



Selenium Maven plugin

Requirements document

Date: 2013.01.02

feedzai[↑]

AUTHORS AND CONTRIBUTORS

Code	Name	Contact
AK	Andrey Klimachev	andrey.klimachev@feedzai.com

DOCUMENT APPROVAL

Version	Date	Approved By	Role

DISTRIBUTION LIST

SVP Product Development

COPYRIGHT

All rights reserved. No part of this document may be reproduced or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior written permission of FeedZai S.A.

TABLE OF CONTENTS

1	Purpose.....	5
2	Introduction	5
	2.1 Overview	5
	2.2 Requirement Structure	7
	2.3 Requirements.....	7
3	Proposed Implementation	9
	3.1 Configuration.....	9
	3.2 Type specification	11
	3.3 Example	11

GLOSSARY

Acronym	Description

TABLE OF TABLES

Table 1 – Requirement SE-MV-01 7

Table 2 - Requirement SE-MV-02 7

Table 3 - Requirement SE-MV-03 7

Table 4 - Requirement SE-MV-04 7

Table 5 - Requirement SE-MV-05 8

Table 6 - Requirement SE-MV-06 8

Table 7 - Requirement SE-MV-07 8

Table 8 - Requirement SE-MV-08 8

Table 9 - Requirement SE-MV-09 8

Table 10 - Requirement SE-MV-10 9

Table 11 – Requirement SE-MV-11 9

Table 12 – Requirement SE-MV-12 9

TABLE OF FIGURES

Figure 1 – Overview of the maven test phase..... 6

Figure 2 - Available tags in the maven plugin configuration 10

Figure 3 - Configuration example for the plugin 12

1 Purpose

This document describes the requirements for a Selenium Maven plugin and provides possible structure of the plugin.

Currently, the configuration of the Selenium Remote Controls is very complex. For example, if it is needed to add a new Selenium Remote Control, that control needs to be replicated in several places of the Project Object Model file. The purpose of development of this maven plugin is to make the configuration smaller and generic.

The principal objective of the Maven plugin is configuring the Pulse test environments for executing the functional tests in Selenium Grid.

2 Introduction

2.1 Overview

On a very high level the maven test phase can be described by Figure 1.

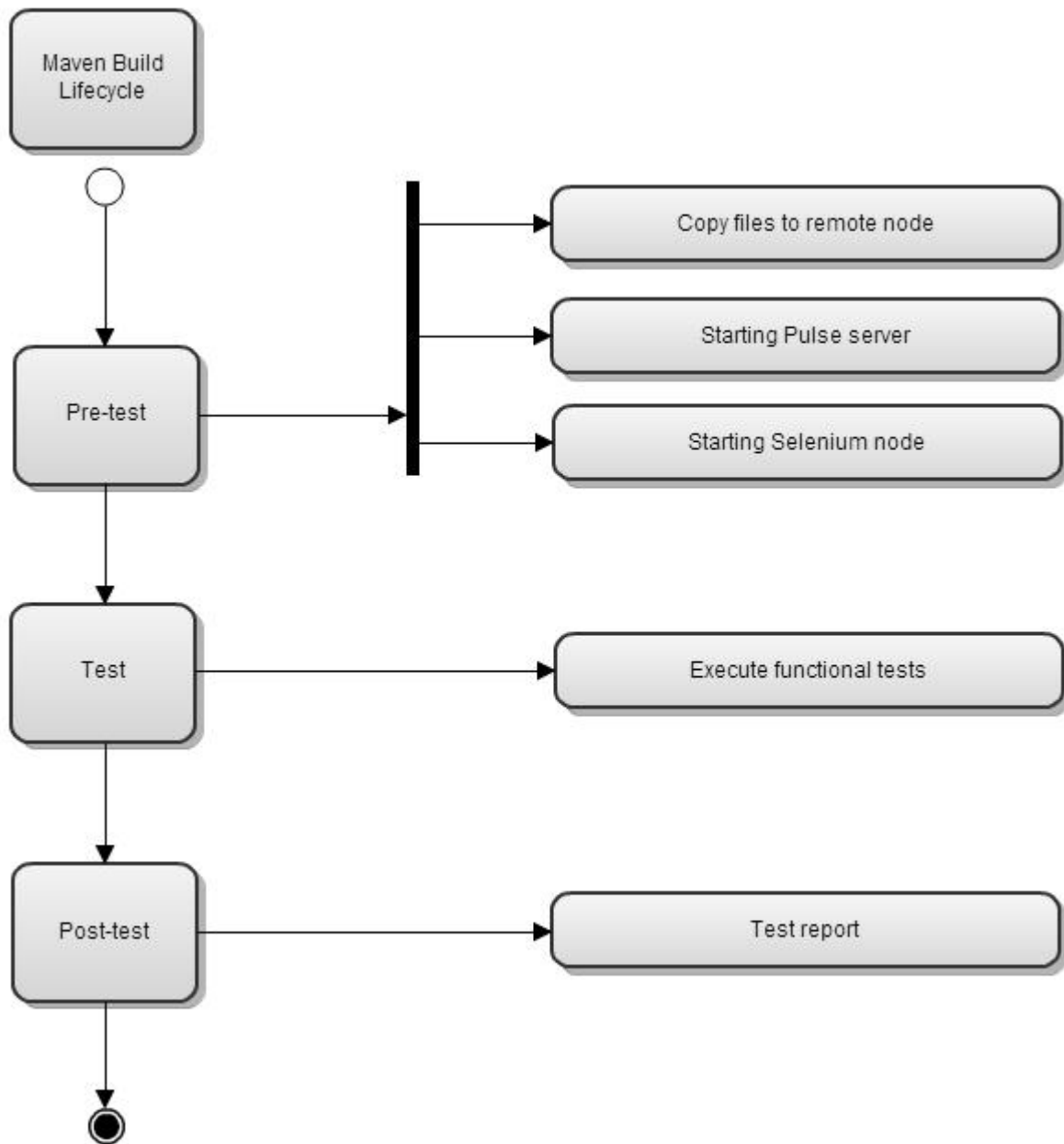


Figure 1 – Overview of the maven test phase

The 'pre-test' phase is responsible for preparing the environment for execution of the functional tests of PulseViews. The 'test' phase is responsible for the execution of the Selenium tests on a configured environment. The 'Post-test' phase is responsible for returning the test report.

The objective of the maven plugin is to prepare the remote machines for the execution of the functional tests by copying the files and executing the commands.

2.2 Requirement Structure

Each requirement has the following fields:

- **ID** – Unique identifier of the requirement
- **Name** – Name of the requirement
- **Priority** – Priority of the requirement
- **Description** – Brief description of the requirement

2.3 Requirements

Field	Description
ID	SE-MV-01
Name	Support operating systems (Windows, Linux)
Priority	Must
Description	Plugin should be able to connect to operating systems (Windows, Linux)

Table 1 – Requirement SE-MV-01

Field	Description
ID	SE-MV-02
Name	Support only one operating system at the same time
Priority	Must
Description	Plugin should only support one type of operating system at the same time

Table 2 - Requirement SE-MV-02

Field	Description
ID	SE-MV-03
Name	Create directories on remote machine
Priority	Must
Description	Plugin should be able to create directories on the remote machine

Table 3 - Requirement SE-MV-03

Field	Description
ID	SE-MV-04
Name	Replace directories on remote machine
Priority	Must
Description	Plugin should be able to replace directories on the remote machine if this already exist

Table 4 - Requirement SE-MV-04

Field	Description
-------	-------------

ID	SE-MV-05
Name	Pack/Unpack files
Priority	Must
Description	Plugin should be able to pack locally and unpack remotely the files

Table 5 - Requirement SE-MV-05

Field	Description
ID	SE-MV-06
Name	Copy files
Priority	Must
Description	Plugin should be able to copy one or several files to the remote machine

Table 6 - Requirement SE-MV-06

Field	Description
ID	SE-MV-07
Name	Support for copy any file
Priority	Must
Description	Plugin should be able to copy any file to the remote machine

Table 7 - Requirement SE-MV-07

Field	Description
ID	SE-MV-08
Name	Execute BASH command
Priority	Must
Description	Plugin should be able to execute Unix command on the remote machine

Table 8 - Requirement SE-MV-08

Field	Description
ID	SE-MV-09
Name	Execute several bash commands for each machine
Priority	Must
Description	Plugin should be able to execute more than one command for each machine

Table 9 - Requirement SE-MV-09

Field	Description
ID	SE-MV-10
Name	Accept a specific command for configuration of the Selenium node
Priority	Must

Description	Plugin should accept only one specific command for configuration of the Selenium node
-------------	---

Table 10 - Requirement SE-MV-10

Field	Description
ID	SE-MV-11
Name	Supported Maven version
Priority	Must
Description	Plugin should support Maven version 3 or greater.

Table 11 – Requirement SE-MV-11

Field	Description
ID	SE-MV-12
Name	Pulse server folder
Priority	Must
Description	Should be possible to specify the Pulse server folder.

Table 12 – Requirement SE-MV-12

3 Proposed Implementation

3.1 Configuration

You can configure the plugin by specifying some tags as shown below.

```
<configuration>
  <linux>
    <path>...</path>
    <username>...</username>
    <password>...</password>
    <nodes>
      ...
    </nodes>
    <testBehaviour>
      <copyFiles>
        ...
      </copyFiles>
      <commands>
        ...
      </commands>
      <nodeLauncher>...</nodeLauncher>
    </testBehaviour>
  </linux>
  <windows>
    <path>...</path>
    <username>...</username>
    <password>...</password>
    <nodes>
      <node>...</node>
      <node>...</node>
      <node>...</node>
    </nodes>
    <testBehaviour>
      <copyFiles>
        ...
      </copyFiles>
      <commands>
        ...
      </commands>
      <nodeLauncher>...</nodeLauncher>
    </testBehaviour>
  </windows>
</configuration>
```

Figure 2 - Available tags in the maven plugin configuration

- **<linux>** and **<windows>** specify the operating system.
- **<path>** specifies the directory where the files are copied and where the command will be executed.
- **<username>** and **<password>** are needed to authenticate on the remote machine.

- **<node>** specifies the url of the remote machine.
- **<copyFile>** specifies the path of the file that will be copied.
- **<commnad>** specifies the command that will be executed.
- **<nodeLauncher>** specifies the command for starting the Selenium node on the remote machine.
- **<pulseServerFolder>** specifies the folder of the Pulse

3.2 Type specification

This section intends to explain each tag of the plugin.

Name	Optional	Description
<linux>	true	Identifies the operating system (Linux).
<windows>	true	Identifies the operating system (Windows).
<path>	true	Defines the path where the files are copied and the commands are executed.
<username>	true	Username for authentication on the remote machine.
<password>	true	Password for authentication on the remote machine.
<nodes>	true	Contains list of <node>
<node>	true	Defines the url of remote machine.
<copyFiles>	false	Contains list of <copyFile>
<commands>	false	Contain list of <command>.
<command>	false	The command that is executed on each remote machine of <nodes>.
<nodeLauncher>	false	The command for starting the Selenium node on remote machine.
<pulseServerFolder>	true	The command to specify the pulse server folder.

3.3 Example

The follow example illustrates a possible configuration for the plugin.

```

<plugin>
  <groupId>com.feedzai.selenium</groupId>
  <artifactId>feedzai-selenium-maven-plugin</artifactId>
  <version>0.0.1</version>
  <configuration>
    <linux>
      <path>/opt</path>
      <username>username</username>
      <password>password</password>
      <nodes>
        <node>test-selenium-linux-1.zai</node>
        <node>test-selenium-linux-2.zai</node>
        <node>test-selenium-linux-3.zai</node>
      </nodes>
      <testBehaviour>
        <copyFiles>
          <copyFile>(project.basedir)/Pulse.zip</copyFile>
          <copyFile>$(project.basedir)/src/test/resources/chromedriver</copyFile>
          <copyFile>$(project.basedir)/src/test/resources/selenium-server-standalone-2.25.0.jar</copyFile>
          <copyFile>$(project.basedir)/src/test/resources/database-drivers/com.microsoft.sqlserver.jdbc-3.0.jar</copyFile>
          <copyFile>$(project.basedir)/src/test/resources/database-drivers/mysql-connector-java-5.1.17.jar</copyFile>
          <copyFile>$(project.basedir)/src/test/resources/database-drivers/ojdbc6-11.2.0.2.0.jar</copyFile>
        </copyFiles>
        <commands>
          <command>jps | grep PulseManagerImpl | awk '{print $1}' | xargs kill -9</command>
          <command>jps | grep selenium-server-standalone | awk '{print $1}' | xargs kill -9</command>
          <command>rm -rf /home/pulse/</command>
          <command>rm -rf /home/output.txt</command>
          <command>rm -Rf ~/.Trash/*</command>
          <command>unzip Pulse.zip -d /home/pulse/</command>
          <command>echo "services.modules.kpi.batchflusher.size=10" >> /home/pulse/conf/pulse.properties</command>
          <command>chmod 777 /home/pulse/bin/startup.sh</command>
          <command>chmod 777 /home/chromedriver</command>
          <command>mv com.microsoft.sqlserver.jdbc-3.0.jar /home/pulse/lib/</command>
          <command>mv mysql-connector-java-5.1.17.jar /home/pulse/lib/</command>
          <command>mv ojdbc6-11.2.0.2.0.jar /home/pulse/lib/</command>
          <command>dos2unix /home/pulse/bin/startup.sh</command>
          <command>sed -i 's/MaxPermSize=128m/MaxPermSize=256m/g' /home/pulse/bin/startup.sh</command>
          <command>nohup sh /home/pulse/bin/startup.sh >> /home/pulse/log/output.txt &</command>
        </commands>
      </testBehaviour>
    </linux>
  </configuration>
</plugin>

```

Figure 3 - Configuration example for the plugin