**Test suite: Maven projects**

* The following test scenario is not available anymore:

1. Create Quickstart archetypeGo to main menu File > New ProjectIn New Project wizard select Maven > Maven Project and push Next

2. Create Web Application

TESTED OK (Java EE 5 added jsp-api-2.1.jar and servlet-api-2.5.jar with provided scope, Java EE 6 added javaee-web-api-6.0.jar  with provided scope)

3. Create EJB Module

TESTED OK (Java EE 5 added javaee-api-5.jar with provided scope, Java EE 6 added javaee-api-6.0.jar with provided scope)

4. Create Enterprise Application

"There's missing dependencies warning icon in Projects tab. Do build and this will disappear." I Don't see this icon

TESTED OK

5. Create NetBeans Module archetype

* I couldn't find "Select Maven NetBeans Module Archetype and push Next", but I assumed it to be ""NetBeans Module", and continued the test with this assumption.

* A choise about "Allow OSGi bundles as dependencies". It was not checked as default, and I let it be this way.

***RESULT***

* <misconfigued Maven project in NBModulePrj>

* Complaining about missing dependency: org.netbeans.api:org-netbeans-api-annotations-common:jar:RELEASE70-BETA

* Stacktrace:

2 problems were encountered while building the effective model for com.mycompany:NBModulePrj:1.0-SNAPSHOT

[ERROR] Unresolveable build extension: Plugin org.codehaus.mojo:nbm-maven-plugin:3.5 or one of its dependencies could not be resolved: The repository system is offline but the artifact org.codehaus.mojo:nbm-maven-plugin:jar:3.5 is not available in the local repository. @

[ERROR] Unknown packaging: nbm @ line 8, column 16

for project com.mycompany:NBModulePrj:1.0-SNAPSHOT at /home/viggo/NetBeansProjects/NBModulePrj/pom.xml

* Doesn't build

TESTED NOT OK

6. Create NetBeans Platform Application archetype

* I couldn't find "Select Maven NetBeans Platform Application Archetype and push Next", but I assumed it to be "NetBeans Application",  and continued the test with this assumption.

* A choise about "Allow OSGi bundles as dependencies". It was not checked as default, and I let it be this way.

* A choise about "Create Module Project: NBAppPrj-sample". It was not checked as default, and I let it be this way.

***RESULT***

*** <misconfigued Maven project in application>**

* Complaining about missing dependency: org.netbeans.api:org-netbeans-api-annotations-common:jar:RELEASE70-BETA

* Doesn't build

TESTED NOT OK

**Test suite: Clean, Build, Run, Debug maven project**

1. Clean project

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!), but I did a test on the Web project instead.

TESTED OK (both Java EE 5 and 6)

1. Build project

*** Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!), but I did a test on the Web project instead.**

***RESULT***

Artifact is built and put in target, but doesn't contain the version number in the file name, QuickstartPrj.war instead of QuickstartPrj-1.0-SNAPSHOT.war

TESTED NOT OK (both Java EE 5 and 6)

1. Run project

*** Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!)**

TESTED NOT OK

**1. Debug project**

***  Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!)**

TESTED NOT OK

**Test suite: Run, Debug, Profile, Test single file**

1. Run file

* "Select Glassfish v3 server in next window and click OK". It's GlassFish Server 3.1 which is bundled with Netbeans 7.0 Beta 2.

* The servlet is displayed in browser, but since all code in processRequest of the generated servlet is commented out(!) the page is empty. I removed the comment and ran it once more, then the text "

**Servlet MavenPrjServlet at /com.mycompany_QuickstartPrj_JavaEE5_war_1.0-SNAPSHOT**

" was shown.

TESTED OK (both Java EE 5 and 6)

2. Debug file

**TESTED OK (both Java EE 5 and 6)**

**3. Profile file**

* When trying to profile it I got a popup window with the message "The Java Platform this project

runs on cannot be used for profiling. Please selevt Java Platform to use: JDK 1.6 (Default).". I clicked OK and continued, and then I got this in the ouput window:

```
NetBeans: Deploying on GlassFish Server 3.1
    profile mode: true
    debug mode: false
    force redeploy: true
Stopping the profiler
Profiled server stopped
Server GlassFish Server 3.1 is stopping
GlassFish Server 3.1 was stopped.
Starting GlassFish Server 3.1
GlassFish Server 3.1 is running.
In-place deployment at /home/viggo/NetBeansProjects/QuickstartPrj_JavaEE5/target/QuickstartPrj_JavaEE5
deploy?DEFAULT=/home/viggo/NetBeansProjects/QuickstartPrj_JavaEE5/target/QuickstartPrj_JavaEE5&name=com.mycompany_QuickstartPrj_JavaEE5_war_1.0-SNAPSHOT&force=true failed. Instance is not GlassFish Server 3.1.
The module has not been deployed.
at org.netbeans.modules.j2ee.deployment.devmodules.api.Deployment.deploy(Deployment.java:187)
at org.netbeans.modules.maven.j2ee.ExecutionChecker.performDeploy(ExecutionChecker.java:167)
at org.netbeans.modules.maven.j2ee.ExecutionChecker.executionResult(ExecutionChecker.java:123)
at org.netbeans.modules.maven.execute.MavenCommandLineExecutor.run(MavenCommandLineExecutor.java:208)
at org.netbeans.core.execution.RunClassThread.run(RunClassThread.java:154)
```

TESTED NOT OK (both Java EE 5 and 6)

**4. Test file**

* I wasn't able to use the exact steps from the script, had to right-click the Project->JUnit->Test for Existing Class

* When I ran the test I got this in the output window:

```
Testsuite: org.mavenprj.MavenPrjServletTest
processRequest
doGet
doPost
getServletInfo
Tests run: 4, Failures: 0, Errors: 4, Time elapsed: 0.072 sec
```

```
------------- Standard Output ---------------
processRequest
doGet
doPost
getServletInfo
------------- ---------------- ---------------
Testcase: testProcessRequest(org.mavenprj.MavenPrjServletTest): Caused an ERROR
Absent Code attribute in method that is not native or abstract in class file
javax/servlet/http/HttpServlet
java.lang.ClassFormatError: Absent Code attribute in method that is not native or abstract in class
file javax/servlet/http/HttpServlet
at java.lang.ClassLoader.defineClass1(Native Method)
at java.lang.ClassLoader.defineClassCond(ClassLoader.java:632)
at java.lang.ClassLoader.defineClassCond(ClassLoader.java:632)
at java.lang.ClassLoader.defineClass(ClassLoader.java:616)
at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:141)
at java.net.URLClassLoader.defineClass(URLClassLoader.java:283)
at java.net.URLClassLoader.access$000(URLClassLoader.java:58)
at java.net.URLClassLoader$1.run(URLClassLoader.java:197)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader.findClass(URLClassLoader.java:190)
at java.lang.ClassLoader.loadClass(ClassLoader.java:307)
at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:301)
at java.lang.ClassLoader.loadClass(ClassLoader.java:248)
at java.lang.ClassLoader.defineClass1(Native Method)
at java.lang.ClassLoader.defineClassCond(ClassLoader.java:632)
at java.lang.ClassLoader.defineClass(ClassLoader.java:616)
at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:141)
at java.net.URLClassLoader.defineClass(URLClassLoader.java:283)
at java.net.URLClassLoader.access$000(URLClassLoader.java:58)
at java.net.URLClassLoader$1.run(URLClassLoader.java:197)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader.findClass(URLClassLoader.java:190)
at java.lang.ClassLoader.loadClass(ClassLoader.java:307)
at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:301)
at java.lang.ClassLoader.loadClass(ClassLoader.java:248)
at org.mavenprj.MavenPrjServletTest.testProcessRequest(MavenPrjServletTest.java:50)
```

```
Testcase: testDoGet(org.mavenprj.MavenPrjServletTest): Caused an ERROR
org/mavenprj/MavenPrjServlet
java.lang.ClassFormatError: org/mavenprj/MavenPrjServlet
at org.mavenprj.MavenPrjServletTest.testDoGet(MavenPrjServletTest.java:64)
Testcase: testDoPost(org.mavenprj.MavenPrjServletTest): Caused an ERROR
org/mavenprj/MavenPrjServlet
java.lang.ClassFormatError: org/mavenprj/MavenPrjServlet
at org.mavenprj.MavenPrjServletTest.testDoPost(MavenPrjServletTest.java:78)
Testcase: testGetServletInfo(org.mavenprj.MavenPrjServletTest): Caused an ERROR
org/mavenprj/MavenPrjServlet
java.lang.ClassFormatError: org/mavenprj/MavenPrjServlet
at org.mavenprj.MavenPrjServletTest.testGetServletInfo(MavenPrjServletTest.java:90)
Test org.mavenprj.MavenPrjServletTest FAILED
test:
Deleting: /tmp/TEST-org.mavenprj.MavenPrjServletTest.xml
BUILD SUCCESSFUL (total time: 1 second)
```

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!)

**Test suite: Project properties**

1. Test General properties

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!), but I did a test on the Web project instead.

TESTED OK (both Java EE 5 and 6)

1. Test Sources properties

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!),but I did a test on the Web project instead.

TESTED OK (both Java EE 5 and 6)

1. Test Configurations properties

* nb-configurations.xml didn't contain the id nor the actilve profiles!

TESTED NOT OK (both Java EE 5 and 6)

1. Test Compile properties

TESTED OK (both Java EE 5 and 6)

1. Test Configurations

* Didn't understand this!

1. Test Actions properties

TESTED OK (both Java EE 5 and 6)

1. Test Formatting properties

TESTED OK (both Java EE 5 and 6)

**Test suite: Create templates in maven projects**

1. Create package

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!), but I did a test on the Web project instead.

**TESTED OK (both Java EE 5 and 6)**

**1. Create JFrame form**

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!)

1. Create XML document

* Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!), but I did a test on the Web project instead.

**TESTED OK (both Java EE 5 and 6)**

**1. Create new Java class**

*** Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!), but I did a test on the Web project instead.**

**TESTED OK (both Java EE 5 and 6)**

**Test suite: Refactoring**

1. Rename package

**TESTED OK (both Java EE 5 and 6)**

**1. Rename java class**

**TESTED OK (both Java EE 5 and 6)**

**1. Rename JFrame form**

*** Unable to test QuickstartPrj project since this wasn't an option in the Maven menu (see above!)**

**Test suite: Project operations**

1. Rename project

**TESTED OK (both Java EE 5 and 6)**

**1. Move project**

* Problems
- Unable to properly load project:

1 problem was encountered while building the effective model

[FATAL] Non-readable POM /home/viggo/NetBeansProjects/tmp/QuickstartPrj_JavaEE5/pom.xml: input contained no data @

for project  at /home/viggo/NetBeansProjects/tmp/QuickstartPrj_JavaEE5/pom.xml

- Broken nb-configuration.xml file:

The $project_basedir/nb-configuration.xml file cannot be parsed. The information contained in the file will be ignored until fixed. This affects several features in the IDE that will not work properly as a result.

The parsing exception follows:

Premature end of file.

**TESTED OK (Java EE 6)**

**TESTED NOT OK (Java EE 5)**

**1. Delete project**

**TESTED OK (both Java EE 5 and 6)**

**1. Delete project along with sources**

**TESTED OK (both Java EE 5 and 6)**

**Test suite: Maven related options**

**1. Set up External Maven Home**

**TESTED OK**

**2. Set up Skip Tests**

**TESTED OK**

**3. Set up Global Options**

**TESTED OK**

**4. Set up frequence of download**

**TESTED OK**

**5. Edit Custom Global Definitions**

*** Not to detailed test, should more be added?**

**TESTED OK**

6. Set up frequence of index updates

**TESTED OK**

**7. Maven related editor hints**

TESTED OK

**Test suite: Project libraries**

1. Add library

* There is no node called "Library", it's called "Dependencies" or "Test dependencies". Assume Library=Dependency and continue testing with this assumption

* I suggest adding more test on different scopes to see that they are generated properly!

TESTED OK

**2. Show Dependency Graph**

*** There is no node called "Library", it's called "Dependencies" or "Test dependencies". Assume Library=Dependency and continue testing with this assumption**

*** There is no "Show dependency Graph" option available.**

*** As far as I know, you can only show dependency graph by opening the pom.xml, right-click and choose "Show Dependency Graph"**

**TESTED OK (but test script must be updated!!)**

**3. Download Missing Dependencies**

*** There is no node called "Library", it's called "Dependencies" or "Test dependencies". Assume Library=Dependency and continue testing with this assumption**

NOT TESTED, didn't find any dependencies I didn't have in my repository:)

4. Download Javadoc

*** There is no node called "Library", it's called "Dependencies" or "Test dependencies". Assume Library=Dependency and continue testing with this assumption**

Not sure how to verify this. I didn't get any errors, but the test script doesn't explicitly tell what/where to check!

NOT TESTED

5. Download Sources

*** There is no node called "Library", it's called "Dependencies" or "Test dependencies". Assume Library=Dependency and continue testing with this assumption**

**Not sure how to verify this. I didn't get any errors, but the test script doesn't explicitly tell what/where to check!**

**NOT TESTE**D

This was all I had time to test today,will continue with rest of the script tomorrow!

* Create Web application and EJB Module were missing information about which Java EE Version to choose. I think this is vital to give input on, since I assume it will have consequences for which dependencies are generated to the pom.xml files. I also think that the verification should include checking that the correct dependencies are generated, Java EE 5 vs Java EE 6. I ran each test with both Java EE 5 and Java EE 6.