution of tests, access the test data, and compares the actual result against the expected result.

Manual testing:

Manual testing is testing of the software where tests are executed manually by the tester. It is performed to discover bugs in software under development.

In Manual testing, the tester checks all the essential features of the given application or software. In this process, the software testers execute the test cases and generate the test reports without the help of any automation software testing tools.

Parameter	Automation Testing	Manual Testing
Definition	Automation Testing uses automation tools to execute test cases.	In manual testing, test cases are executed by a human tester and software.

Reliability	It is reliable because it tests the application with the help of tools and test scripts.	It is not reliable because there is a possibility of human error, which may not be delivered the bug-free application.
Reused	The script can be reused across multiple releases.	It could be possible when the test case only needs to run once or twice.
Batch Execution	Batch execution is possible using automation testing because all the written scripts can be executed parallelly or simultaneously.	Batch execution is not possible in manual testing.
Programming knowledge	Without having an understanding of programming language, we cannot write the test script.	There is no need to know programming language but should have the product knowledge to write the test case.
Parallel Execution	This testing can be executed on different operating platforms in parallel and reduce test execution time.	Manual tests can be executed in parallel but would need to increase your human resource which is expensive
Processing time	Automated testing is significantly faster than a manual approach.	Manual testing is time-consuming and takes up human resources.
Operating system compatibility	Automation testing can also be performed on different systems with different	Operating system compatibility is not possible in manual

	operating system platforms and various programming languages.	testing because the different tester is required to perform such tasks.
Regression testing	Whenever the code changes happen due to the enhancement of the release, then automation test engineer performs the regression testing.	When the test engineer executes the test case for the first time, it may be useful, but there is a possibility that it will not catch the regression bugs because of changing requirements frequently.

2.2 Automation Testing tools:

WinRunner:

WinRunner is an Automation Software Testing Tool that is owned by HP and was developed by Mercury Interactive.

HP WinRunner software was an automated functional GUI testing tool that allowed a user to record and play back user interface (UI) interactions as test scripts.

It is known and extensively used for its ability to supports the majority of the programming languages and web development technologies such as C, C++, C#, Visual Basic, Java, HTML etc.

It is used for performing various testing techniques, which includes the functional testing, user interface testing, integration testing, regression testing, etc.

Advantages:

- Enables Rapid Testing

- Provides Consistency
- Reusability of tests
- Customizable for future changes

Disadvantages

- Doesn't apply for Load or Scalability Testing.
- Doesn't support .net programming
- Tester should have programming knowledge/experience
- It doesn't support multimedia systems.

The Testing process in WinRunner flows from the identification of functions, the recording of application activity, the test script management, the execution of tests, and finally the analysis & reporting of the gathered test results.

JMeter:

The Apache JMeter is pure Java open source software, which was first developed by the Apache Software Foundation, designed to load test functional behaviour and measure performance.

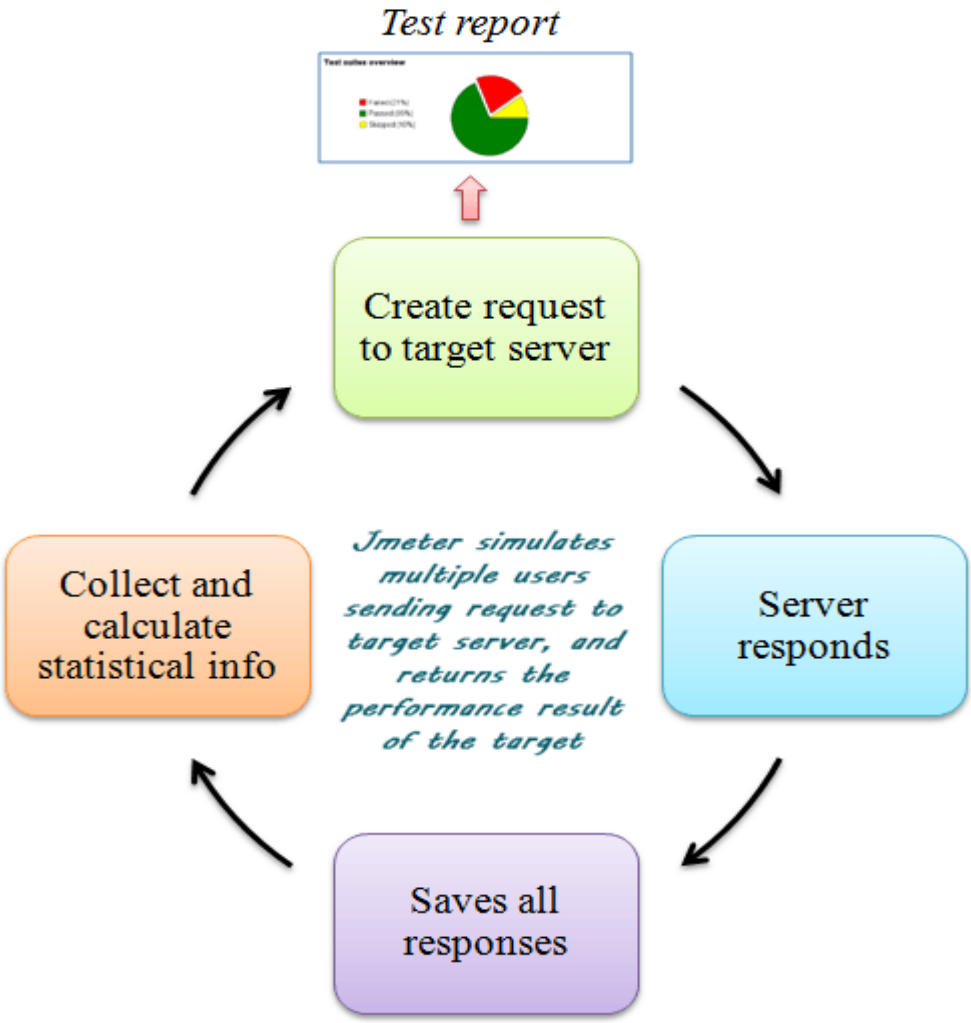
You can use JMeter to analyse and measure the performance of web application or a variety of services.

Performance Testing means testing a web application against heavy load, multiple and concurrent user traffic.

JMeter originally is used for testing Web Application or FTP application.

Nowadays, it is used for a functional test, database server test etc.

Workflow of JMeter:



JMeter simulates a group of users sending requests to a target server, and return statistics information of target server through graphical diagrams.

Applications of JMeter:

1. Open source license: JMeter is totally free, allows developer use the source code for the development
2. Friendly GUI: JMeter is extremely easy to use and doesn't take time to get familiar with it.
3. Support multi-protocol: JMeter does not only support web application testing but also evaluate database server performance. All basic protocols such as HTTP, JDBC, LDAP, SOAP, JMS, and FTP are supported by JMeter.

4. Platform independent: JMeter is 100% pure Java desktop application. So it can run on multiple platforms.
5. Visualize Test Result: Test result can be displayed in a different format such as chart, table, tree and log file.
6. Record & Playback - Record the user activity on the browser and simulate them in a web application using JMeter.

Test Director:

Software Automated Tool TestDirector simplifies test management by helping you organize and manage all phases of the software testing process, including planning, creating tests, executing tests, and tracking defects.

With TestDirector, you maintain a project's database of tests.

As you execute tests, TestDirector lets you report defects detected in the software. Defect records are stored in a database where you can track them until they are resolved in the software.

TestDirector works together with WinRunner, Mercury Interactive's automated GUI Testing tool.

WinRunner enables you to create and execute automated test scripts. You can include WinRunner automated tests in your project, and execute them directly from TestDirector.

TestDirector activates WinRunner, runs the tests, and displays the results, TestDirector offers integration with other Mercury Interactive testing tools (LoadRunner, Visual API and XRunner), as well as with third-party and custom testing tools.

IBM Rational:

IBM Rational Functional Tester is an automated functional testing and regression testing tool.

This software provides automated testing capabilities for functional, regression, GUI and data-driven testing.

It supports a range of applications, such as web-based, .Net, Java, Siebel, SAP, terminal emulator-based applications, PowerBuilder, Ajax, Adobe Flex, Dojo Toolkit, GEF, Adobe PDF documents.

LoadRunner:

LoadRunner is a Performance Testing tool which was developed by Mercury in 1999.

LoadRunner was later acquired by HPE in 2006. In 2016, LoadRunner was acquired by MicroFocus.

LoadRunner supports various development tools, technologies and communication protocols. In fact, this is the only tool in market which supports such a large number of protocols to conduct Performance Testing.

Performance Test Results produced by LoadRunner software are used as a benchmark against other tools.

LoadRunner works on a principal of simulating Virtual Users on the subject application. These Virtual Users also termed as VUsers, replicate client's requests and expect a corresponding response to passing a transaction.

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 requests between a client web browser and an application's web server.

LoadRunner tool supports HTTP/HTML, Ajax, Mobile, SAP, Oracle, MS SQL Server, Mail.

There is no competitor tool in the market which could offer such wide variety of protocols vested in a single tool.

2.3 Criteria for selecting test tools:

1. Flexibility and Ease of Use
    - In the absence of a proper tool, MS Excel is the most popular offline medium for managing test cases. One of the major reasons for this is the ease of use and flexibility of the tool.
    - So, while moving from Excel to a standard test tool, it has to be made sure that the tool is very easy to use and its training and user adaptation time is very less.
    - Every organization, and in some cases every project follows their own model of testing.
    - The testing tool should be configurable enough to support these model variances.
2. Support for End-to-End Traceability
    - It is very important for the testers to able to trace back all their work in a centralized test management system.
    - A bi-directional traceability between test cases and the associated requirements and defects increase the efficiency of measuring quality of a project.
    - It also allows organizations to track the coverage of both Requirements and Test Cases, failing which may lead to missing information, loss of productivity and fall in quality.
3. Real-time Reports and Dashboards
    - Most software projects fail due to the lack of proper visualization of analytical data related to a project's progress.

- In the absence of a centralized tool, the entire process of reporting is dependent on manual interactions, making it error-prone.
- So, the tool to be procured should have the facility of providing real-time reports and dashboards, keeping stakeholders updated with the latest status of progress and assess quality at every step.

4. Support for Test Automation
    - Today, due to market competition, Test Automation is no more a choice but has become a mandate for organizations.
    - A testing tool must have the support for managing Test Automation scripts from a single repository.
    - Features like the central execution of test automation scripts, automatic capturing of test results and making them visible from a central platform are necessary.
    - Viewing Test Automation results needs to be a part of the end-to-end traceability chain.
5. Integration with Other Phases of Application Lifecycle
    - Today, testing is no longer an isolated phase or a security gate to final delivery, but an integral part of the entire lifecycle.
    - To achieve this and ensure quality right from the beginning, testing should get involved at every stage of the lifecycle.
    - Therefore, a testing tool should have the capability to integrate with tools from other phases of the lifecycle, so that a centralized status update on the project's progress and quality can be achieved.