IT7320 – Software Development and Testing

Assignment 3

Jenkins

*Compiled October 2018*

*Aleisha Perry - 2115039*

For

*Chalinor Baliuag*

*Due Date: Week 11, Trimester 2, 2018*

# 

Contents

[Part 2 2](#_Toc527545146)

[Using Jenkins 2](#_Toc527545147)

[Appendix A 4](#_Toc527545148)

[GitHub 4](#_Toc527545149)

Table of Figures

[Figure 1 - Original build in Jenkins to test connection 2](#_Toc527545154)

[Figure 2 - All subsequent Jenkins builds not recognising 'mvn' as a command to run tests 3](#_Toc527545155)

[Figure 3 - Failure executing 'mnv' command with Jenkins 3](#_Toc527545156)

[Figure 4 - Jenkins connected to GitHub 3](#_Toc527545157)

[Figure 5 - Pointing to the pom.xml file in the maven project 3](#_Toc527545158)

# Part 2

## Using Jenkins

1. *What help/convenience does Jenkins give to your development and testing processes?*

Jenkins allows testing to take place from a shared version controlled repository everyone can access, showing all changes made to the code. It also allows automated testing. As Jenkins is a continuous integration software tool, it comes with the more agile approach and speeds up the build cycles by encouraging small changes over large changes to code.

1. *Identify some advantages when implementing continuous integration using Jenkins.*

* Installation is easier compared to other software that provides continuous integration services.
* Can clearly see if a build has passed or any of the alternatives.
* Provides a command line results window to check where the tests have gone wrong.
* Continuously check the code compiles.
* Can run a set of basic tests that verify the code after every commit.

1. *Now create a simple Dynamic Web Application in Maven and link that to Jenkins.*

I had a lot of trouble with this step. My maven project was linked to Jenkins, and it ran the first time I compiled the test, but ever since that step it has stopped recognising ‘mvn’ as a command.

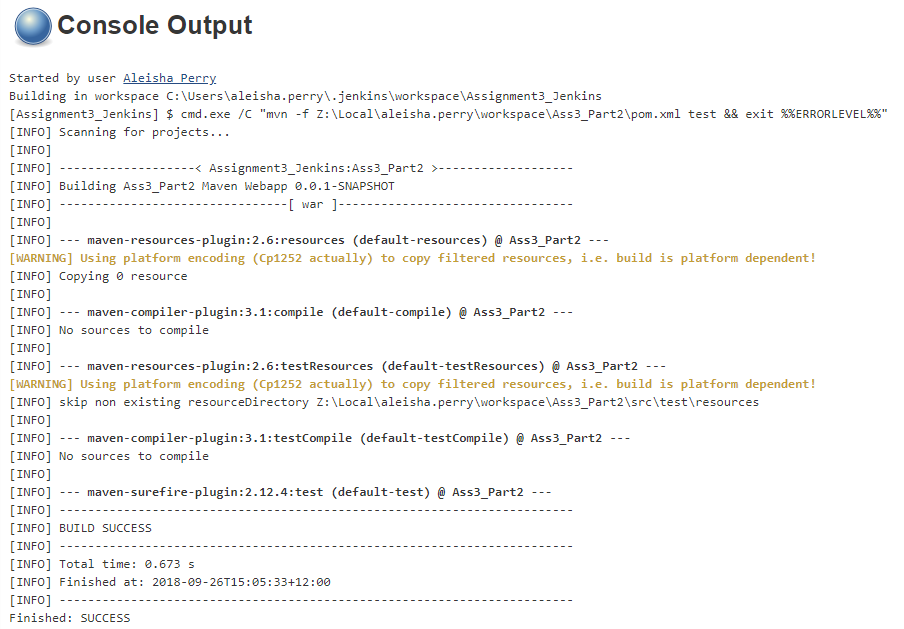


Figure 1 - Original build in Jenkins to test connection

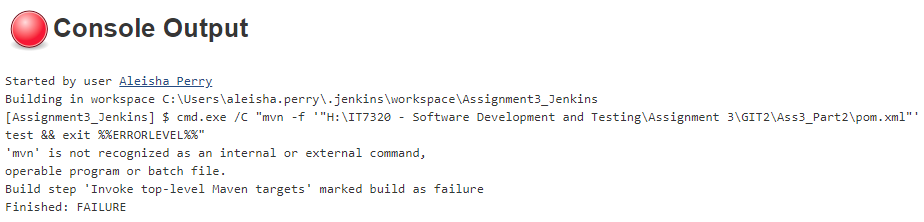


Figure 2 - All subsequent Jenkins builds not recognising 'mvn' as a command to run tests

The following image shows the console output in eclipse, proving that my maven project is linked to Jenkins, but it fails to execute.



Figure 3 - Failure executing 'mnv' command with Jenkins

Jenkins linked to my GitHub:

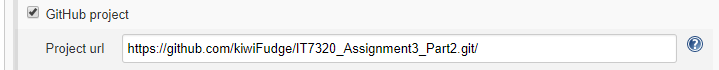


Figure 4 - Jenkins connected to GitHub

Jenkins project configuration:

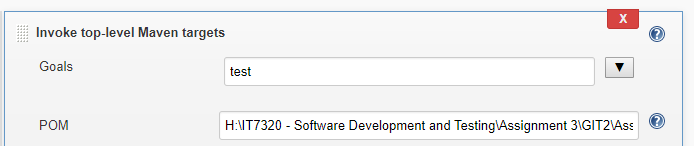


Figure 5 - Pointing to the pom.xml file in the maven project

# Appendix A

## GitHub

GitHub was used for the duration of this assignment in order to submit assignment in stages (Parts 1 - 4).

**Profile - Aleisha**

Remote Repository main branch (owner) link: <https://github.com/kiwiFudge/IT7320_Assignment3_Part2.git/>