**Performance Testing – Very important for any website to test its scalability, responsiveness which is key point for end user.**

**Performance Testing-** definitely we need some tool

**tools:** Jmeter**,** Load Runner, Silk Performer

**Jmeter-**100% java based open source tool used to explicitly do performance testing – Load(Concurrent request server can handle within min response time), volume testing(db records), endurance testing(more amount of time)

**Different protocols supported/that can be tested by Jmter:**

Web-http/https, Web services – soap/xml, Database-jdbc

Messaging,Mail- Smtp,Jms,MailReader Files – ftp, LDAP etc

**What angles you think of website as user or stakeholder-developer/QA/BA:**

User Perspective- how **fast** is the response, time – he don’t care about the code, how many people are using the website, db issue.

Stakeholder perspective(Developer,BA, ProjMgr, QA) -Technical & MemoryAspects

**Where do you start:**

Make sure that application is functionally stable before starting load testing – there is no point in doing load testing without functionality.

How many Concurrent users my website can take – 10000

How much is the minimum response time

Depending upon the core functionality – get the information of the pages/ scenarios to test.

Setup – Tools required, system, OS-mac, windows, CPU, memory, N/W Bandwidth, DB

Huge requests – Grid is to handle larger requests from multiple systems- Master(admin system with test pan)/Slaves(systems which give different loads)

**How do you start implementing performance testing using jmeter or any tool:**

Send Requests – ThreadGroup, Samplers

Logically handling requests - Controllers

Validate & verify Check whether requests are working fine - Assertions

Analysing,Observe and Report – Listeners

**Jmeter Elements:**

**ThreadGroup:**

Thread = User, no of Threads – no of users

Rampup period – the time taken to make the threads up and running

Eg: users = 5 and ramp up =20 i.e., for every 4 seconds it starts a new thread(user request)

**Understanding Rampup:**

Threads- users-4

Response Time – time for execution of a request of website - Ist request is taking 1 second

Ramp up period – delay between for starting 2 or more - threads total time to start all the threads - 2 seconds

Within 8 seconds 4 requests would start

0th sec – 1st user

0.5nd sec – 2 user

1th sec – 3rd user

1.5 sec - 4th user

**Loop count** – no of times requests get repeated

**Total requests** -Loop count \* no of threads

**Scheduler**- to schedule load test at particular time

**Things to observe in output – depending on which u again tune rampup and assertions. Also these parameters are main output of your load testing to analyse your website performance.**

LoadTIme/ResponseTime – total time to get the response

ConnectTime – time taken to establich connection to server

Latency – subset of your response time i.e., time to get first byte of the response

Content- length- total size of response in bytes

ThroughPut: number of request/ total time or requests/sec

Sample error % - 5%- depends upon projects

**Workbench –** not part of ur test plan – but it helps to Add benefits to your test plan

**Options-Log Viewer-** which shows logs of jmeter so that we can do basic debugging of any error by reading the logs.

**Recording the requests:**

You need proxy configuration to route the requests from jmeter so that jmeter will capture and record your actions in browser – some kind of link b/n your jmeter and the browser

Do manual proxy settings to ur jmeter **HttpTestScriptRecorder** port and check if proxy is established fine.

**Workbench – HttpTestScriptRecorder – start the recorder**

Go to browser and perform the actions which u wanted to get recorded in ur jmeter- u need to observe recording controller when u r doing actions on browser

**Samplers- kind of protocol/request you use- http, jdbc, soap, rest, ftp, smtp**

**LogicControllers –** RecordingController, Simple Controller, ForEach,If, Random

**Assertions-** html,response,duration,size,xml,xpath

**ConfigElements:** Add logic to requests, User defined variables, csv data config- **Data Driven Testing ,** HttpRequestDEfaults**,** HttpHeaderManager**,** HttpCookieManager**,** HttpAuthorization

**Timers**- delays b/n elements - constanttimer, random timer, constant thrughtput timer

**Preprocessors**- to deal with logic that get executed before request- userparameters, jdbc preprocessor, link parsing

**PostProcessors**- logic to get executed after request – xpath extractor, jdbc post processor, result status action

**Listeners**: to observe o/ps- view results tree, summary report, aggregate graph, graph report etc

**Csvdata,xpath,textnode assertion**

**Additonal plugins to view different ouput results:**

**Cpu utilisation of server- perfmon plugin**

**Compate 2 results by giving filepath in viewresult tree**

**SoapRequest**

**JDBC request- connection details of Db – jdbc connection configuration**

**Download and Add ojdbc jar to jmeter/lib**

**Rest REquest**