ENSE 375 Jmeter Documentation

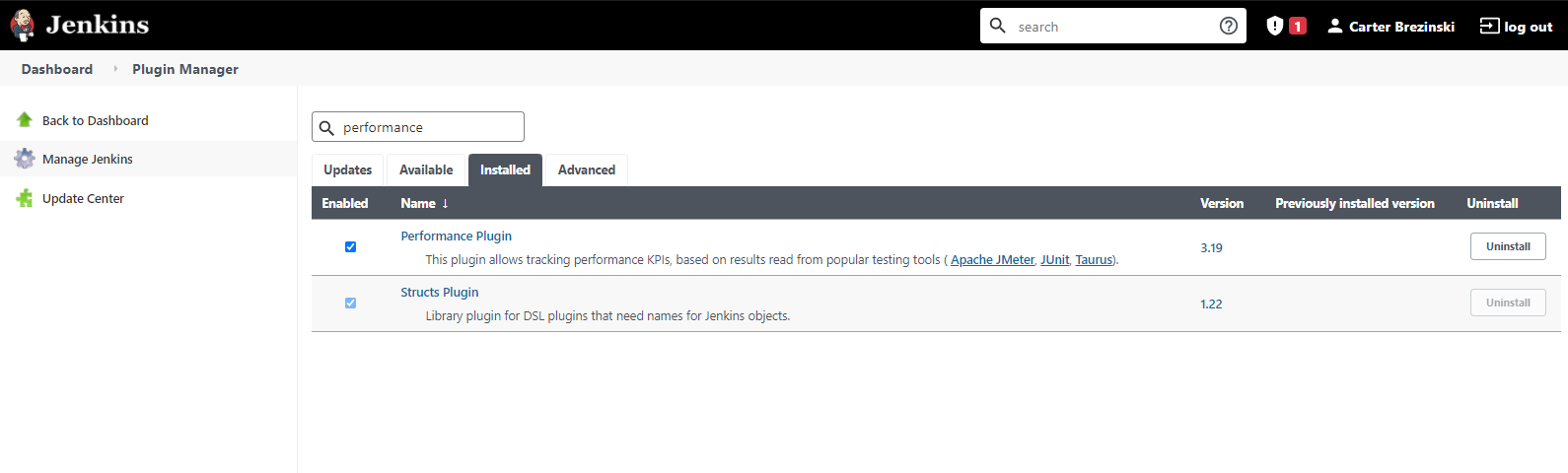
Carter Brezinski, Li Pan, Yash Patel, Abdelrahman Rabaa, Jacob Sauer

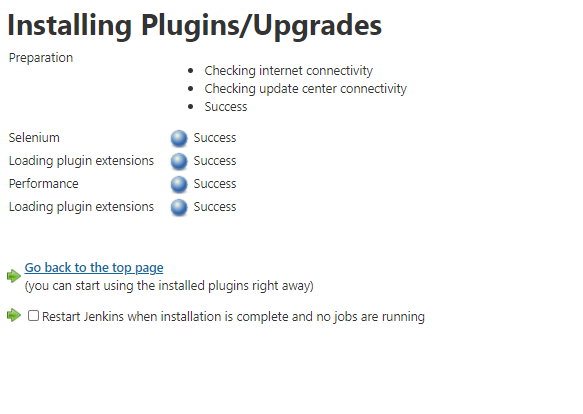
Jmeter Introduction:

Jmeter as it stands is one of the most useful open-source tools for CI and performance testing. It can be used to test the performance of static and dynamic websites and web applications. One of the benefits of using a plugin like Jmeter with Jenkins is that it provides a secure test report of each build with information on the failures that occurred within the build.

Installing Jmeter:

The installation and incorporation of the Jmeter service comes from linking the performance plugin to the Jmeter files. These are the steps if a person would want it to be working locally.

The first step in having a working installation of Jmeter is having the performance plugin installed on your system.



1. The next step is to install Jmeter files.

These files will either be located locally or on your github repository.

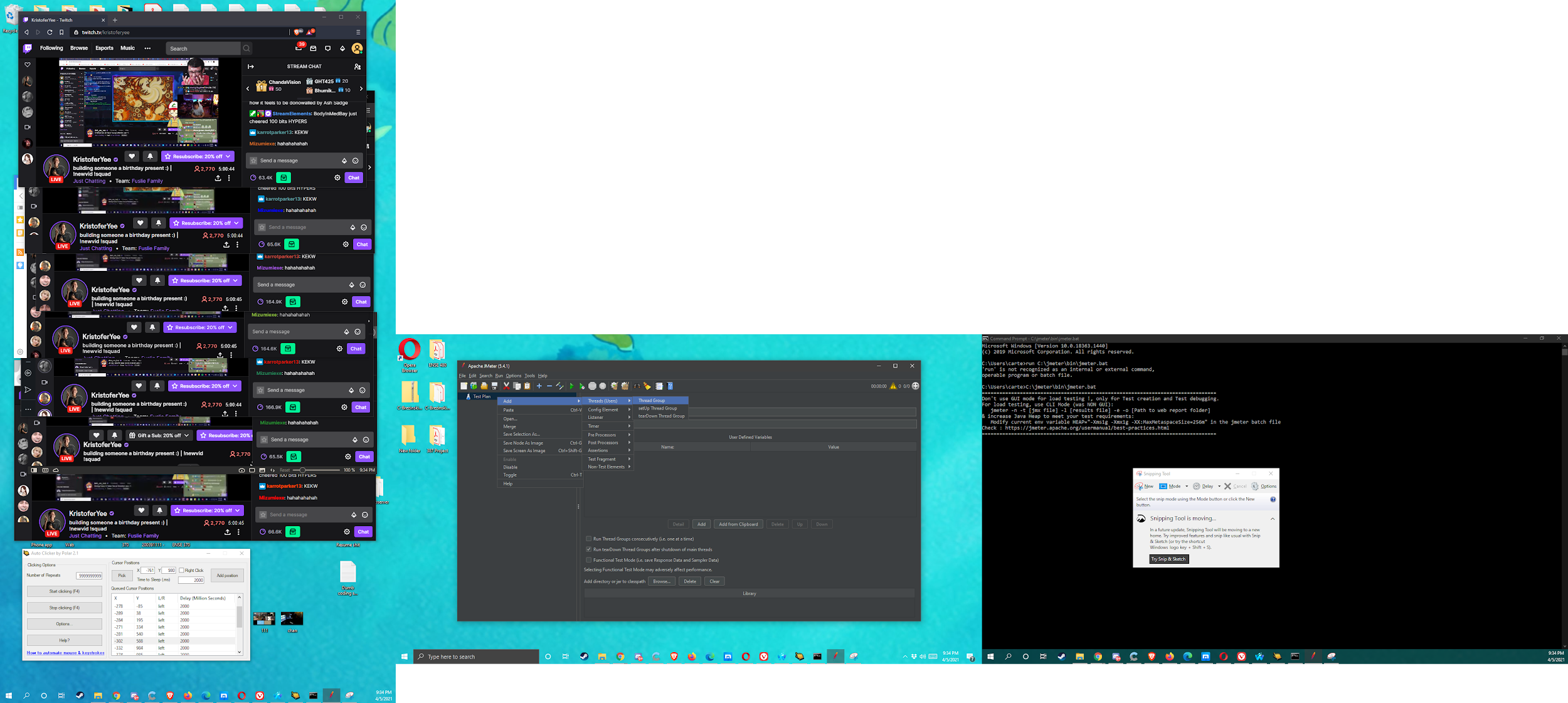
The files can be found here: <https://jmeter.apache.org/download_jmeter.cgi>

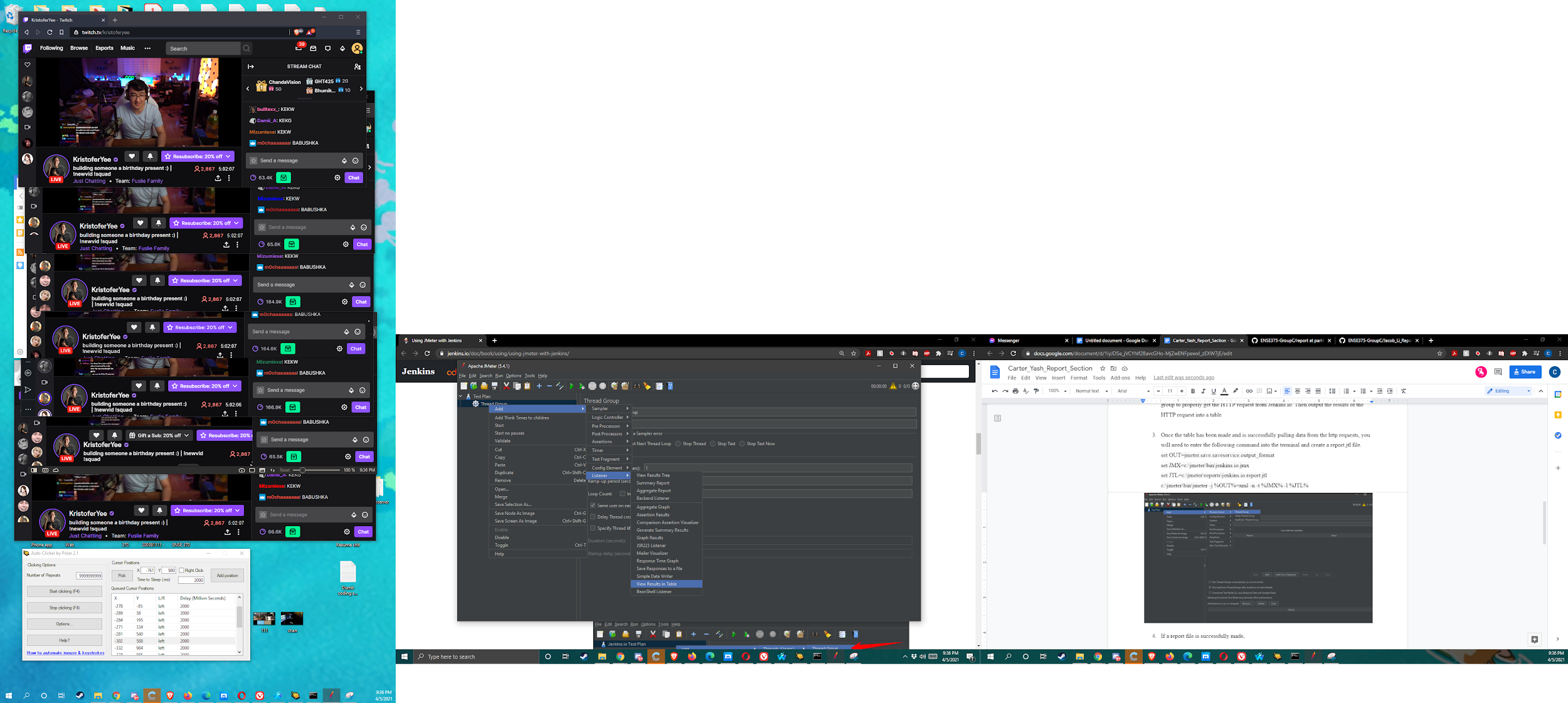
Once the files have been extracted to wherever the user desires, you need to edit the user.properties file.

The file should include this at the bottom of the properties text file:

“jmeter.save.saveservice.output\_format=xml”

1. After this, the user needs to run the Jmeter.bat file, and configure a test plan and thread group to properly get the HTTP request from Jenkins.io. Then output the results of the HTTP request into a table. It’s important for the HTTP requests to be properly configured to the systems server name and port number.





1. Once the table has been made and is successfully pulling data from the http requests, you will need to enter the following command into the terminal and create a report.jtl file.

set OUT=jmeter.save.saveservice.output\_format

set JMX=c:\jmeter\bin\jenkins.io.jmx

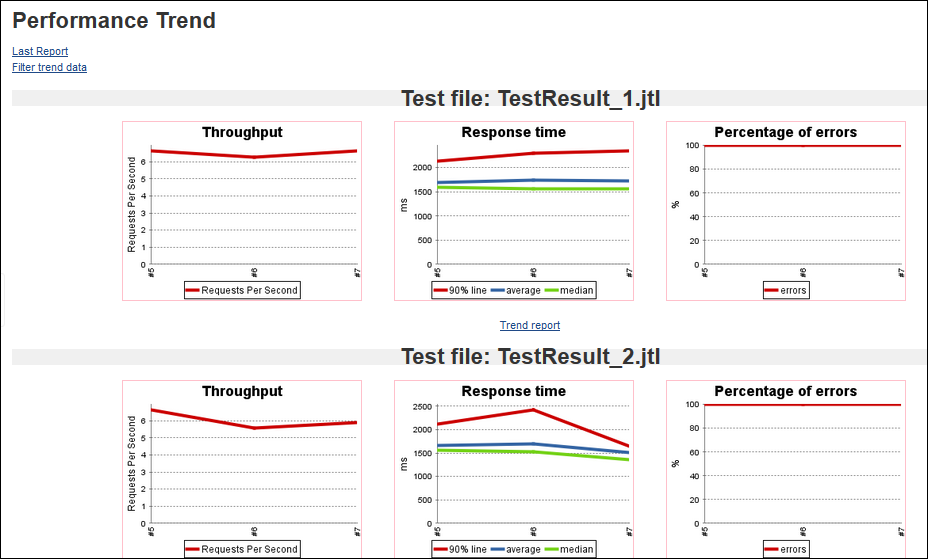
set JTL=c:\jmeter\reports\jenkins.io.report.jtl

c:\jmeter\bin\jmeter -j %OUT%=xml -n -t %JMX% -l %JTL%

1. Within the Jenkins item configuration, tutorials recommend making a build action with the following batch command (The code portions involving \*path location\* should be the path location of your jmeter folder/files.) :

“\*path location\* -j jmeter.save.saveservice.output\_format=xml -n -t \*path location\* -l \*path location\reports\jenkins.io.report.jtl\*“

1. And a post-build action to publish the performance test report where the source data file is the current location of the jenkins.io.report.jtl file present either locally or in the repository.
2. Finally once these steps are all complete, users should be able to run a build in jenkins and see a tab labelled “Performance Report”. This will display the Jmeter table showing the progress report of the current build. An example of this is shown below:



Group Issues Encountered:

In our installation of Jmeter, many of the tutorials we encountered online work of the idea that the Jmeter files are configured and work off of a local system, not a github repository.

Many of the group members gave an attempt on their different computers and operating systems to try and have Jmeter working both locally and from the repository. One group member even attempted to have our Jenkins running off an Ec2 service and have the Jmeter files located locally there, and unfortunately we weren’t able to get a solid final working build with Jmeter installed.

Yash Issues Encountered:

I saw a couple of different methods online for installation of Jmeter on my system, but almost all of those methods only seemed to work for local files rather than github repositories. The one method that showed me a way to use github repositories for jmeter.

PROCESS :-

After installing the performance plugin,

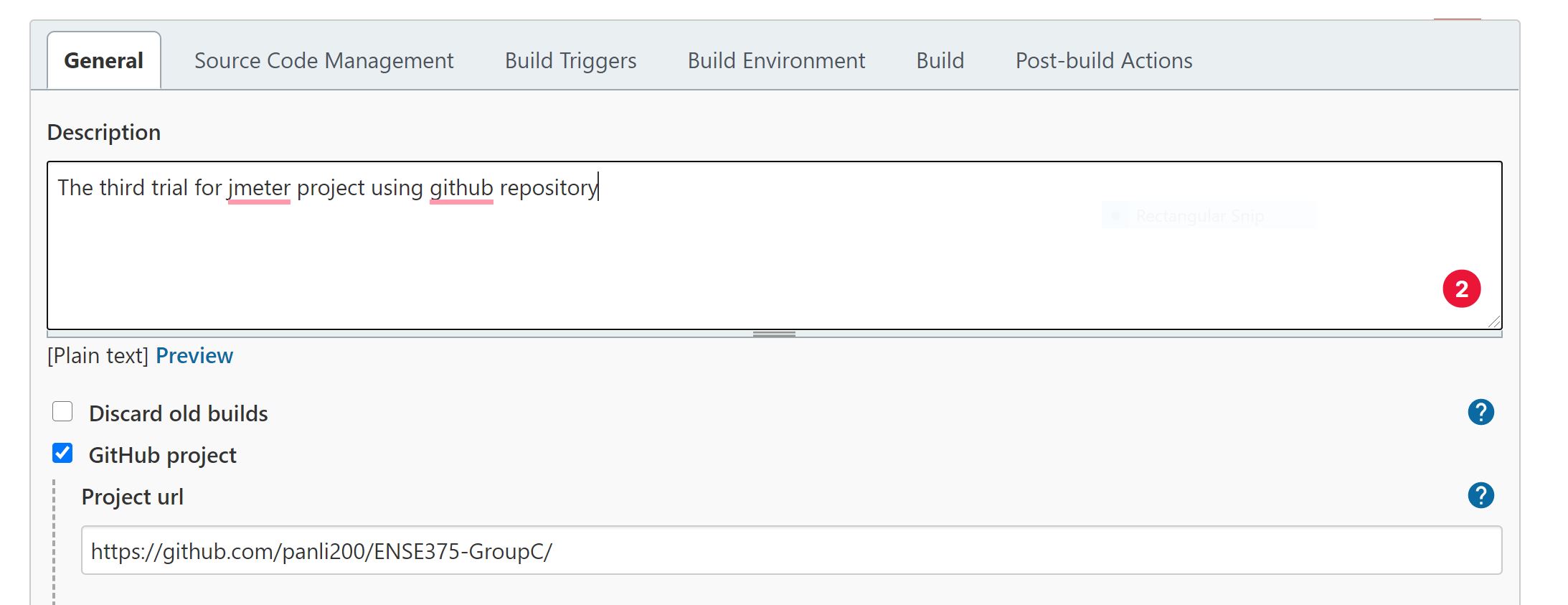
Step 1 : Create a new job on Jenkins homepage.

Step 2 : Click on the general tab, and then configure the general details for the project.

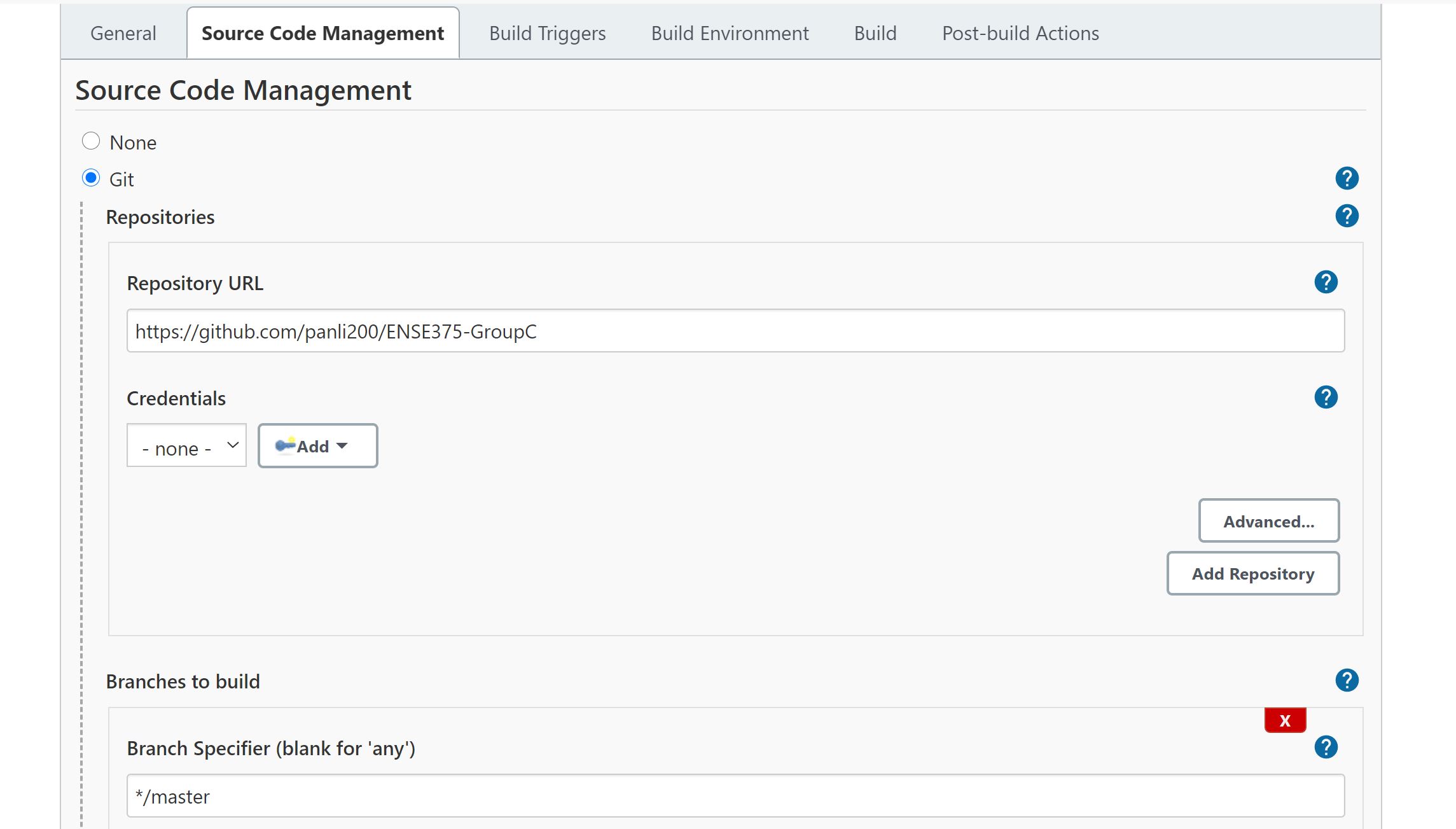
Set the project name

Give it a proper description

Step 3 : Tick the option for “github project”.



Step 4 : Under the source code management section, set the repository URL

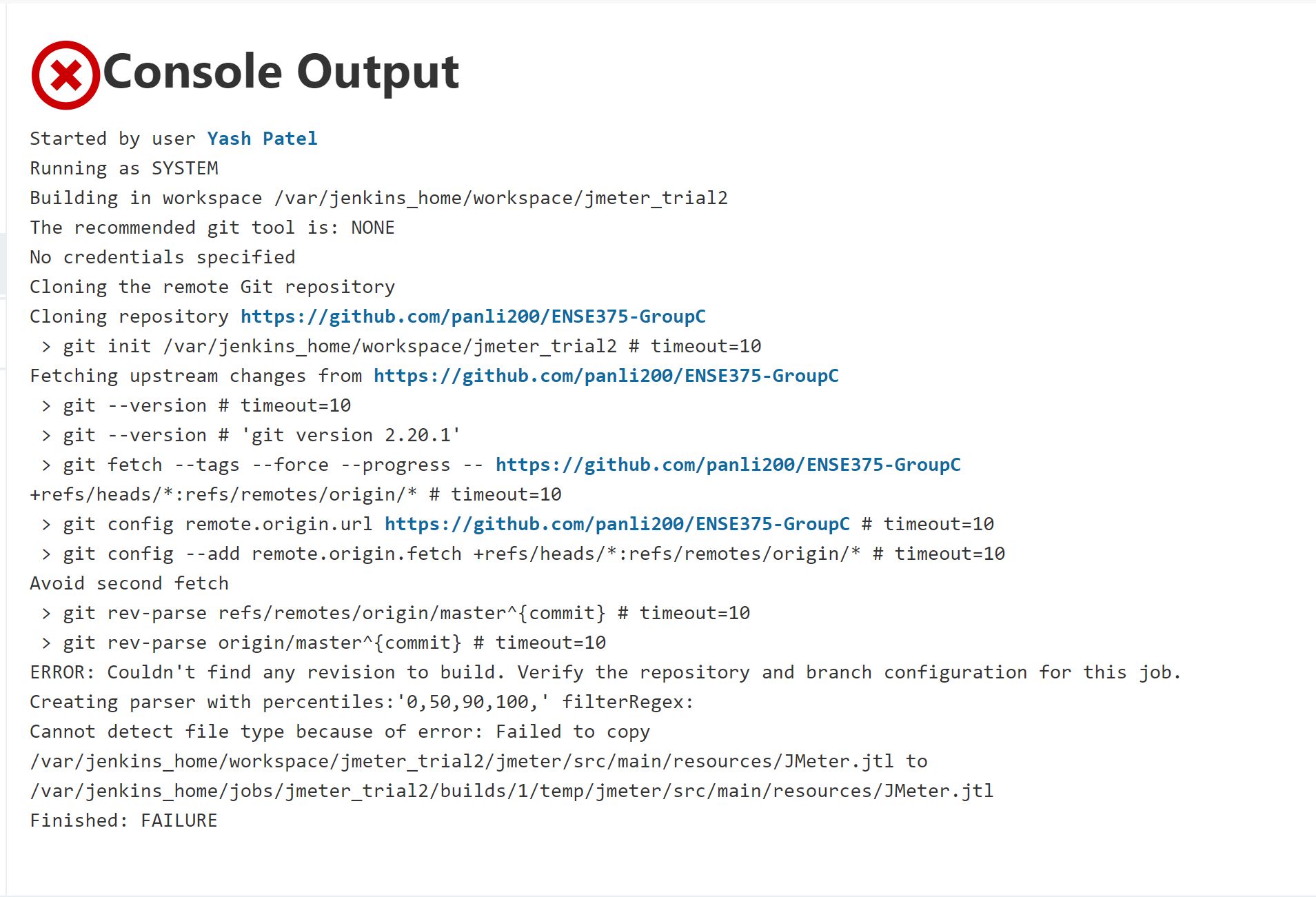


Step 5 : Now go to the *Build* *Tab* to specify how we'll build the project

Add a build step, and write a script under the ‘execute shell command” that will clone the repository and then carry out the testing when the project is built.

Step 6: Save the changes by clicking on “save” button.

The error that I received : -



Unfortunately, the cloning of the github is not working and I was unable to solve it. Since it was not cloned, I am also unable to perform the Jmeter. This problem was faced by me and my entire group, as none of the tutorials give a proper explanation on how to solve this issue.

Carter Issues Encountered:

When attempting to install Jmeter on my system, I tried to make it work both locally and on a repository instance. In both cases, I was unsuccessful in having a working build. I know a few of my group members were able to get a version of Jmeter to successfully build when the files were installed locally, I was unable to due an inability to work with the jenkins.io.jtl and .jmx files on my system. I would assume my issue resulted from the terminal commands provided in the online tutorials not being correct for the given situation.

Unfortunately for myself and all my group members, all the tutorials online never specified how to make Jmeter operate properly with the Jmeter files located on the repository.

References/Sources used by the group in an attempt to run Jmeter:

<https://www.jenkins.io/doc/book/using/using-jmeter-with-jenkins/>

<https://wiki.jenkins.io/display/JENKINS/How+to+run+JMeter+with+Jenkins>

<https://www.baeldung.com/jenkins-and-jmeter>

<https://ribblescode.wordpress.com/2012/04/16/how-to-automate-jmeter-tests-with-maven-and-jenkins-hudson-8/>

<http://blogs.quovantis.com/automating-jmeter-scripts-with-maven-and-jenkins/><https://www.blazemeter.com/blog/how-run-jmeter-test-jenkins-20-pipelines-and-github>