



UNIVERSIDAD DE OVIEDO

ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN

MÁSTER EN INGENIERÍA INFORMÁTICA

TRABAJO FIN DE MÁSTER

SPRING ROO ADD-ONS PARA PROTOTIPADO RÁPIDO



JAVIER MENÉNDEZ ÁLVAREZ
JULIO 2014



UNIVERSIDAD DE OVIEDO

ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN

MÁSTER EN INGENIERÍA INFORMÁTICA

TRABAJO FIN DE MÁSTER

SPRING ROO ADD-ONS PARA PROTOTIPADO RÁPIDO

DOCUMENTO Nº VI

PRUEBAS



JAVIER MENÉNDEZ ÁLVAREZ
JULIO 2014

**ÁREA DE CIENCIAS DE LA
COMPUTACIÓN E INTELIGENCIA
ARTIFICIAL**
TUTOR: M^a JOSÉ SUÁREZ CABAL

Índice

Introducción.....	7
Pruebas unitarias.....	8
1. Clase CreateModules.java.....	8
2. Clase FileUtilities.java.....	9
3. Clase PomUtils.java.....	12
4. Clase ProjectUtils.java.....	18
5. Clase WebModuleUtils.java.....	19
6. Clase XmlUtils.java.....	26
Fallos detectados en pruebas unitarias.....	33
Fallo 1.....	33
Fallo 2.....	33
Fallo 3.....	33
Fallo 4.....	33
Fallo 5.....	33
Fallo 6.....	34
Fallo 7.....	34
Pruebas de sistema.....	35
Situaciones de prueba.....	35
Casos de prueba.....	36
Caso de prueba 1.....	36
Caso de prueba 2.....	36
Caso de prueba 3.....	37
Caso de prueba 4.....	37
Caso de prueba 5.....	38
Caso de prueba 6.....	39
Caso de prueba 7.....	40
Caso de prueba 8.....	40
Caso de prueba 9.....	41
Caso de prueba 10.....	43
Caso de prueba 11.....	44
Caso de prueba 12.....	45
Caso de prueba 13.....	46
Caso de prueba 14.....	47
Caso de prueba 15.....	47
Caso de prueba 16.....	48
Caso de prueba 17.....	50
Caso de prueba 18.....	51
Caso de prueba 19.....	52
Caso de prueba 20.....	52
Caso de prueba 21.....	53
Caso de prueba 22.....	54
Caso de prueba 23.....	54
Fallos detectados en pruebas de sistema.....	55
Fallo 1.....	55
Fallo 2.....	55

Fallo 3.....	55
Fallo 4.....	55
Conclusiones.....	60
ANEXO I: clases de Test utilizadas en las pruebas unitarias.....	61
CreateModulesTest.java.....	61
FileUtilitiesTest.java.....	64
PomUtilsTest.java.....	69
ProjectUtilsTest.java.....	79
WebModuleUtilsTest.java.....	82
XmlUtilsTest.java.....	90
ANEXO II: Recursos del JAR de Roo++.....	99
dirs.txt.....	99
pom.xml.....	100
textfiles.txt.....	100
core.....	101
pom.xml.....	101
docs.....	101
pom.xml.....	101
src/InstallationManual.xml.....	102
src/UserManual.xml.....	103
src/assembly/package.xml.....	103
ear.....	103
pom.xml.....	103
src/main/application/META-INF/weblogic-application.xml.....	104
package.....	104
pom.xml.....	104
src/assembly/package.xml.....	105
scripts.....	105
pom.xml.....	105
src/assembly/compress-zip.xml.....	106
web.....	106
plugin-license.xml.....	106
plugin-source.xml.....	106
resource-ref.xml.....	107
src/main/jetty/jetty-database.xml.....	107
src/main/jetty/jetty.xml.....	107
src/main/jetty/etc/realm.properties.....	109
src/main/jetty/etc/webdefault.xml.....	110
src/main/resources/META-INF/spring/application-properties-bean.xml.....	114
src/main/resources/META-INF/spring/application.properties.....	114
src/main/resources/META-INF/spring/applicationContext-security.xml.....	115
src/main/resources/META-INF/spring/database.properties.....	115
src/main/resources/META-INF/spring/entity-manager-factory.xml.....	116
src/main/resources/META-INF/spring/jndi-datasource.xml.....	116
src/main/resources/META-INF/spring/message-source.xml.....	116
src/main/resources/META-INF/spring/property-placeholder-configurer.xml.....	116
src/main/tomcat/context.xml.....	116
src/main/tomcat/server.xml.....	116
src/main/webapp/WEB-INF/casfailure-view.xml.....	116

src/main/webapp/WEB-INF/views/casfailure.jspx.....	117
ANEXO III: Contenido del directorio de configuración.....	117
applicationContext-platfowm.xml.....	117
applicationContext-portal.xml.....	118
breadcrumb-dependency.xml.....	118
breadcrumbs-beans.xml.....	118
breadcrumbs-url.txt.....	118
context-dataload.sql.....	118
context-dependencies.txt.....	118
context-messages.properties.....	118
context-portal.properties.....	118
licenses.xml.....	119
locations-application-properties.xml.....	119
messages.properties.....	119
mssql.txt.....	119
mysql.txt.....	119
oracle.txt.....	119
oracle-sid.txt.....	119
parent.txt.....	119
platform.txt.....	119
plugin-license.xml.....	119
portal-header.jspx.....	120
portal-url.txt.....	120
portal.properties.....	120
portal.txt.....	120
postgresql.txt.....	120
usermgmt.txt.....	120
usermgmt_application.properties.....	121
usermgmt_users.txt.....	121
contextBeans/portal/nodesToAdd/nodeToAdd.txt.....	121
contextBeans/portal/portal_bean.xml.....	121
contextBeans/security/authenticationSuccessHandler.txt.....	121
contextBeans/security/security_bean_1.xml.....	121
contextBeans/security/security_bean_2.xml.....	121
contextHeader/add-to-portal-header.txt.....	121
contextHeader/directive-pages.jspx.....	121
contextHeader/scriptlets.jspx.....	122
contextViews/change.jspx.....	122
contextViews/create.jspx.....	122
contextViews/list.jspx.....	122
contextViews/show.jspx.....	122
contextViews/update.jspx.....	123
doc/docbkx/custom.xsl.....	123
doc/fonts.xml.....	126
layout/load-scripts.jspx.....	127
layout/classes/estilo.properties.....	127
layout/classes/style.properties.....	127
layout/layouts/alreves.jspx.....	127
layout/scripts/script-esp.js.....	127

layout/scripts/script.js.....	127
layout/styles/estilo.css.....	127
layout/styles/style.css.....	131
layout/tags/box/roundbox.tagx.....	135
usermgmt/application.properties.....	135
usermgmt/applicationContext-security.xml.....	136
usermgmt/artifactItems.txt.....	136
usermgmt/changePassword.jspx.....	136
usermgmt/userInfo.jspx.....	136
usermgmt/usermgmt.txt.....	136

Introducción

En este documento se trata el Plan de Pruebas llevado a cabo con el add-on para Spring Roo, Roo++. El Plan de Pruebas consta de 2 partes: una centrada en pruebas unitarias, y otra en las pruebas de sistema. Para cada una de estas partes se indica el plan propiamente dicho (es decir, la descripción de cada una de las pruebas a realizar, y los resultados obtenidos tras su ejecución), y también se incluye la lista de los fallos detectados durante dicha ejecución, así como el modo en que ha sido corregido cada uno de ellos, si ha sido posible.

Las pruebas unitarias han sido ejecutadas mediante el software JUnit, que permite realizarlas de una forma relativamente simple y automatizada. Estas pruebas no se han realizado con métodos de las clases Java correspondientes a los comandos; es decir, tan solo se han realizado con las clases pertenecientes al subsistema (paquete) *Utils*. Las razones para hacerlo así son fundamentalmente 3: por un lado, las pruebas de sistema abordan exactamente las mismas situaciones: la ejecución y la disponibilidad de cada uno de los comandos; por otro lado, realizar unas pruebas tan amplias con JUnit sería enormemente complejo, siendo mucho más probable cometer errores en el desarrollo de las clases de test que en el propio software a probar; y finalmente, y más importante, al estar estas clases diseñadas solamente para ser invocadas por Roo, no se podría reproducir su comportamiento al ejecutarlas con JUnit, ya que algunos objetos utilizados en estas clases son construidos por Roo.

Tampoco se han realizado pruebas unitarias de métodos del subsistema *Utils* que son triviales o que utilizan parámetros procedentes de Roo, por la misma razón que acabamos de ver. En cualquier caso, las pruebas de sistema cubren prácticamente todo el código que puede ser ejecutado durante el uso de Roo++, si bien sin centrarse tanto en el detalle.

En ningún caso se han realizado pruebas unitarias de métodos privados, ya que JUnit no lo permite, y además estos métodos siempre son cubiertos por los que los invocan, y por tanto también acaban siendo cubiertos por las pruebas unitarias.

Para la ejecución de las pruebas unitarias se han creado clases de test (con el mismo nombre de la clase a probar, seguido de la palabra “Test”), y se ha hecho uso de un directorio “target/tests”, dentro del proyecto de Roo++, en el que se creaban todos los archivos y directorios necesarios para la ejecución de dichas pruebas.

En el caso de las pruebas de sistema, se ha decidido probar cada uno de los comandos de Roo++, y, en el caso de los que tienen un comportamiento claramente diferenciado dependiendo de la situación, se han probado tantas veces como ha sido necesario para cubrir todas las situaciones relevantes, de forma que se cubriera en estas pruebas toda o casi toda la funcionalidad del software.

Pruebas unitarias

1. Clase *CreateModules.java*

Prueba n°:	1.1
Entrada:	<code>copyResources("proyectoCopyResources", "target/tests/proyectoCopyResources", "rooAddon/", this.getClass() .getClassLoader());</code>
Salida esperada:	<ol style="list-style-type: none">No existe el directorio "target/tests/proyectoCopyResources/src"Existe cada uno de los archivos target/tests/proyectoCopyResources/proyectoCopyResources-*, siendo *: pom.xml core/pom.xml docs/pom.xml docs/src/InstallationManual.xml docs/src/UserManual.xml docs/src/assembly/package.xml ear/pom.xml ear/src/main/application/META-INF/weblogic-application.xml package/pom.xml package/src/assembly/package.xml scripts/pom.xml scripts/src/assembly/compress-zip.xmlExiste cada uno de los directorios: target/tests/proyectoCopyResources/proyectoCopyResources-*, siendo *: core core/src core/src/main core/src/main/java core/src/main/resources core/src/test core/src/test/java core/src/test/resources docs docs/src docs/src/assembly docs/src/docbkx docs/src/fonts docs/src/front_pages docs/src/images ear ear/src ear/src/main ear/src/main/application ear/src/main/application/META-INF package

	package/src package/src/assembly scripts scripts/src scripts/src/assembly scripts/src/main scripts/src/main/resources
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

2. Clase *FileUtilities.java*

Prueba n°:	2.1
Condiciones:	El fichero "target/tests/este_fichero_no_existe.txt" no existe
Entrada:	<i>FileUtilities</i> <code>.readFileAsStringThrows("target/tests/este_fichero_no_existe.txt");</code>
Salida esperada:	Se produce una IOException
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	2.2
Condiciones:	El fichero "target/tests/este_fichero_existe.txt" contiene el texto "Este fichero\nexiste."
Entrada:	<i>FileUtilities</i> <code>.readFileAsStringThrows("target/tests/este_fichero_existe.txt");</code>
Salida esperada:	El método devuelve la cadena "Este fichero\nexiste."
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	2.3
Condiciones:	El fichero "target/tests/este_fichero_no_existe.txt" no existe
Entrada:	<i>FileUtilities</i> <code>.readFileAsString("target/tests/este_fichero_no_existe.txt");</code>
Salida esperada:	El método devuelve la cadena vacía
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.4
Condiciones:	El fichero "target/tests/este_fichero_existe.txt" contiene el texto "Este fichero\nexiste."
Entrada:	<i>FileUtilities</i> <i>.readFileAsString("target/tests/este_fichero_existe.txt");</i>
Salida esperada:	El método devuelve la cadena "Este fichero\nexiste."
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.5
Condiciones:	El fichero "target/tests/reemplazo_de_texto.txt" contiene el texto "Este archivo contiene un texto a ser reemplazado por otro texto."
Entrada:	<i>FileUtilities.replaceText("texto a ser reemplazado", "texto tras reemplazarlo", "target/tests/reemplazo_de_texto.txt");</i>
Salida esperada:	El archivo contiene el texto "Este archivo contiene un texto tras reemplazarlo por otro texto."
Salida obtenida:	El archivo contiene ese texto más un salto de línea adicional al final.
Fallos detectados:	Fallo 1

Prueba nº:	2.6
Condiciones:	El fichero "target/tests/reemplazo_de_texto.txt" contiene el texto "Este archivo contiene un texto tras reemplazarlo por otro texto."
Entrada:	<i>FileUtilities.replaceText("texto a ser reemplazado", "texto tras reemplazarlo", "target/tests/reemplazo_de_texto.txt");</i>
Salida esperada:	El archivo contiene el texto "Este archivo contiene un texto tras reemplazarlo por otro texto."
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.7
Condiciones:	En la Máquina Virtual de Java, se ha establecido la propiedad ROOADDON con el valor "src/main/resources/RooAddonsData".
Entrada:	<i>FileUtilities.getConfigurationFilePath("doc/fonts.xml");</i>
Salida esperada:	El método devuelve la ruta completa del archivo "src/main/resources/RooAddonsData/doc/fonts.xml"
Salida obtenida:	Igual a la esperada

Fallos detectados:	Ninguno
---------------------------	---------

Prueba nº:	2.8
Condiciones:	El archivo "target/tests/archivo_lineas_vacias.txt" contiene el texto "Este fichero\ntiene\n \n\n\nalguna \n\t \t\n que otra linea\nvacía\n"
Entrada:	<i>FileUtilities</i> <i>.removeBlankLines("target/tests/archivo_lineas_vacias.txt");</i>
Salida esperada:	El archivo contiene el texto "Este fichero\ntiene\nalguna \n que otra linea\nvacía".
Salida obtenida:	El archivo contiene ese texto más un salto de línea adicional al final.
Fallos detectados:	Fallo 2

Prueba nº:	2.9
Condiciones:	El archivo "target/tests/directorio_a_copiar/archivo1.txt" contiene el texto "Contenido del archivo 1.\n", y "target/tests/directorio_a_copiar/subdirectorio/archivo2.txt" el texto "Contenido del archivo 2.\n". Además, el directorio "target/tests/directorio_copiado_que_no_existia" no existe.
Entrada:	<i>FileUtilities.copyDirContent("target/tests/directorio_a_copiar", "target/tests/directorio_copiado_que_no_existia");</i>
Salida esperada:	"target/tests/directorio_copiado_que_no_existia/archivo1.txt" contiene el texto "Contenido del archivo 1.\n", y "target/tests/directorio_copiado_que_no_existia/subdirectorio/archivo2.txt" contiene "Contenido del archivo 2.\n".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.10
Condiciones:	El archivo "target/tests/directorio_a_copiar/archivo1.txt" contiene el texto "Contenido del archivo 1.\n", y "target/tests/directorio_a_copiar/subdirectorio/archivo2.txt" el texto "Contenido del archivo 2.\n". Además, el directorio "target/tests/directorio_copiado_que_ya_existia" existe.
Entrada:	<i>FileUtilities.copyDirContent("target/tests/directorio_a_copiar", "target/tests/directorio_copiado_que_ya_existia");</i>
Salida esperada:	"target/tests/directorio_copiado_que_ya_existia/archivo1.txt" contiene el texto "Contenido del archivo 1.\n", y "target/tests/directorio_copiado_que_ya_existia/subdirectorio/archivo2.txt" contiene "Contenido del archivo 2.\n".

Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.11
Entrada:	<i>FileUtilities.readFile("textfiles.txt", "rooAddon/", this.getClass().getClassLoader());</i>
Salida esperada:	El método devuelve la cadena "pom.xml\r\core\pom.xml\r\docs\pom.xml\r\docs\src\InstallationManual.xml\r\docs\src\UserManual.xml\r\docs\src\assembly\package.xml\r\near\pom.xml\r\near\src\main\application\META-INF\weblogic-application.xml\r\npackage\pom.xml\r\npackage\src\assembly\package.xml\r\nscripts\pom.xml\r\nscripts\src\assembly\compress-zip.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.12
Condiciones:	El contenido de la primera línea del archivo de configuración <i>licenses.xml</i> es "<licenses>".
Entrada:	<i>FileUtilities.getFirstLineConfFile("licenses.xml");</i>
Salida esperada:	El método devuelve la cadena "<licenses>".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	2.13
Condiciones:	El archivo "esto_no_existe.xml" no existe dentro del directorio de configuración.
Entrada:	<i>FileUtilities.getFirstLineConfFile("esto_no_existe.xml");</i>
Salida esperada:	Se produce una IOException
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

3. Clase PomUtils.java

Prueba nº:	3.1
Condiciones:	El archivo "target/tests/pom.xml" contiene el texto del archivo pom.xml del módulo docs, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>PomUtils.replaceVariables("target/tests/pom.xml", "", "proyecto",</i>

	<i>"1.0", "top.level.package");</i>
Salida esperada:	<p>El archivo contiene:</p> <pre> <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd"> <modelVersion>4.0.0</modelVersion> <parent> <groupId>top.level.package</groupId> <artifactId>proyecto</artifactId> <version>1.0</version> </parent> <groupId>top.level.package</groupId> <artifactId>proyecto-docs</artifactId> <version>1.0</version> <name>proyecto-docs</name> <packaging>pom</packaging> <build> <plugins> <plugin> <groupId>com.agilejava.docbkk</groupId> <artifactId>docbkk-maven-plugin</artifactId> <version>2.0.8</version> <executions> <execution> <id>generate-help-pdf</id> <goals> <goal>generate-pdf</goal> </goals> <phase>generate-resources</phase> <configuration> <foCustomization>\${ {basedir}/src/docbkk/custom.xsl</foCustomization> <paperType>A4</paperType> <imgSrcPath>\${ {basedir}/src/</imgSrcPath> </configuration> </execution> </executions> </plugin> <plugin> <groupId>com.agilejava.docbkk</groupId> <artifactId>docbkk-fop-support</artifactId> <version>2.0.11</version> <executions> <execution> <phase>generate-sources</phase> <goals> <goal>generate</goal> </goals> </execution> </executions> </plugin> </plugins> </build> </project> </pre>

	<pre> </goals> <configuration> <ansi>true</ansi> </configuration> </execution> </executions> </plugin> </plugins> <plugin> <groupId>org.apache.maven.plugins</groupId> <artifactId>maven-assembly-plugin</artifactId> <version>2.4</version> <configuration> <descriptors> <descriptor>src/assembly/package.xml</descriptor> </descriptors> <appendAssemblyId>>false</appendAssemblyId> </configuration> <executions> <execution> <id>generate-am-help</id> <phase>package</phase> <goals> <goal>attached</goal> </goals> </execution> </executions> </plugin> </plugins> </build> </project> </pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.2
Condiciones:	El archivo "target/tests/pom.xml" contiene el texto del archivo pom.xml del módulo docs, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>PomUtils.addDependency("target/tests/pom.xml", "groupId", "artifactId", "version");</i>
Salida esperada:	El archivo acaba con el texto: <pre> <dependencies>\n<dependency>\n<groupId>groupId</groupId>\n<artifactId>artifactId</artifactId>\n<version>version</version>\n</dependency>\n</dependencies>\n</project> </pre> siendo el resto de su contenido el mismo que el inicial.
Salida obtenida:	El archivo acaba con el texto: <pre> <dependencies>\n<dependency>\n<artifactId>artifactId</artifactId>\n<groupId>groupId</groupId>\n<version>version</version>\n</dependency>\n</dependencies>\n</project> </pre> siendo el resto de su contenido el mismo que el inicial.
Fallos detectados:	Fallo 3

Prueba nº:	3.3
Condiciones:	El archivo "target/tests/pom.xml" contiene el texto del archivo pom.xml del módulo docs, tal como se encuentra en los "resources" del JAR de Roo++.

Entrada:	<i>PomUtils.addDependencyToBuildPlugin("target/tests/pom.xml", "docbkx-maven-plugin", "groupId", "artifactId", "version");</i>
Salida esperada:	Como último hijo del nodo “dependencies”, se introduce: <dependency>\n<groupId>groupId</groupId>\n<artifactId>artifactId</artifactId>\n<version>version</version>\n</dependency>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.4
Condiciones:	El archivo "target/tests/scope-pom.xml" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los “resources” del JAR de Roo++.
Entrada:	<i>PomUtils.addScope("test", "/project/build/plugins/plugin/artifactId", doc, "target/tests/scope-pom.xml");</i>
Salida esperada:	Como último hijo del único nodo de nombre “plugin” en el archivo es introducido el nodo: <scope>test</scope>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.5
Condiciones:	El archivo "target/tests/properties-pom.xml" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los “resources” del JAR de Roo++.
Entrada:	<i>PomUtils.addProperty("target/tests/properties-pom.xml", "propiedad1", "valor1");</i>
Salida esperada:	Como último hijo del nodo “project” es introducido el siguiente: <properties>\n<propiedad1>valor1</propiedad1>\n</properties>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.6
Condiciones:	El archivo "target/tests/properties-pom.xml" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los “resources” del JAR de Roo++, y además se ha ejecutado la prueba 3.5 sobre el mismo.
Entrada:	<i>PomUtils.addProperty("target/tests/properties-pom.xml", "propiedad2", "valor2");</i>
Salida esperada:	Es introducido como hijo del nodo “properties” el siguiente: <propiedad2>valor2</propiedad2>

Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.7
Condiciones:	El archivo "target/tests/pom-que-no-existe.xml" no existe.
Entrada:	<i>PomUtils.addProperty("target/tests/pom-que-no-existe.xml", "propiedad2", "valor2");</i>
Salida esperada:	Se produce una IOException
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.8
Condiciones:	El archivo "target/tests/version-pom.xml" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los “resources” del JAR de Roo++.
Entrada:	<i>PomUtils.setVersion("target/tests/version-pom.xml", "3.0", fileManager);</i>
Salida esperada:	El valor del nodo “version” pasa de ser <code>\${project.version}</code> a 3.0
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.9
Condiciones:	El archivo "target/tests/license-plugin-pom.xml" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los “resources” del JAR de Roo++.
Entrada:	<i>PomUtils.addLicensePlugin("target/tests/license-plugin-pom.xml");</i>
Salida esperada:	Como último hijo del nodo “plugins” es introducido el nodo contenido en el archivo de configuración "plugin-license.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	3.10
Condiciones:	El archivo "target/tests/license-pom.xml" contiene el texto del archivo pom.xml del módulo raíz, tal como se encuentra en los “resources” del JAR de Roo++.
Entrada:	<i>PomUtils.addLicense("target/tests/license-pom.xml");</i>
Salida esperada:	El nodo contenido en el archivo de configuración "licenses.xml" es introducido como último hijo de “project”.

Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	3.11
Condiciones:	El archivo "target/tests/license-pom.xml" contiene el texto del archivo pom.xml del módulo raíz, tal como se encuentra en los “resources” del JAR de Roo++, y se ha ejecutado la prueba 3.10 sobre el mismo.
Entrada:	<i>PomUtils.addLicense("target/tests/license-pom.xml");</i>
Salida esperada:	No se producen cambios en el archivo.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	3.12
Condiciones:	El archivo "target/tests/build-plugins-pom.xml" contiene el texto del archivo pom.xml del módulo raíz, tal como se encuentra en los “resources” del JAR de Roo++, y <i>plugin</i> es un objeto Node que tiene 3 nodos hijos: <i>groupId</i> (valor <i>groupId</i>), <i>artifactId</i> (valor <i>artifactId</i>) y <i>version</i> (valor <i>version</i>).
Entrada:	<i>PomUtils.addBuildPlugin("target/tests/build-plugins-pom.xml", plugin);</i>
Salida esperada:	Se crea, como último hijo del nodo “project”, un nuevo elemento “build”, que tiene como hijo un nuevo nodo “plugins”, y este a su vez el nodo pasado como parámetro, del que “cuelgan” los que ya eran sus nodos hijos
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	3.13
Condiciones:	El archivo "target/tests/build-plugins-pom.xml" contiene el texto del archivo pom.xml del módulo raíz, tal como se encuentra en los “resources” del JAR de Roo++, pero además con un nodo “build” como último hijo del elemento “project”; y <i>plugin</i> es un objeto Node que tiene 3 nodos hijos: <i>groupId</i> (valor <i>groupId</i>), <i>artifactId</i> (valor <i>artifactId</i>) y <i>version</i> (valor <i>version</i>).
Entrada:	<i>PomUtils.addBuildPlugin("target/tests/build-plugins-pom.xml", plugin);</i>
Salida esperada:	Se crea, como último hijo del nodo “build”, que tiene como hijo un nuevo nodo “plugins”, y este a su vez el nodo pasado como parámetro, del que “cuelgan” los que ya eran sus nodos hijos
Salida obtenida:	Igual a la esperada
Fallos	Ninguno

detectados:	
Prueba nº:	3.14
Condiciones:	El archivo "target/tests/build-plugins-pom.xml" contiene el texto del archivo pom.xml del módulo raíz, tal como se encuentra en los "resources" del JAR de Roo++, pero además con un nodo "build" como último hijo del elemento "project"; se ha ejecutado la prueba 3.13 sobre él; y <i>plugin</i> es un objeto Node que tiene 3 nodos hijos: <i>groupId</i> (valor <i>groupId2</i>), <i>artifactId</i> (valor <i>artifactId2</i>) y <i>version</i> (valor <i>version2</i>).
Entrada:	<i>PomUtils.addBuildPlugin("target/tests/build-plugins-pom.xml", plugin);</i>
Salida esperada:	Se introduce como último hijo del nodo "plugins" el nodo pasado como parámetro, del que a su vez "cuelgan" los que ya eran sus nodos hijos
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

4. Clase *ProjectUtils.java*

Prueba nº:	4.1
Condiciones:	Existe el directorio "target/tests/proyecto/proyecto-core", y está vacío
Entrada:	<i>ProjectUtils.copyTextFile("target/tests/proyecto", "proyecto", "core/pom.xml", "rooAddon/", this.getClass().getClassLoader());</i>
Salida esperada:	El archivo "target/tests/proyecto/proyecto-core/pom.xml" contiene el mismo texto que el archivo "core/pom.xml", tal como se encuentra en los "resources" del JAR de Roo++.
Salida obtenida:	El archivo "target/tests/proyecto/proyecto-core/pom.xml" contiene el mismo texto que el archivo "core/pom.xml", como se encuentra en los "resources" del JAR de Roo++, pero con un salto de línea adicional al final.
Fallos detectados:	Fallo 4

Prueba nº:	4.2
Condiciones:	El archivo "target/tests/archivoACopiar.txt" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los "resources" del JAR de Roo++, y el directorio "target/tests/proyecto/proyecto-otromodulo" existe y está vacío.
Entrada:	<i>ProjectUtils.copyExternTextFile("target/tests/proyecto", "proyecto", "otromodulo/pom.xml", "target/tests/archivoACopiar.txt");</i>
Salida esperada:	El archivo "target/tests/proyecto/proyecto-otromodulo/archivoACopiar.txt" contiene el texto del archivo pom.xml del módulo core, tal como se encuentra en los "resources" del JAR de Roo++

Salida obtenida:	El archivo "target/tests/proyecto/proyecto-otromodulo/archivoACopiar.txt" contiene el texto del archivo pom.xml del módulo core, como se encuentra en los "resources" del JAR de Roo++, pero con un salto de línea adicional al final.
Fallos detectados:	Fallo 5

Prueba nº:	4.3
Condiciones:	El archivo "target/tests/proyecto/proyecto-core/archivodetexto.txt" contiene el texto "Este archivo contiene texto."
Entrada:	<i>ProjectUtils.appendTextToFile("Texto introducido en el archivo", "target/tests/proyecto", "proyecto", "core/archivodetexto.txt");</i>
Salida esperada:	El archivo "target/tests/proyecto/proyecto-core/archivodetexto.txt" contiene el texto "Este archivo contiene texto.\nTexto introducido en el archivo"
Salida obtenida:	El archivo contiene ese texto pero con un salto de línea adicional al final.
Fallos detectados:	Fallo 6

5. Clase *WebModuleUtils.java*

Prueba nº:	5.1
Condiciones:	Existen todos los directorios de la ruta "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring".
Entrada:	<i>WebModuleUtils.copyApplicationPropertiesToWebModule("target/tests/proyecto", "rooAddon/", "proyecto", this.getClass().getClassLoader());</i>
Salida esperada:	El archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/application.properties" contiene el mismo texto que el archivo equivalente de los "resources" del JAR de Roo++.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	5.2
Condiciones:	<ul style="list-style-type: none"> El contenido del archivo "target/tests/proyecto/proyecto-web/pom.xml" es: <pre><project><dependencies><dependency><groupId>mysql</groupId></dependency></dependencies></project></pre> Existen todos los directorios de la ruta "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". El contenido del archivo de configuración <i>mysql.txt</i> es: <pre>5.1.18 MySQL5InnoDBDialect</pre>
Entrada:	<i>WebModuleUtils.copyDatabasePropertiesToWebModule(</i>

	<code>"target/tests/proyecto", "rooAddon/", "proyecto", "database", this .getClass().getClassLoader());</code>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/database.properties" es: #Datasource JNDI jndi.datasource=java:comp/env/jdbc/proyectoDS #Persistence Unit database.persistence.unit=persistenceUnitProyecto #Hibernate properties database.hibernate.default_schema=DATABASE database.hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100
Salida obtenida:	Se obtiene el siguiente mensaje de error: File web/src/main/resources/META-INF/spring/database.properties not found
Fallos detectados:	Fallo 7

Prueba n°:	5.3
Condiciones:	<ul style="list-style-type: none"> El contenido del archivo "target/tests/proyecto/proyecto-web/pom.xml" es: <project><dependencies><dependency><groupId>net.sourceforge.jtds</groupId></dependency></dependencies></project> Existen todos los directorios de la ruta "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". El contenido del archivo de configuración <i>mssql.txt</i> es: 1.2.4 SQLServer2008Dialect
Entrada:	<code>WebModuleUtils.copyDatabasePropertiesToWebModule("target/tests/proyecto", "rooAddon/", "proyecto", "database", this .getClass().getClassLoader());</code>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/database.properties" es: #Datasource JNDI jndi.datasource=java:comp/env/jdbc/proyectoDS #Persistence Unit database.persistence.unit=persistenceUnitProyecto #Hibernate properties database.hibernate.default_schema=DATABASE database.hibernate.dialect=org.hibernate.dialect.SQLServer2008Dialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.4
Condiciones:	<ul style="list-style-type: none"> El contenido del archivo "target/tests/proyecto/proyecto-web/pom.xml" es: <pre><project><dependencies><dependency><groupId>ojdbc</groupId></dependency></dependencies></project></pre> Existen todos los directorios de la ruta "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". El contenido del archivo de configuración <i>oracle.txt</i> es: <pre>16 Oracle10gDialect</pre>
Entrada:	<i>WebModuleUtils.copyDatabasePropertiesToWebModule("target/tests/proyecto", "rooAddon/", "proyecto", "database", this.getClass().getClassLoader());</i>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/database.properties" es: <pre>#Datasource JNDI jndi.datasource=java:comp/env/jdbc/proyectoDS #Persistence Unit database.persistence.unit=persistenceUnitProyecto #Hibernate properties database.hibernate.default_schema=DATABASE database.hibernate.dialect=org.hibernate.dialect.Oracle10gDialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100</pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.5
Condiciones:	<ul style="list-style-type: none"> El contenido del archivo "target/tests/proyecto/proyecto-web/pom.xml" es: <pre><project><dependencies><dependency><groupId>postgresql</groupId></dependency></dependencies></project></pre> Existen todos los directorios de la ruta "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". El contenido del archivo de configuración <i>postgresql.txt</i> es: <pre>9.0-801.jdbc4 PostgreSQLDialect</pre>
Entrada:	<i>WebModuleUtils.copyDatabasePropertiesToWebModule("target/tests/proyecto", "rooAddon/", "proyecto", "database", this.getClass().getClassLoader());</i>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/database.properties" es: <pre>#Datasource JNDI jndi.datasource=java:comp/env/jdbc/proyectoDS</pre>

	<pre>#Persistence Unit database.persistence.unit=persistenceUnitProyecto #Hibernate properties database.hibernate.default_schema=DATABASE database.hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100</pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.6
Condiciones:	<ul style="list-style-type: none"> • Existe el directorio: "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". • El archivo de configuración applicationContext-platform.xml existe y tiene contenido.
Entrada:	<i>WebModuleUtils.copyApplicationContextPlatformToWebModule("target/tests/proyecto", "proyecto");</i>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext-platform.xml" es el mismo que el del archivo de configuración del mismo nombre.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.7
Condiciones:	<ul style="list-style-type: none"> • Existe el directorio: "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". • El archivo de configuración usermgmt/applicationContext-security.xml existe y tiene contenido.
Entrada:	<i>WebModuleUtils.copyApplicationContextSecurityToWebModule("target/tests/proyecto", "proyecto");</i>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext-security.xml" es el mismo que el del archivo de configuración usermgmt/applicationContext-security.xml
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.8
Condiciones:	<ul style="list-style-type: none"> • Existe el directorio: "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring". • El archivo de configuración applicationContext-portal.xml existe y tiene contenido.
Entrada:	<i>WebModuleUtils.copyApplicationContextPortalToWebModule("target/tests/proyecto", "proyecto");</i>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext-portal.xml" es el mismo que el del archivo de configuración del mismo nombre.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.9
Condiciones:	<ul style="list-style-type: none"> • El contenido del archivo "proyecto/proyecto-web/src/main/webapp/WEB-INF/views/header.jspx" es: Archivo a ser sustituido • El contenido del archivo de configuración portal-header.jspx es: <pre><div id="header" xmlns:jsp="http://java.sun.com/JSP/Page" xmlns:fn="http://java.sun.com/jsp/jstl/functions" xmlns:c="http://java.sun.com/jsp/jstl/core" xmlns:spring="http://www.springframework.org/tags" xmlns:portal="http://www.empresa.com/portal" version="2.0"> <jsp:directive.page import="org.springframework.web.servlet.support.RequestContextUtils" /> <jsp:directive.page import="org.springframework.context.i18n.LocaleContextHolder" /> <jsp:directive.page import="com.empresa.portal.handler.PortalClientHandler" /> <jsp:scriptlet> pageContext.setAttribute("applications", PortalClientHandler.getApplications(), PageContext.PAGE_SCOPE); pageContext.setAttribute("configuration", PortalClientHandler.getConfiguration(), PageContext.PAGE_SCOPE); pageContext.setAttribute("currentLocale", LocaleContextHolder.getLocale(), PageContext.PAGE_SCOPE); pageContext.setAttribute("requestUri", org.springframework.security.web.util.UrlUtils .buildFullRequestUrl(pageContext.getRequest().getScheme(), pageContext.getRequest().getServerName(), pageContext.getRequest().getServerPort(), pageContext.getAttribute("javax.servlet.forward.request_uri", 2).toString(), null), 2); </jsp:scriptlet> <portal:portal-header applications="\${applications}" currentLocale="\${ currentLocale}" configuration="\${configuration}" requestUri="\${requestUri}" /></pre>
Entrada:	<i>WebModuleUtils.replaceHeader("target/tests/proyecto", "proyecto");</i>
Salida esperada:	El contenido del archivo "target/tests/proyecto/proyecto-web/src/main/webapp/WEB-INF/views/header.jspx" es el mismo que el arriba

	indicado para el archivo de configuración portal-header.jspx.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.10
Condiciones:	El contenido del archivo "proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext.xml" es: <pre><beans> <bean class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean"> </bean> <bean id = "dataSource"> </bean> </beans></pre>
Entrada:	<i>WebModuleUtils.updateAppContextDatabase("target/tests/proyecto", "rooAddon/", "proyecto", this.getClass().getClassLoader());</i>
Salida esperada:	El contenido del archivo "proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext.xml" es: <pre><beans xmlns:jee="http://www.springframework.org/schema/jee"> <jee:jndi-lookup id="dataSource" jndi-name="{jndi.datasource}"/> <bean class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean" id="entityManagerFactory"> <property name="persistenceXmlLocation" value="classpath:META-INF/persistenceInfo.xml"/> <property name="persistenceUnitName" value="{database.persistence.unit}"/> <property name="dataSource" ref="dataSource"/> <property name="jpaPropertyMap"> <map> <entry key="hibernate.default_schema" value="{database.hibernate.default_schema}"/> <entry key="hibernate.dialect" value="{database.hibernate.dialect}"/> <entry key="hibernate.hbm2ddl.auto" value="{database.hibernate.hbm2ddl.auto}"/> <entry key="hibernate.ejb.naming_strategy" value="{database.hibernate.ejb.naming_strategy}"/> <entry key="hibernate.connection.charset" value="{database.hibernate.connection.charset}"/> <entry key="hibernate.jdbc.fetch_size" value="{database.hibernate.jdbc.fetch_size}"/> <entry key="hibernate.jdbc.batch_size" value="{database.hibernate.jdbc.batch_size}"/> </map> </property> </bean> </beans></pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.11
Condiciones:	Existe el directorio "target/tests/proyecto/proyecto-web/src/main".

Entrada:	<i>WebModuleUtils.copyJettyConfWithoutDatabase("target/tests/proyecto", "proyecto", "rooAddon/", this.getClass().getClassLoader());</i>
Salida esperada:	<ol style="list-style-type: none"> 1. Dentro de "target/tests/proyecto/proyecto-web/src/main/", existe cada uno de los directorios: "jetty", "jetty/etc", "jetty/logs", "jetty/webapps", "jetty/webapps-plus". 2. El archivo "target/tests/proyecto/proyecto-web/src/main/jetty/jetty.xml" contiene lo mismo que el archivo "web/src/main/jetty/jetty.xml", dentro de los "resources" del JAR de Roo++. 3. El archivo "target/tests/proyecto/proyecto-web/src/main/jetty/etc/realm.properties" contiene lo mismo que el archivo ""web/src/main/jetty/etc/realm.properties"", dentro de los "resources" del JAR de Roo++. 4. El archivo "target/tests/proyecto/proyecto-web/src/main/jetty/etc/webdefault.xml" contiene lo mismo que el archivo "web/src/main/jetty/etc/webdefault.xml", dentro de los "resources" del JAR de Roo++.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.12
Condiciones:	El archivo "target/tests/proyecto2/proyecto2-web/pom.xml" contiene: <pre><project> <build> <plugins> </plugins> </build> </project></pre>
Entrada:	<i>WebModuleUtils.addLicenseToWebModule("target/tests/proyecto2", "rooAddon/", "proyecto2", this.getClass().getClassLoader());</i>
Salida esperada:	El archivo "target/tests/proyecto2/proyecto2-web/pom.xml" contiene: <pre><project> <build> <plugins> <plugin> <groupId>org.codehaus.mojo</groupId> <artifactId>license-maven-plugin</artifactId> <version>1.0</version> <configuration> <excludedScopes>test,system,provided</excludedScopes> <generateBundle>>false</generateBundle> <groupByLicense>>true</groupByLicense> <useMissingFile>>true</useMissingFile> <useRepositoryMissingFiles>>true</useRepositoryMissingFiles> <deployMissingFile>>true</deployMissingFile> <force>>false</force> <licenseMerges> <licenseMerge><<![CDATA[The Apache Software License, Version 2.0 Apache License, Version 2.0 Apache License Version 2.0 Apache 2]]></licenseMerge> <licenseMerge><<![CDATA[GNU LESSER GENERAL PUBLIC LICENSE GNU Lesser General Public Licence GNU Lesser General Public License]]></licenseMerge> <licenseMerge><<![CDATA[Eclipse Public License v1.0 Eclipse Public License - v 1.0]]></licenseMerge> </licenseMerges></pre>

	<pre> </configuration> <executions> <execution> <id>add-third-party</id> <goals> <goal>add-third-party</goal> </goals> <phase>process-sources</phase> </execution> </executions> </plugin> </plugins> </build> </project> </pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.13
Condiciones:	El archivo "target/tests/proyecto2/proyecto2-web/pom.xml" contiene: <pre> <project> <build> <plugins> </plugins> </build> </project> </pre>
Entrada:	<i>WebModuleUtils.addSourcePlugin("target/tests/proyecto2/proyecto2-web/pom.xml", "rooAddon/", this.getClass().getClassLoader());</i>
Salida esperada:	El archivo "target/tests/proyecto2/proyecto2-web/pom.xml" contiene: <pre> <project> <build> <plugins> <plugin> <groupId>org.apache.maven.plugins</groupId> <artifactId>maven-source-plugin</artifactId> <version>2.1.1</version> <executions> <execution> <id>attach-sources</id> <phase>verify</phase> <goals> <goal>jar-no-fork</goal> </goals> </execution> </executions> </plugin> </plugins> </build> </project> </pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	5.14
Condiciones:	El archivo "target/tests/renamepersistence/renamepersistence-web/src/main/resources/META-INF/persistence.xml" contiene el texto: <pre> persistence-unit name="persistenceUnit" </pre>

Entrada:	<code>WebModuleUtils.renamePersistence("target/tests/renamepersistence", "renamepersistence");</code>
Salida esperada:	<ol style="list-style-type: none"> 1. El archivo "target/tests/renamepersistence/renamepersistence-web/src/main/resources/META-INF/persistence.xml" no existe. 2. El archivo "target/tests/renamepersistence/renamepersistence-web/src/main/resources/META-INF/persistenceInfo.xml" contiene el texto: persistence-unit name="persistenceUnitRenamepersistence"
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

6. Clase *XmlUtils.java*

Prueba nº:	6.1
Condiciones:	El archivo "target/tests/addChildToRoot.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<code>XmlUtils.addChildToRoot("target/tests/addChildToRoot.xml", "nombredelnodo", "valordelnodo");</code>
Salida esperada:	Se ha introducido el nodo <code><nombredelnodo>valordelnodo</nombredelnodo></code> como último hijo del nodo raíz de "target/tests/addChildToRoot.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.2
Condiciones:	El archivo "target/tests/addChildToRoot.xml" contenía el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++, y se ha ejecutado sobre él la prueba 6.1.
Entrada:	<code>XmlUtils.addChildToRoot("target/tests/addChildToRoot.xml", "nodovacio", "");</code>
Salida esperada:	Se ha introducido el nodo <code><nodovacio/></code> como último hijo del nodo raíz de "target/tests/addChildToRoot.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.3
Condiciones:	El archivo "target/tests/addChildToNode.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.

Entrada:	<i>XmlUtils.addChildToNode("target/tests/addChildToNode.xml", "/project/parent", "nodovacio", "");</i>
Salida esperada:	En el archivo "target/tests/addChildToNode.xml", se ha introducido como último hijo de "/project/parent" el nodo <nodovacio/>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.4
Condiciones:	El archivo "target/tests/addChildToNode.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++, y se ha ejecutado sobre el mismo la prueba 6.3.
Entrada:	<i>XmlUtils.addChildToNode("target/tests/addChildToNode.xml", "/project/parent/nodovacio", "nombredelnodo", "valordelnodo");</i>
Salida esperada:	En el archivo "target/tests/addChildToNode.xml", se ha introducido como último hijo de "/project/parent/nodovacio" el nodo <nombredelnodo>valordelnodo</nombredelnodo>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.5
Condiciones:	El archivo "target/tests/changeNodeValue.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>XmlUtils.changeNodeValue("target/tests/changeNodeValue.xml", "/project/parent/groupId", "New value");</i>
Salida esperada:	El nodo "/project/parent/groupId" del archivo "target/tests/changeNodeValue.xml" ahora es: <groupId>New value</groupId>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.6
Condiciones:	El archivo "target/tests/setNodeAttribute.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>XmlUtils.setNodeAttribute("target/tests/setNodeAttribute.xml", "/project/modules", "atributo", "valor");</i>
Salida esperada:	El nodo <modules> de ese archivo XML ha pasado a ser <modules

	<i>atributo="valor">.</i>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.7
Condiciones:	El archivo "target/tests/appendFileToRoot.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>XmlUtils.appendFileToRoot("target/tests/appendFileToRoot.xml", "rooAddon/", "web/resource-ref.xml", this.getClass() .getClassLoader());</i>
Salida esperada:	El nodo raíz del documento XML "web/resource-ref.xml", situado en los "resources" del JAR de Roo++, ha sido introducido como último hijo del nodo raíz del documento "target/tests/appendFileToRoot.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.8
Condiciones:	El archivo "target/tests/eliminaNodoSiExiste.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>XmlUtils.eliminaNodoSiExiste("target/tests/eliminaNodoSiExiste.xml", "/project/packaging");</i>
Salida esperada:	El nodo <code><packaging>pom</packaging></code> ha desaparecido del documento "target/tests/eliminaNodoSiExiste.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.9
Condiciones:	El archivo "target/tests/eliminaNodoSiExiste.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++, y se ha ejecutado la prueba 6.8 sobre el mismo.
Entrada:	<i>XmlUtils.eliminaNodoSiExiste("target/tests/eliminaNodoSiExiste.xml", "/project/packaging");</i>
Salida esperada:	No se ha producido ningún cambio en el documento "target/tests/eliminaNodoSiExiste.xml".
Salida obtenida:	Igual a la esperada

Fallos detectados:	Ninguno
---------------------------	---------

Prueba nº:	6.10
Condiciones:	El archivo "target/tests/guardaXml.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entradas:	<ol style="list-style-type: none"> 1. <i>doc</i> = <code>XmlUtils.createDocumentFromFile("target/tests/guardaXml.xml");</code> 2. <code>Node node = doc.createElement("nodo");</code> 3. <code>node.setTextContent("valor");</code> 4. <code>doc.getDocumentElement().appendChild(node);</code> 5. <code>XmlUtils.guardaXml(doc, "target/tests/guardaXml.xml");</code>
Salida esperada:	En el archivo "target/tests/guardaXml.xml", se ha introducido como último hijo de la raíz el nodo <code><nodo>valor</nodo></code> .
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.11
Condiciones:	El documento XML "doc" ha sido construido a partir del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<code>XmlUtils.evaluateXPath("/project/modules/module", doc);</code>
Salida esperada:	Devuelve un objeto NodeList que contiene los siguientes elementos: <ol style="list-style-type: none"> 1. <code><module>\${project}-core</module></code> 2. <code><module>\${project}-docs</module></code> 3. <code><module>\${project}-scripts</module></code> 4. <code><module>\${project}-web</module></code> 5. <code><module>\${project}-ear</module></code> 6. <code><module>\${project}-package</module></code>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba nº:	6.12
Condiciones:	El documento XML "doc" ha sido construido a partir del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<code>XmlUtils.evaluateXPath("/este/nodo/no/existe", doc);</code>
Salida esperada:	Devuelve un objeto NodeList que contiene la lista vacía.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.13
Condiciones:	El archivo "target/tests/createDocumentFromFile.xml" contiene el texto del archivo pom.xml raíz del proyecto, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>XmlUtils</i> <i>.createDocumentFromFile("target/tests/createDocumentFromFile.xml");</i>
Salida esperada:	Devuelve un documento que tiene exactamente el mismo contenido XML que el del archivo "target/tests/createDocumentFromFile.xml".
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.14
Condiciones:	El archivo "target/tests/archivoQueNoExiste.xml" no existe.
Entrada:	<i>XmlUtils</i> <i>.createDocumentFromFile("target/tests/archivoQueNoExiste.xml");</i>
Salida esperada:	Se produce una IOException.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.15
Condiciones:	El archivo "target/tests/xmlNoValido.xml" contiene el texto del archivo textfiles.txt, tal como se encuentra en los "resources" del JAR de Roo++.
Entrada:	<i>Document doc =</i> <i>XmlUtils.createDocumentFromFile("target/tests/xmlNoValido.xml");</i>
Salida esperada:	Se produce una ParserConfigurationException.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.16
Entrada:	<i>XmlUtils.createDocumentFromString("<raiz> <nodo1/> </raiz>");</i>
Salida esperada:	Devuelve un documento XML que tiene a <raiz> como nodo raíz, y a <nodo1/> como su único hijo.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.17
-------------------	------

Entrada:	<i>XmlUtils.createDocumentFromString("<raiz1> <nodo1/> </raiz1> <raiz2> <nodo2/> </raiz2>");</i>
Salida esperada:	Se producen una SAXException y una ParserConfigurationException.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.18
Entrada:	<i>XmlUtils.createDocumentFromString("<raiz> <nodo1/>");</i>
Salida esperada:	Se producen una SAXException y una ParserConfigurationException.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Prueba n°:	6.19
Entrada:	<i>XmlUtils.createDocumentFromString("Esto no es XML.")</i>
Salida esperada:	Se producen una SAXException y una ParserConfigurationException.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Fallos detectados en pruebas unitarias

Fallo 1

Clase:	FileUtilities
Método:	replaceText
Descripción	El archivo con el texto reemplazado contiene un salto de línea adicional al final.
Solución:	Utilizar el método <i>print</i> en lugar de <i>println</i> para escribir el texto resultante en el fichero.

Fallo 2

Clase:	FileUtilities
Método:	removeBlankLines
Descripción	El archivo del que se han eliminado las líneas en blanco contiene un salto de línea adicional al final.
Solución:	Utilizar el método <i>print</i> en lugar de <i>println</i> para escribir el texto resultante en el fichero.

Fallo 3

Clase:	PomUtils
Método:	addDependency
Descripción	Los elementos <i>groupId</i> y <i>artifactId</i> de la nueva dependencia introducida se encuentran en el orden contrario al esperado.
Solución:	En el método privado <i>addDependency</i> , que es invocado por el método público del mismo nombre, intercambiar el orden de las llamadas correspondientes a <i>Element.appendChild</i> .

Fallo 4

Clase:	ProjectUtils
Método:	copyTextFile
Descripción	El archivo resultante de la copia contiene un salto de línea adicional al final.
Solución:	Tras la lectura de una línea del archivo a copiar, introducir en la copia un salto de línea sólo si no se ha llegado al final del archivo.

Fallo 5

Clase:	ProjectUtils
---------------	--------------

Método:	copyExternTextFile
Descripción	El archivo resultante de la copia contiene un salto de línea adicional al final.
Solución:	Tras la lectura de una línea del archivo a copiar, introducir en la copia un salto de línea sólo si no se ha llegado al final del archivo.

Fallo 6

Clase:	ProjectUtils
Método:	appendTextToFile
Descripción	El archivo al que se le ha añadido el texto contiene además un salto de línea no esperado al final.
Solución:	Utilizar el método <i>print</i> en lugar de <i>println</i> para la escritura del nuevo texto en la parte final del archivo.

Fallo 7

Clase:	WebModuleUtils
Método:	copyDatabasePropertiesToWebModule
Descripción	Obtenido mensaje de “archivo no encontrado”, en referencia al archivo <i>database.properties</i> .
Solución:	Modificar el comportamiento del método <i>appendTextToFile</i> , de la clase <i>ProjectUtils</i> , de forma que si el archivo al que se intenta añadir el texto no existe dicho archivo sea creado de forma automática.

Pruebas de sistema

Situaciones de prueba

1. Ejecución del comando *roo++ project create modules*
 - 1.1 Sin especificar número de versión para el proyecto
 - 1.2 Especificando número de versión para el proyecto
 - 1.3 Existe el archivo de configuración *parent.txt*, y tiene el contenido esperado
 - 1.4 Existe el archivo de configuración *parent.txt*, pero no tiene el contenido esperado
 - 1.5 No existe el archivo de configuración *parent.txt*
2. Ejecución del comando *roo++ project license*
3. Ejecución del comando *roo++ project web configuration*
4. Ejecución del comando *roo++ project web jpa*
 - 4.1 No se especifican los parámetros opcionales
 - 4.2 Se especifican los parámetros opcionales
 - 4.3 Se indica MySQL como sistema de gestión de bases de datos
 - 4.4 Se indica Oracle como sistema de gestión de bases de datos
 - 4.5 Se indica PostgreSQL como sistema de gestión de bases de datos
 - 4.6 Se indica Microsoft SQL Server como sistema de gestión de bases de datos
 - 4.7 No se especifica ningún parámetro
5. Ejecución del comando *roo++ project web deploy*
 - 5.1 Se había ejecutado previamente el comando *roo++ project web jpa*
 - 5.1.1 Se había ejecutado previamente el comando *roo++ project web deploy*
 - 5.1.2 No se había ejecutado previamente el comando *roo++ project web deploy*
 - 5.2 No se había ejecutado previamente el comando *roo++ project web jpa*
 - 5.2.1 Se había ejecutado previamente el comando *roo++ project web deploy*
 - 5.2.2 No se había ejecutado previamente el comando *roo++ project web deploy*
6. Ejecución del comando *roo++ web breadcrumb setup*
7. Ejecución del comando *roo++ context setup*
8. Ejecución del comando *roo++ layout setup*
9. Ejecución del comando *roo++ platform setup*
10. Ejecución del comando *roo++ platform configuration*
11. Ejecución del comando *roo++ portal setup*
12. Ejecución del comando *roo++ usermgmt setup*
 - 12.1 Existe el directorio de configuración *usermgmt*

12.2 No existe el directorio de configuración *usermgmt*

13. Compilación del proyecto

Casos de prueba

NOTA: siempre que se habla de un archivo de configuración en un caso de prueba, hace referencia a un archivo situado en el directorio “RooAddonsData”, que puede encontrarse, o bien en el directorio “home” del usuario, o bien en cualquier otro lugar indicado por la variable de entorno *ROOADDON*, si ésta se encuentra definida en el sistema operativo. En estas pruebas se ha optado por la primera de las opciones, habiéndose probado anteriormente la otra mediante la ejecución de las pruebas unitarias. Además, si no se hace ninguna mención al mismo, el contenido de estos archivos será el mismo que en la versión de dichos archivos que es distribuida junto con el software. La razón de hacerlo así es evitar una especificación de casos de prueba que hubiera sido demasiado compleja y poco legible, ya que algunos de los casos pueden implicar hasta más de una decena de archivos, cada uno de ellos, a su vez, de un cierto tamaño y complejidad. Ver los documentos *DSI (Diseño del Sistema de Información)* y el *Manual del Usuario de Roo++* para más información sobre los archivos del proyecto en desarrollo y sobre los archivos de configuración.

Caso de prueba 1

Objetivo:	Cubrir las situaciones de prueba 1.1 y 1.3
Condiciones:	<ul style="list-style-type: none">• Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i>• El archivo de configuración <i>parent.txt</i> contiene el texto: parent.prueba.groupid elartifacitid 1.2.3.4
Entradas:	Ejecución del comando <i>roo++ project create-modules</i>
Salida esperada:	<ol style="list-style-type: none">1. Se elimina el directorio <i>src</i> preexistente2. Se crean los directorios de los diferentes módulos, cada uno con su correspondiente contenido3. El archivo <i>pom.xml</i> raíz del proyecto es modificado, pasando a contener referencias a los diferentes módulos, indicando <i>com.pruebas.prueba</i> como <i>groupId</i> del proyecto, <i>prueba</i> como <i>artifactId</i> y <i>1.0.BUILD-SNAPSHOT</i> como <i>version</i>, siendo el valor de estos 3 campos para el <i>parent</i> del proyecto el indicado en <i>parent.txt</i>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 2

Objetivo:	Cubrir las situaciones de prueba 1.2 y 1.4
Condiciones:	<ul style="list-style-type: none">• Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i>• El archivo de configuración <i>parent.txt</i> contiene el texto: parent.prueba.groupid

Entradas:	Ejecución del comando <code>roo++ project create-modules --version 2.4.5</code>
Salida esperada:	<ol style="list-style-type: none"> 1. Se elimina el directorio <code>src</code> preexistente 2. Se crean los directorios de los diferentes módulos, cada uno con su correspondiente contenido 3. El archivo <code>pom.xml</code> raíz del proyecto es modificado, pasando a contener referencias a los diferentes módulos, indicando <code>com.pruebas.prueba</code> como <code>groupId</code> del proyecto, <code>prueba</code> como <code>artifactId</code> y <code>2.4.5</code> como <code>version</code>, y no existiendo en dicho archivo el nodo <code>parent</code>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 3

Objetivo:	Cubrir la situación de prueba 1.5
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <code>project --topLevelPackage com.pruebas.prueba</code> • El archivo de configuración <code>parent.txt</code> no existe.
Entradas:	Ejecución del comando <code>roo++ project create-modules --version 2.4.5</code>
Salida esperada:	<ol style="list-style-type: none"> 1. Se elimina el directorio <code>src</code> preexistente 2. Se crean los directorios de los diferentes módulos, cada uno con su correspondiente contenido 3. El archivo <code>pom.xml</code> raíz del proyecto es modificado, pasando a contener referencias a los diferentes módulos, indicando <code>com.pruebas.prueba</code> como <code>groupId</code> del proyecto, <code>prueba</code> como <code>artifactId</code> y <code>2.4.5</code> como <code>version</code>, y no existiendo en dicho archivo el nodo <code>parent</code>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 4

Objetivo:	Cubrir la situación de prueba 2
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <code>project --topLevelPackage com.pruebas.prueba</code> • Haber ejecutado el caso de prueba 1 • El archivo de configuración <code>licenses.xml</code> contiene: <pre><licenses> <license> <name>Software license</name> <url>\${license.url}softwarelicense/license.txt</url> <distribution>repo</distribution> <comments>This is a software license</comments> </license> </licenses></pre> • El archivo de configuración <code>plugin-license.xml</code> contiene: <pre><plugin> <groupId>org.codehaus.mojo</groupId> <artifactId>license-maven-plugin</artifactId></pre>

	<pre> <version>1.0</version> <configuration> <licenseName>license</licenseName> <licenseResolver>\${license.url}</licenseResolver> <bundleLicensePath>META-INF/LICENSE.txt</bundleLicensePath> <generateBundle>>false</generateBundle> </configuration> <executions> <execution> <id>update-project-license</id> <goals> <goal>update-project-license</goal> </goals> <phase>process-sources</phase> </execution> </executions> </plugin> </pre>
Entradas:	Ejecución del comando <code>roo++ project license</code>
Salida esperada:	<p>1. En el archivo <code>pom.xml</code> del módulo web del proyecto, como hijo del nodo <code>/project/build/plugins</code>, se ha introducido el siguiente:</p> <pre> <plugin> <groupId>org.codehaus.mojo</groupId> <artifactId>license-maven-plugin</artifactId> <version>1.0</version> <configuration> <excludedScopes>test,system,provided</excludedScopes> <generateBundle>>false</generateBundle> <groupByLicense>>true</groupByLicense> <useMissingFile>>true</useMissingFile> <useRepositoryMissingFiles>>true</useRepositoryMissingFiles> <deployMissingFile>>true</deployMissingFile> <force>>false</force> <licenseMerges> <licenseMerge><![CDATA[The Apache Software License, Version 2.0 Apache License, Version 2.0 Apache License Version 2.0 Apache 2]]></licenseMerge> <licenseMerge><![CDATA[GNU LESSER GENERAL PUBLIC LICENSE GNU Lesser General Public Licence GNU Lesser General Public License]]></licenseMerge> <licenseMerge><![CDATA[Eclipse Public License v1.0 Eclipse Public License - v 1.0]]></licenseMerge> </licenseMerges> </configuration> <executions> <execution> <id>add-third-party</id> <goals> <goal>add-third-party</goal> </goals> <phase>process-sources</phase> </execution> </executions> </plugin> </pre> <p>2. En el archivo <code>pom.xml</code> del proyecto completo, se ha introducido como hijo de <code>/project/build/plugins</code> el nodo XML contenido en el archivo de configuración <code>plugin-license.xml</code></p> <p>3. En el archivo <code>pom.xml</code> del proyecto completo, se ha introducido como hijo de <code>/project</code> el nodo XML contenido en el archivo de configuración <code>licenses.xml</code></p>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 5

Objetivo:	Cubrir la situación de prueba 3
------------------	---------------------------------

Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1 y 4 • El archivo de configuración <i>locations-application-properties.xml</i> contiene el texto: <pre><value>file:///\${TEST_HOME}/test-core.properties</value> <value>file:///\${TEST_HOME}/test-web.properties</value></pre>
Entradas:	Ejecución del comando <i>roo++ project web configuration</i>
Salida esperada:	<ol style="list-style-type: none"> 1. Se ha creado el archivo <i>src/main/resources/META-INF/spring/application.properties</i>, dentro del módulo <i>web</i> del proyecto 2. En el archivo <i>pom.xml</i> del módulo <i>web</i> del proyecto, el campo <i>useProjectReferences</i> de <i>maven-eclipse-plugin</i> ha sido establecido a <i>false</i>. 3. En el archivo <i>src/main/resources/META-INF/spring/applicationContext.xml</i>, se han introducido los beans <i>messageSource</i>, <i>applicationPropertiesBean</i> y <i>property-placeholder</i>. Además, el segundo de estos beans incluye el contenido de <i>locations-application-properties.xml</i> en su lista de <i>locations</i>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 6

Objetivo:	Cubrir la situación de prueba 5.2.2
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4 y 5
Entradas:	Ejecución del comando <i>roo++ project web deploy</i>
Salida esperada:	<ol style="list-style-type: none"> 1. En el archivo <i>pom.xml</i> del módulo <i>web</i> del proyecto, se ha ajustado la versión del plug-in de Jetty a 6.1.18, y el nombre éste ha cambiado de <i>jetty-maven-plugin</i> a <i>maven-jetty-plugin</i>. 2. En ese mismo plug-in, se ha introducido, como hijos del nodo <i>configuration</i>: <pre><scanIntervalSeconds>10</scanIntervalSeconds> <jettyConfig>src/main/jetty/jetty.xml</jettyConfig> <webAppConfig> <contextPath>/\${project.name}</contextPath> </webAppConfig></pre> 3. En el mismo archivo, en el plug-in de Tomcat, los hijos del nodo <i>configuration</i> son: <pre><path>/\${project.name}</path> <uriEncoding>UTF-8</uriEncoding> <additionalConfigFilesDir>\${basedir}/src/main/tomcat</additionalConfigFilesDir> <systemProperties></pre>

	<pre><com.sun.management.jmxremote.ssl>false</com.sun.management.jmxremote.ssl> </systemProperties></pre> <p>4. Se han creado los diferentes directorios y archivos de configuración de Jetty, no apareciendo en <i>jetty.xml</i> ninguna referencia a bases de datos.</p>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 7

Objetivo:	Cubrir la situación de prueba 5.2.1
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado los casos de prueba 1, 4, 5 y 6
Entradas:	Ejecución del comando <i>roo++ project web deploy</i>
Salida esperada:	No se produce ningún cambio en los archivos del proyecto
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 8

Objetivo:	Cubrir las situaciones de prueba 4.1 y 4.4
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado los casos de prueba 1, 4, y 5 El contenido del archivo de configuración <i>oracle.txt</i> es: 16 Oracle10gDialect El contenido del archivo de configuración <i>oracle_sid.txt</i> es: XE
Entradas:	Ejecución del comando <i>roo++ project web jpa --user username --password userpassword</i>
Salida esperada:	<ol style="list-style-type: none"> En el archivo <i>pom.xml</i> del módulo web del proyecto, se ha introducido la siguiente dependencia respecto al driver de Oracle: <pre><groupId>ojdbc</groupId> <artifactId>ojdbc</artifactId> <version>16</version></pre> Se ha creado el archivo <i>src/main/resources/META-INF/persistenceInfo.xml</i>, dentro del módulo web del proyecto, y no existe el archivo <i>persistence.xml</i> en esa misma ubicación. Se han creado todos los ficheros y directorios de configuración de Tomcat y Jetty. Se ha introducido en los archivos de configuración <i>jetty.xml</i> y <i>context.xml</i> (éste último relativo a Tomcat) la referencia a la base de datos utilizada. En el primero de ellos, la parte relativa a la configuración de la base de datos es: <pre><!-- DataSources CIDI--></pre>

	<pre> <New id="pruebaDS" class="org.mortbay.jetty.plus.naming.Resource"> <Arg>jdbc/pruebaDS</Arg> <Arg> <New class="org.apache.commons.dbcp.BasicDataSource"> <Set name="driverClassName">oracle.jdbc.OracleDriver</Set> <Set name="url">jdbc:oracle:thin:@localhost:1521:XE</Set> <Set name="username">username</Set> <Set name="password">userpassword</Set> </New> </Arg> </New> </pre> <p>Y el contenido del segundo:</p> <pre> <?xml version='1.0' encoding='utf-8'?> <Context> <Resource name="jdbc/pruebaDS" auth="Container" type="javax.sql.DataSource" username="username" password="userpassword" driverClassName="oracle.jdbc.OracleDriver" url="jdbc:oracle:thin:@localhost:1521:XE" maxActive="3" maxIdle="1"/> </Context> </pre> <p>5. En el archivo de configuración <i>applicationContext.xml</i> se ha introducido el bean <i>entityManagerFactory</i> y el <i>dataSource</i> de jndi, tal y como se encuentran en sus correspondientes archivos, dentro de los recursos del JAR de Roo++.</p> <p>6. Se ha creado el archivo <i>database.properties</i>, con el siguiente contenido:</p> <pre> database.driverClassName=oracle.jdbc.OracleDriver database.url=jdbc\:oracle\:thin\:@localhost\:1521 database.username= database.password= #Datasource JNDI jndi.datasource=java:comp/env/jdbc/pruebaDS #Persistence Unit database.persistence.unit=persistenceUnitPrueba #Hibernate properties database.hibernate.default_schema=prueba database.hibernate.dialect=org.hibernate.dialect.Oracle10gDialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100 </pre> <p>7. En el archivo de configuración <i>web.xml</i> es introducida una referencia al <i>dataSource pruebaDS</i>.</p>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 9

Objetivo:	Cubrir las situaciones de prueba 4.2 y 4.3
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4, 5, 6 y 7

	<ul style="list-style-type: none"> El contenido del archivo de configuración <i>mysql.txt</i> es: 5.1.18 MySQL5InnoDBDialect
Entradas:	Ejecución del comando <i>roo++ project web jpa --user username --password userpassword --dbms mysql --server 10.0.0.1 --hibernateversion 3.6.9.Final --port 5555 --database databasename</i>
Salida esperada:	<ol style="list-style-type: none"> En el archivo <i>pom.xml</i> del módulo web del proyecto, se ha introducido la siguiente dependencia respecto al driver de MySQL: <pre><groupId>mysql</groupId> <artifactId>mysql-connector-java</artifactId> <version>5.1.18</version></pre> Además, en ese mismo archivo, las dependencias cuyos <i>artifactId</i> son <i>hibernate-core</i> e <i>hibernate-entitymanager</i> tienen como versión <i>3.6.9.Final</i>. Se ha creado el archivo <i>src/main/resources/META-INF/persistenceInfo.xml</i>, dentro del módulo web del proyecto, y no existe el archivo <i>persistence.xml</i> en esa misma ubicación. Se ha introducido en los archivos de configuración <i>jetty.xml</i> y <i>context.xml</i> (éste último relativo a Tomcat) la referencia a la base de datos utilizada. En el primero de ellos, la parte relativa a la configuración de la base de datos es: <pre><!-- DataSources CIDI--> <New id="pruebaDS" class="org.mortbay.jetty.plus.naming.Resource"> <Arg>jdbc/pruebaDS</Arg> <Arg> <New class="org.apache.commons.dbcp.BasicDataSource"> <Set name="driverClassName">com.mysql.jdbc.Driver</Set> <Set name="url">jdbc:mysql://10.0.0.1:5555/databasename</Set> <Set name="username">username</Set> <Set name="password">userpassword</Set> </New> </Arg> </New></pre> Y el contenido del segundo: <pre><?xml version='1.0' encoding='utf-8'?> <Context> <Resource name="jdbc/pruebaDS" auth="Container" type="javax.sql.DataSource" username="username" password="userpassword" driverClassName="com.mysql.jdbc.Driver" url="jdbc:mysql://10.0.0.1:5555/databasename" maxActive="3" maxIdle="1"/> </Context></pre> En el archivo de configuración <i>applicationContext.xml</i> se ha introducido el bean <i>entityManagerFactory</i> y el <i>dataSource</i> de jndi, tal y como se encuentran en sus correspondientes archivos, dentro de los recursos del JAR de Roo++. Se ha creado el archivo <i>database.properties</i>, con el siguiente contenido: <pre>database.driverClassName=com.mysql.jdbc.Driver database.url=jdbc:mysql://localhost:3306 database.username= database.password= #DataSource JNDI</pre>

	<pre>jndi.datasource=java:comp/env/jdbc/pruebaDS #Persistence Unit database.persistence.unit=persistenceUnitPrueba #Hibernate properties database.hibernate.default_schema=databasename database.hibernate.dialect=org.hibernate.dialect.MySQL5InnoDBDialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100</pre> <p>6. En el archivo de configuración <i>web.xml</i> es introducida una referencia al <i>dataSource pruebaDS</i>.</p>
Salida obtenida:	Respecto a la salida esperada, en <i>database.properties</i> , en lugar de: <pre>database.url=jdbc:mysql://10.0.0.1:5555</pre> aparece: <pre>database.url=jdbc:mysql://localhost:3306</pre>
Fallos detectados:	Fallo 1

Caso de prueba 10

Objetivo:	Cubrir la situación de prueba 4.6
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4, 5, 6 y 7 • El contenido del archivo de configuración <i>mssql.txt</i> es: <pre>1.2.4 SQLServer2008Dialect</pre>
Entradas:	Ejecución del comando <i>roo++ project web jpa --user username --password userpassword --dbms mssql</i>
Salida esperada:	<ol style="list-style-type: none"> 1. En el archivo <i>pom.xml</i> del módulo web del proyecto, se ha introducido la siguiente dependencia respecto al driver de Microsoft SQL Server: <pre><groupId>net.sourceforge.jtds</groupId> <artifactId>jtds</artifactId> <version>1.2.4</version></pre> 2. Se ha creado el archivo <i>src/main/resources/META-INF/persistenceInfo.xml</i>, dentro del módulo web del proyecto, y no existe el archivo <i>persistence.xml</i> en esa misma ubicación. 3. Se ha introducido en los archivos de configuración <i>jetty.xml</i> y <i>context.xml</i> (éste último relativo a Tomcat) la referencia a la base de datos utilizada. En el primero de ellos, la parte relativa a la configuración de la base de datos es: <pre><!-- DataSources CIDI--> <New id="pruebaDS" class="org.mortbay.jetty.plus.naming.Resource"> <Arg>jdbc/pruebaDS</Arg> <Arg> <New class="org.apache.commons.dbcp.BasicDataSource"> <Set name="driverClassName">net.sourceforge.jtds.jdbc.Driver</Set> <Set name="url">jdbc:jtds:sqlserver://localhost:1433/prueba</Set></pre>

	<pre> <Set name="username">username</Set> <Set name="password">userpassword</Set> </New> </Arg> </New> </pre> <p>Y el contenido del segundo:</p> <pre> <?xml version='1.0' encoding='utf-8'?> <Context> <Resource name="jdbc/pruebaDS" auth="Container" type="javax.sql.DataSource" username="username" password="userpassword" driverClassName="net.sourceforge.jtds.jdbc.Driver" url="jdbc:jtds:sqlserver://localhost:1433/prueba" maxActive="3" maxIdle="1"/> </Context> </pre> <p>4. En el archivo de configuración <i>applicationContext.xml</i> se ha introducido el bean <i>entityManagerFactory</i> y el <i>dataSource</i> de jndi, tal y como se encuentran en sus correspondientes archivos, dentro de los resources del JAR de Roo++.</p> <p>5. Se ha creado el archivo <i>database.properties</i>, con el siguiente contenido:</p> <pre> database.driverClassName=net.sourceforge.jtds.jdbc.Driver database.url=jdbc:jtds:sqlserver://localhost:1433/prueba-web database.username= database.password= #Datasource JNDI jndi.datasource=java:comp/env/jdbc/pruebaDS #Persistence Unit database.persistence.unit=persistenceUnitPrueba #Hibernate properties database.hibernate.default_schema=prueba database.hibernate.dialect=org.hibernate.dialect.SQLServer2008Dialect database.hibernate.hbm2ddl.auto=validate database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy database.hibernate.connection.charset=UTF-8 database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100 </pre> <p>6. En el archivo de configuración <i>web.xml</i> es introducida una referencia al <i>dataSource pruebaDS</i>.</p>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 11

Objetivo:	Cubrir la situación de prueba 4.5
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4, 5, 6 y 7 • El contenido del archivo de configuración <i>postgresql.txt</i> es: 9.0-801.jdbc4 PostgreSQLDialect
Entradas:	Ejecución del comando <i>roo++ project web jpa --user username --password userpassword --dbms postgres</i>

Salida esperada:

1. En el archivo *pom.xml* del módulo web del proyecto, se ha introducido la siguiente dependencia respecto al driver de PostgreSQL:

```
<groupId>postgresql</groupId>
<artifactId>postgresql</artifactId>
<version>9.0-801.jdbc4</version>
```

2. Se ha creado el archivo *src/main/resources/META-INF/persistenceInfo.xml*, dentro del módulo web del proyecto, y no existe el archivo *persistence.xml* en esa misma ubicación.

3. Se ha introducido en los archivos de configuración *jetty.xml* y *context.xml* (éste último relativo a Tomcat) la referencia a la base de datos utilizada. En el primero de ellos, la parte relativa a la configuración de la base de datos es:

```
<!-- DataSources CIDI-->

<New id="pruebaDS" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/pruebaDS</Arg>
  <Arg>
    <New class="org.apache.commons.dbcp.BasicDataSource">
      <Set
name="driverClassName">org.postgresql.Driver</Set>
      <Set
name="url">jdbc:postgresql://localhost:5432/prueba</Set>
      <Set name="username">username</Set>
      <Set name="password">userpassword</Set>
    </New>
  </Arg>
</New>
```

Y el contenido del segundo:

```
<?xml version='1.0' encoding='utf-8'?>
<Context>
  <Resource name="jdbc/pruebaDS"
    auth="Container"
    type="javax.sql.DataSource"
    username="username"
    password="userpassword"
    driverClassName="org.postgresql.Driver"
    url="jdbc:postgresql://localhost:5432/prueba"
    maxActive="3" maxIdle="1"/>
</Context>
```

4. En el archivo de configuración *applicationContext.xml* se ha introducido el bean *entityManagerFactory* y el *dataSource* de jndi, tal y como se encuentran en sus correspondientes archivos, dentro de los recursos del JAR de Roo++.

5. Se ha creado el archivo *database.properties*, con el siguiente contenido:

```
database.driverClassName=org.postgresql.Driver
database.url=jdbc:postgresql://localhost:5432
database.username=
database.password=

#Datasource JNDI
jndi.datasource=java:comp/env/jdbc/pruebaDS

#Persistence Unit
database.persistence.unit=persistenceUnitPrueba

#Hibernate properties
database.hibernate.default_schema=prueba
database.hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect
database.hibernate.hbm2ddl.auto=validate

database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy
database.hibernate.connection.charset=UTF-8
```

	<pre>database.hibernate.jdbc.fetch_size=100 database.hibernate.jdbc.batch_size=100</pre> <p>6. En el archivo de configuración <i>web.xml</i> es introducida una referencia al <i>dataSource pruebaDS</i>.</p>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 12

Objetivo:	Cubrir la situación de prueba 5.1.1
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado los casos de prueba 1, 4, 5, 6, 7 y 9
Entradas:	Ejecución del comando <i>roo++ project web deploy</i>
Salida esperada:	<ol style="list-style-type: none"> Se ha añadido la siguiente dependencia al plug-in de Tomcat existente en el archivo <i>pom.xml</i> del módulo web: <pre><groupId>mysql</groupId> <artifactId>mysql-connector-java</artifactId> <version>5.1.18</version></pre> En el caso del plug-in de Jetty, en ese mismo archivo, se han introducido las siguientes dependencias: <pre><groupId>mysql</groupId> <artifactId>mysql-connector-java</artifactId> <version>5.1.18</version> <scope> runtime </scope> <groupId>commons-dbcp</groupId> <artifactId>commons-dbcp</artifactId> <version>1.4</version> <scope>runtime</scope></pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 13

Objetivo:	Cubrir la situación de prueba 5.1.2
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado los casos de prueba 1, 4, 5 y 8
Entradas:	Ejecución del comando <i>roo++ project web deploy</i>
Salida esperada:	<ol style="list-style-type: none"> En el archivo <i>pom.xml</i> del módulo web del proyecto, se ha ajustado la versión del plug-in de Jetty a 6.1.18, y el nombre éste ha cambiado de <i>jetty-maven-plugin</i> a <i>maven-jetty-plugin</i>. En ese mismo plug-in, se ha introducido, como hijos del nodo <i>configuration</i>: <pre><scanIntervalSeconds>10</scanIntervalSeconds> <jettyConfig>src/main/jetty/jetty.xml</jettyConfig> <webAppConfig> <contextPath>/\${project.name}</contextPath> </webAppConfig></pre> En el mismo archivo, en el plug-in de Tomcat, los hijos del nodo

	<pre> configuration son: <path>/\${project.name}</path> <uriEncoding>UTF-8</uriEncoding> <additionalConfigFilesDir>\$ {basedir}/src/main/tomcat</additionalConfigFilesDir> <systemProperties> <com.sun.management.jmxremote.ssl>>false</com.sun.management.jmxremote.ssl> </systemProperties> 4. Se ha añadido la siguiente dependencia a los plug-in de Tomcat existente en el archivo <i>pom.xml</i> del módulo web: <groupId>ojdbc</groupId> <artifactId>ojdbc</artifactId> <version>16</version> 5. En el caso del plug-in de Jetty, se han introducido las siguientes dependencias: <groupId>ojdbc</groupId> <artifactId>ojdbc</artifactId> <version>16</version> <scope>runtime</scope> <groupId>commons-logging</groupId> <artifactId>commons-logging</artifactId> <version>1.4</version> <scope>runtime</scope> </pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 14

Objetivo:	Cubrir la situación de prueba 9
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9 y 12 El archivo de configuración <i>platform.txt</i> tiene el siguiente contenido: plataforma 2.0 test-module
Entradas:	Ejecución del comando <i>roo++ platform setup</i>
Salida esperada:	<ol style="list-style-type: none"> En el archivo <i>pom.xml</i> del proyecto, se ha introducido la definición: <platform.version>2.0</platform.version> En el archivo <i>pom.xml</i> del módulo core del proyecto, se han introducido las siguientes dependencias: <groupId>plataforma</groupId> <artifactId>plataforma-core</artifactId> <version>\${platform.version}</version> <groupId>plataforma</groupId> <artifactId>plataforma-test-module</artifactId> <version>\${platform.version}</version> En el archivo <i>pom.xml</i> del módulos web del proyecto, se han introducido las siguientes dependencias: <groupId>plataforma</groupId> <artifactId>plataforma-web</artifactId> <version>\${platform.version}</version> <groupId>plataforma</groupId>

	<pre><artifactId>plataforma-test-module</artifactId> <version>\${platform.version}</version></pre>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 15

Objetivo:	Cubrir la situación de prueba 10
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <code>project --topLevelPackage com.pruebas.prueba</code> • Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12 y 14
Entradas:	Ejecución del comando <code>roo++ plataforma configuration</code>
Salida esperada:	<ol style="list-style-type: none"> 1. El archivo <code>applicationContext-platform.xml</code> ha sido creado, y tiene exactamente el mismo contenido que el archivo de configuración del mismo nombre. 2. En los archivos <code>applicationContext.xml</code> y <code>webmvc-config.xml</code> del proyecto, se ha incluido <code>plataforma</code> como <code>base-package</code> en <code>component-scan</code>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 16

Objetivo:	Cubrir la situación de prueba 12.1
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <code>project --topLevelPackage com.pruebas.prueba</code> • Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12, 14 y 15 • El directorio de configuración <code>usermgmt</code> existe, y contiene todos los archivos correspondientes • El archivo de configuración <code>usermgmt/usermgmt.txt</code> contiene: <pre>com.empresa.usermgmt usermgmt-client 1.0 http://cas.empresa.com:8080 appname</pre> • El archivo de configuración <code>usermgmt/applicationContext-security.xml</code> contiene: <pre><beans> </beans></pre> • El contenido del archivo de configuración <code>usermgmt/artifactItems.txt</code> es: <pre>usermgmt-application war \${usermgmt.delivery}.war usermgmt-doc zip \${usermgmt.delivery}-docs.zip</pre>
Entradas:	Ejecución del comando <code>roo++ usermgmt setup</code>

Salida esperada:

1. En el archivo *pom.xml* del módulo *package* del proyecto, se han introducido los siguientes nodos:

```
<artifactItem>
  <groupId>com.empresa.usermgmt </groupId>
  <artifactId>usermgmt-application </artifactId>
  <version>${usermgmt.version}</version>
  <type>war</type>
  <destFileName>${usermgmt.delivery}.war </destFileName>
</artifactItem>
<artifactItem>
  <groupId>com.empresa.usermgmt </groupId>
  <artifactId>usermgmt-doc</artifactId>
  <version>${usermgmt.version}</version>
  <type>zip</type>
  <destFileName>${usermgmt.delivery}-docs.zip</destFileName>
</artifactItem>
```

2. Al final del archivo *application.properties* ha sido introducido el contenido del archivo de configuración

usermgmt/application.properties

3. En el archivo *applicationContext.xml*, se han establecido las ubicaciones de los mensajes de localización de la siguiente manera:

```
p:basenames="WEB-INF/i18n/messages,WEB-INF/i18n/application, WEB-
INF/i18n/portal, classpath:/META-INF/web-
resources/i18n/messagesPortalContext, classpath:/META-INF/web-
resources/i18n/messagesUsermgmtClient, classpath:/META-INF/web-
resources/i18n/messagesUsermgmtContext, classpath:/META-INF/web-
resources/i18n/messagesUsermgmtAcl, classpath:/META-INF/web-
resources/i18n/service_messages"
```

Además, en este mismo archivo, se ha añadido

com.empresa.usermgmt al campo *base-package*.

4. El archivo *applicationContext-security.xml* ha sido copiado al proyecto desde el archivo de configuración del mismo nombre.

5. En el archivo *webmvc-config.xml*, se ha añadido *com.empresa.usermgmt* al campo *base-package*, y se ha introducido la misma referencia a las ubicaciones de mensajes de localización que se ha introducido también en *applicationContext.xml*. Además, se ha añadido a la lista de propiedades del bean *tilesConfigurer* la siguiente:

```
<value>classpath*/META-INF/web-
resources/views/**/views.xml</value>
```

6. En el módulo web del proyecto se ha creado el directorio *src/main/webapp/WEB-INF/views/users*, y se han introducido en el mismo copias de los archivos de configuración *changePassword.jspx* y *userInfo.jspx*.

7. Se ha creado la vista *casfailure.jspx*, que tiene el contenido del archivo con el mismo nombre que se encuentra en los “resources” del JAR de Roo++.

8. En la vista *footer.jspx*, la URL de cierre de sesión ha pasado a ser */j_spring_security_logout*, en lugar de */resources/j_spring_security_logout*.

9. En el archivo *views.xml*, se ha introducido la siguiente vista:

```
<definition name="casFailure" extends="public">
  <put-attribute name="body" value="/WEB-INF/views/casfailure.jspx"
/>
</definition>
```

10. En el archivo *web.xml* se han introducido los siguientes nodos:

```
<filter>
  <filter-name>springSecurityFilterChain</filter-name>
<filter-
```

	<pre> class>org.springframework.web.filter.DelegatingFilterProxy</filter-class> </filter> <filter-mapping> <filter-name>springSecurityFilterChain</filter-name> <url-pattern>/*</url-pattern> </filter-mapping> </pre> <p>11. En el archivo <i>pom.xml</i> del módulo web se ha introducido la siguiente dependencia:</p> <pre> <groupId>com.empresa.usermgmt<groupId> <artifactId>usermgmt-client</artifactId> <version>\${usermgmt.version}</version> </pre> <p>12. En el archivo <i>pom.xml</i> raíz del proyecto se han introducido las propiedades <i>usermgmt.version</i> y <i>usermgmt.delivery</i>, cuyos valores son <i>1.0</i> y <i>appName</i>, respectivamente.</p>
Salida obtenida:	Igual a la esperada en todos los puntos excepto el segundo de ellos; el contenido de <i>usermgmt/application.properties</i> ha sido introducido en el archivo <i>application.properties</i> del proyecto justo a continuación del texto existente, sin ningún salto de línea intermedio.
Fallos detectados:	Fallo 2

Caso de prueba 17

Objetivo:	Cubrir la situación de prueba 12.2
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12, 14 y 15 • El directorio de configuración <i>usermgmt</i> no existe
Entradas:	Ejecución del comando <i>roo++ usermgmt setup</i>
Salida esperada:	<ol style="list-style-type: none"> 1. Al final del archivo <i>application.properties</i> ha sido introducido el contenido del archivo de configuración <i>usermgmt_application.properties</i> 2. El archivo <i>applicationContext.xml</i> no ha sido modificado. 3. El archivo <i>applicationContext-security.xml</i> ha sido creado, y tiene el contenido del archivo del mismo nombre situado en los “resources” del JAR de Roo++, incluyendo además la definición de los nombres de usuario y contraseñas indicados en el archivo de configuración <i>usermgmt_users.txt</i>. 4. El archivo <i>webmvc-config.xml</i> no ha sido modificado. 5. En el módulo web del proyecto no existe el directorio <i>src/main/webapp/WEB-INF/views/users</i>. 6. Se ha creado la vista <i>casfailure.jspx</i>, que tiene el contenido del archivo con el mismo nombre que se encuentra en los “resources” del JAR de Roo++. 7. En la vista <i>footer.jspx</i>, la URL de cierre de sesión ha pasado a ser <i>/j_spring_security_logout</i>, en lugar de <i>/resources/j_spring_security_logout</i>. 8. En el archivo <i>views.xml</i>, se ha introducido la siguiente vista: <pre> <definition name="casFailure" extends="public"> <put-attribute name="body" value="/WEB-INF/views/casfailure.jspx" /> </definition> </pre>

	<p>9. En el archivo <i>web.xml</i> se han introducido los siguientes nodos:</p> <pre><filter> <filter-name>springSecurityFilterChain</filter-name> <filter- class>org.springframework.web.filter.DelegatingFilterProxy</filter-class> </filter> <filter-mapping> <filter-name>springSecurityFilterChain</filter-name> <url-pattern>/*</url-pattern> </filter-mapping></pre> <p>10. En el archivo <i>pom.xml</i> del módulo web se han introducido las siguientes dependencias:</p> <pre><groupId>org.springframework.security <groupId> <artifactId>spring-security-core </artifactId> <version>3.1.3.RELEASE </version> <groupId>org.springframework.security <groupId> <artifactId>spring-security-config </artifactId> <version>3.1.3.RELEASE </version> <groupId>org.springframework.security <groupId> <artifactId>spring-security-web </artifactId> <version>3.1.3.RELEASE </version> <groupId>org.springframework.security <groupId> <artifactId>spring-security-cas</artifactId> <version>3.1.3.RELEASE </version></pre> <p>11. El archivo <i>pom.xml</i> raíz del proyecto no ha sido modificado.</p>
Salida obtenida:	Igual a la esperada en todos los puntos excepto el primero de ellos; el contenido de <i>usermgmt_application.properties</i> ha sido introducido en el archivo <i>application.properties</i> del proyecto justo a continuación del texto existente, sin ningún salto de línea intermedio.
Fallos detectados:	Fallo 3

Caso de prueba 18

Objetivo:	Cubrir la situación de prueba 11
Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12, 14, 15 y 16 • El contenido del archivo de configuración <i>portal.txt</i> es: com.empresa.portal portal-client 1.0
Entradas:	Ejecución del comando <i>roo++ portal setup</i>
Salida esperada:	<ol style="list-style-type: none"> 1. El archivo de configuración <i>portal.properties</i> es copiado al proyecto en desarrollo 2. El archivo de configuración <i>applicationContext-portal.xml</i> es copiado al proyecto en desarrollo 3. El contenido del archivo de configuración <i>messages.properties</i> es introducido al final en el archivo del mismo nombre situado en el proyecto en desarrollo 4. En el archivo <i>applicationContext.xml</i>, se ha configurado que el classpath incluya los directorios del módulo web del proyecto <i>resources/META-INF</i> y <i>resources/META-INF/spring</i> 5. En el archivo <i>webmvc-config.xml</i> del módulo web, se ha introducido: <mvc:view-controller path="/index" view-name="index"/>

	<p>6. El contenido del archivo <i>header.jspx</i> del proyecto ha sido reemplazado por el del archivo de configuración <i>portal-header.jspx</i></p> <p>7. En el archivo <i>load-scripts.tagx</i> del proyecto, se ha introducido en su cabecera la URL contenida en el archivo de configuración <i>portal-url.txt</i>, se ha movido el nodo <i>portal:load-scripts</i> hacia su parte inferior, y se ha modificado este nodo de forma que su contenido sea: <code><portal:load-scripts noIncludeDojo="true"/></code></p> <p>8. En el archivo <i>pom.xml</i> del módulo web del proyecto, ha sido introducida la siguiente dependencia: <pre><groupId>com.empresa.portal </groupId> <artifactId>portal-client </artifactId> <version>\${portal.version}</version></pre></p> <p>9. En el archivo <i>pom.xml</i> raíz del proyecto, ha sido introducido el campo: <code><portal.version>1.0</portal.version></code></p>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 19

Objetivo:	Cubrir la situación de prueba 8
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <code>project --topLevelPackage com.pruebas.prueba</code> Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12, 14, 15, 16 y 18
Entradas:	Ejecución del comando <code>roo++ layout setup</code>
Salida esperada:	<ol style="list-style-type: none"> En el archivo <i>resources/META-INF/spring/application.properties</i>, es introducida la nueva propiedad: <code>application.theme=standard</code> En la carpeta <i>webapp</i>, los directorios <i>images</i>, <i>scripts</i>, <i>styles</i>, <i>WEB-INF/classes</i>, <i>WEB-INF/i18n</i> y <i>WEB-INF/tags</i> contienen copias de los archivos contenidos en los directorios de configuración <i>images</i>, <i>scripts</i>, <i>styles</i>, <i>classes</i>, <i>properties</i>, y <i>tags</i>, respectivamente. En el archivo <i>webmvc-config.xml</i>, el nombre del bean <i>CookieThemeResolver</i> ha cambiado a <i>FixedThemeResolver</i>. En el archivo <i>footer.jspx</i>, se ha eliminado el nodo <code><util:theme/></code>. En el archivo <i>load-scripts.tagx</i> del proyecto es introducido al final el contenido del archivo de configuración <i>load-scripts.jspx</i>.
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 20

Objetivo:	Cubrir la situación de prueba 6
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <code>project --topLevelPackage com.pruebas.prueba</code> Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12, 14, 15, 16, 18 y 19

	<ul style="list-style-type: none"> El contenido del archivo de configuración <i>breadcrumbs-url.txt</i> es: <code>http://www.empresa.com/breadcrumbs</code>
Entradas:	Ejecución del comando <i>roo++ web breadcrumb setup</i>
Salida esperada:	<ol style="list-style-type: none"> En el archivo <i>pom.xml</i> del módulo web se ha introducido la dependencia contenida en el archivo de configuración <i>breadcrumb-dependency.xml</i> El nodo <i>bean</i> contenido en el archivo de configuración <i>breadcrumbs-beans.xml</i> es introducido en el archivo <i>applicationContext-platform.xml</i>, en el módulo web del proyecto. Además, a dicho bean se le ha añadido como nodo hijo <code><property name="scanPackage" value="com.pruebas.prueba"/></code> En el archivo de layout llamado <i>default.jspx</i>, se ha introducido el texto: <code><breadcrumbs:breadcrumbs parentPath=\\${parentPath}\/></code> Además, en ese mismo archivo, en su cabecera, se ha introducido: <code>xmlns:breadcrumbs="http://www.empresa.com/breadcrumbs"</code>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 21

Objetivo:	Cubrir la situación de prueba 7
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado los casos de prueba 1, 4, 5, 6, 7, 9, 12, 14, 15, 16, 18, 19 y 20 El directorio de configuración <i>nodesToAdd</i> está vacío. El contenido del archivo de configuración <i>context-dependencies.txt</i> es el siguiente: <code>com.empresa.usermgmt</code> <code>usermgmt-contexts</code> <code>1.0</code>
Entradas:	Ejecución del comando <i>roo++ context setup</i>
Salida esperada:	<ol style="list-style-type: none"> El archivo de configuración <i>context-dataload.sql</i> ha sido copiado al módulo <i>scripts</i> del proyecto en desarrollo Se han introducido en el archivo <i>applicationContext-portal.xml</i> todos los nodos XML correspondientes a beans, contenidos en archivos de configuración en el directorio <i>contextBeans/portal</i> En el archivo <i>applicationContext-security.xml</i> se han introducido todos los beans contenidos en archivos de configuración situados en el directorio <i>contextBeans/security</i>, el nodo cuyo atributo <i>name</i> es <i>authenticationSuccessHandler</i> tiene el valor <i>contextAuthenticationSuccessHandler</i> en el atributo <i>ref</i>, mientras que <i>logout-success-url</i> ha sido eliminado. En el archivo <i>portal.properties</i> se ha introducido al final el contenido del archivo de configuración <i>context-portal.properties</i> En la parte final del archivo <i>messages.properties</i> se ha introducido

	<p>el contenido de <i>context-messages.properties</i></p> <ol style="list-style-type: none"> Se ha creado el directorio <i>src/main/webapp/WEB-INF/views/contexts</i> , dentro del módulo web del proyecto, y contiene una copia de cada uno de los archivos de configuración situados en el directorio <i>contextViews</i>. En el archivo <i>header.jspx</i>, se han añadido al nodo <i>portal:portal-header</i> las asignaciones de variables contenidas en el archivo de configuración <i>add-to-portal-header.txt</i> , y se han introducido los nodos contenidos en los archivos de configuración <i>directive-pages.jspx</i> y <i>scriptlets.jspx</i> . En el archivo <i>pom.xml</i> del módulo web se ha introducido la siguiente dependencia: <pre><groupId>com.empresa.usermgmt </groupId> <artifactId>usermgmt-contexts </artifactId> <version>1.0</version></pre>
Salida obtenida:	Se obtiene el siguiente mensaje de error (excepción): <pre>NullPointerException at rooplusplus.roo.addon.context.ContextOperationsImpl.editApplicationContextPortal (ContextOperationsImpl.java:236)</pre>
Fallos detectados:	Fallo 4

Caso de prueba 22

Objetivo:	Cubrir la situación de prueba 13
Condiciones:	<ul style="list-style-type: none"> Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> Haber ejecutado el caso de prueba 3, y a continuación los comandos <i>roo++ project web configuration</i> y <i>roo++ project web deploy</i>.
Entradas:	Compilación del proyecto mediante la ejecución de la orden <i>mvn clean install -Dmaven.test.skip=true</i> en la interfaz de línea de comandos del sistema operativo
Salida esperada:	<ol style="list-style-type: none"> Dentro del archivo comprimido <i>prueba-package/target/prueba-package-2.4.5-zip.zip</i>, se encuentra la aplicación desarrollada, preparada para ser desplegada en un servidor de aplicaciones (como JBoss, Tomcat o Jetty), concretamente en <i>prueba-web.ear</i> En ese mismo archivo ZIP también se encuentra el archivo <i>prueba-docs.zip</i>, que contiene los siguientes archivos: <ul style="list-style-type: none"> - <i>InstallationManual.pdf</i>: su contenido es el indicado en el archivo de configuración <i>InstallationManual.xml</i> - <i>UserManual.pdf</i>: su contenido es el indicado en el archivo de configuración <i>UserManual.xml</i>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Caso de prueba 23

Objetivo:	Cubrir la situación de prueba 4.7
------------------	-----------------------------------

Condiciones:	<ul style="list-style-type: none"> • Haber ejecutado el comando <i>project --topLevelPackage com.pruebas.prueba</i> • Haber ejecutado los casos de prueba 1, 4, y 5 • El contenido del archivo de configuración <i>oracle.txt</i> es: 16 Oracle10gDialect • El contenido del archivo de configuración <i>oracle_sid.txt</i> es: XE
Entradas:	Ejecución del comando <i>roo++ project web jpa</i>
Salida esperada:	Mensaje de error de Roo: <i>You must specify option 'user' for this command</i>
Salida obtenida:	Igual a la esperada
Fallos detectados:	Ninguno

Fallos detectados en pruebas de sistema

Fallo 1

Caso de prueba:	9
Descripción	En el archivo <i>database.properties</i> , en la línea <i>database.url</i> , se indica el puerto por defecto de MySQL (3306) en lugar del que el usuario ha seleccionado (5555), y la máquina local (localhost) en lugar de la seleccionada (10.0.0.1).
Solución:	Creación del método <i>setDbPortServer</i> , en la clase <i>WebModuleUtils</i> , que es invocado cuando el usuario ejecuta el comando <i>web jpa</i> indicando un puerto o máquina diferentes de sus valores por defecto, y que ajusta sus correspondientes valores reales en <i>database.properties</i> .

Fallo 2

Caso de prueba:	16
Descripción	Las propiedades relacionadas con la gestión de usuarios son introducidas en el archivo <i>application.properties</i> justo a continuación del último texto existente (es decir, en la misma línea que éste), y además con un salto de línea adicional al final.
Solución:	En el método privado <i>usermgmtApplicationPropertiesCustom</i> , en la clase <i>UsermgmtUtils</i> , introducir las nuevas propiedades utilizando el método <i>print</i> en lugar de <i>println</i> , e incluir un salto de línea antes de las mismas,

Fallo 3

Caso de prueba:	17
Descripción	Las propiedades relacionadas con la gestión de usuarios son introducidas en el archivo <i>application.properties</i> justo a continuación del último texto existente (es decir, en la misma línea que éste), y además con un salto de línea adicional al final.
Solución:	En el método privado <i>usermgmtApplicationPropertiesSpring</i> , en la clase <i>UsermgmtUtils</i> , introducir las nuevas propiedades utilizando el método <i>print</i> en lugar de <i>println</i> , e incluir un salto de línea antes de las mismas,

Fallo 4

Caso de prueba:	21
Descripción	Se produce una <i>NullPointerException</i> en la siguiente línea del archivo <i>ContextOperationsImpl.java</i> : <pre>for (int i = 0; i < list.length; i++)</pre>
Solución:	Impedir la ejecución de dicha línea si <i>list</i> es <i>null</i> .

Trazabilidad entre requisitos del sistema y casos de prueba

A continuación se indica los requisitos funcionales del sistema que son cubiertos por cada uno de los casos de prueba (ver el Estudio de Viabilidad del Sistema para más información sobre dichos requisitos):

Requisito funcional		Situaciones de prueba que cubren el requisito		Caso de prueba que cubre la situación de prueba
Id	Descripción	Id	Descripción	
1.1	Se debe permitir la creación de proyectos Maven con múltiples módulos	1.1	Ejecución del comando <i>roo++ project create modules</i> sin especificar número de versión para el proyecto.	1
		1.2	Ejecución del comando <i>roo++ project create modules</i> especificando número de versión para el proyecto.	2
		1.3	Ejecución del comando <i>roo++ project create modules</i> cuando existe el archivo de configuración <i>parent.txt</i> , y tiene el contenido esperado.	1
		1.4	Ejecución del comando <i>roo++ project create modules</i> cuando existe el archivo de configuración <i>parent.txt</i> , pero no tiene el contenido esperado.	2
		1.5	Ejecución del comando <i>roo++ project create modules</i> cuando no existe el archivo de configuración <i>parent.txt</i> .	3
1.2	Se debe permitir especificar el número de versión del proyecto a desarrollar	1.2	Ejecución del comando <i>roo++ project create modules</i> especificando	2

			número de versión para el proyecto.	
1.2.1	Si no se especifica el número de versión del proyecto a desarrollar, éste debe ser 1.0.BUILD-SNAPSHOT	1.1	Ejecución del comando <i>roo++ project create modules</i> sin especificar número de versión para el proyecto.	1
1.3	Tras la compilación del proyecto, la aplicación web ha de encontrarse en un archivo EAR listo para su despliegue en un servidor web	13	Compilación del proyecto	22
1.4	Se debe permitir la introducción de licencias en el proyecto en desarrollo	2	Ejecución del comando <i>roo++ project license</i>	4
1.5	Se debe permitir preparar un proyecto para su despliegue en los servidores web Tomcat y Jetty durante su desarrollo, con el fin de realizar pruebas	5.1.1	Ejecución del comando <i>roo++ project web deploy</i> , cuando se habían ejecutado previamente los comandos <i>roo++ project web jpa</i> y <i>roo++project web deploy</i>	12
		5.1.2	Ejecución del comando <i>roo++ project web deploy</i> , cuando se había ejecutado previamente el comando <i>roo++ project web jpa</i> , pero no <i>roo++project web deploy</i>	13
		5.2.1	Ejecución del comando <i>roo++ project web deploy</i> , cuando no se había ejecutado previamente el comando <i>roo++ project web jpa</i> , pero sí <i>roo++project web deploy</i>	7
		5.2.2	Ejecución del comando <i>roo++ project web deploy</i> , cuando no se había ejecutado previamente ninguno de los comandos <i>roo++ project web jpa</i> y <i>roo++project web deploy</i>	6
1.6	Se debe permitir la generación automática de los manuales de instalación y de usuario de la aplicación, mediante	13	Compilación del proyecto	22

	DocBook, a partir de una serie de archivos fuente			
2.1	Se debe permitir configurar la conexión de la aplicación en desarrollo con bases de datos Oracle, MySQL, PostgreSQL y Microsoft SQL Server	4.3	Ejecución del comando <i>roo++ project web jpa</i> , indicando MySQL como sistema de gestión de bases de datos	9
		4.4	Ejecución del comando <i>roo++ project web jpa</i> , indicando Oracle como sistema de gestión de bases de datos	8
		4.5	Ejecución del comando <i>roo++ project web jpa</i> , indicando PostgreSQL como sistema de gestión de bases de datos	11
		4.6	Ejecución del comando <i>roo++ project web jpa</i> , indicando Microsoft SQL Server como sistema de gestión de bases de datos	10
2.1.1	Se debe permitir configurar el número de versión del sistema de gestión de bases de datos a utilizar, así como el nombre de su dialecto SQL	4.2	Ejecución del comando <i>roo++ project web jpa</i> , especificando todos los parámetros opcionales.	9
2.1.2	Si el usuario no indica el sistema de gestión de bases de datos a utilizar por la aplicación en desarrollo, éste ha de ser Oracle	4.1	Ejecución del comando <i>roo++ project web jpa</i> , no especificando ninguno de los parámetros opcionales.	8
2.2	El usuario ha de indicar obligatoriamente el nombre de usuario y la contraseña necesarios para el acceso a la base de datos, no permitiéndose su almacenamiento en ningún archivo de configuración	4.7	Ejecución del comando <i>roo++ project web jpa</i> , no especificando ningún parámetro.	23
2.3	Se ha de permitir indicar en qué máquina y puerto se encuentra la base de datos que la aplicación en desarrollo ha de utilizar, además de su nombre	4.2	Ejecución del comando <i>roo++ project web jpa</i> , especificando todos los parámetros opcionales.	9
3.1	Se debe permitir configurar el proyecto en desarrollo para el uso de utilidades proporcionadas por una plataforma software usada por la empresa	9	Ejecución del comando <i>roo++ platform setup</i>	14
		10	Ejecución del comando <i>roo++ platform</i>	15

			<i>configuration</i>	
3.2	Se debe permitir la integración de la aplicación web en desarrollo dentro del portal web de la organización	11	Ejecución del comando <i>roo++ portal setup</i>	18
3.3	Todo lo relativo a las particularidades de la plataforma software de la organización y a cualquier uso que se haga de ella ha de encontrarse establecido en archivos de configuración	9	Ejecución del comando <i>roo++ platform setup</i>	14
4.1	Se debe permitir establecer un sistema de gestión de usuarios para la aplicación en desarrollo	12.1	Ejecución del comando <i>roo++ usermgmt setup</i> cuando existe el directorio de configuración <i>usermgmt</i>	16
		12.2	Ejecución del comando <i>roo++ usermgmt setup</i> cuando no existe el directorio de configuración <i>usermgmt</i>	17
4.1.1	Se podrá elegir entre el sistema de gestión de usuarios proporcionado por la plataforma software usada por la organización, y el proporcionado por las librerías de Spring Security, en combinación con un sistema CAS (Central Authentication Service)	12.1	Ejecución del comando <i>roo++ usermgmt setup</i> cuando existe el directorio de configuración <i>usermgmt</i>	16
		12.2	Ejecución del comando <i>roo++ usermgmt setup</i> cuando no existe el directorio de configuración <i>usermgmt</i>	17
5.1	Se debe permitir modificar la apariencia de la aplicación web (imágenes, hojas de estilos, ubicación de los elementos en la interfaz)	8	Ejecución del comando <i>roo++ layout setup</i>	19
5.2	Se debe permitir introducir la navegación mediante “migas de pan” en la aplicación web	6	Ejecución del comando <i>roo++ breadcrumb setup</i>	20
5.3	Se debe permitir establecer diferentes conjuntos de elementos en la aplicación web, que se mostrarán a cada usuario según su perfil y sus permisos	7	Ejecución del comando <i>roo++ context setup</i>	21

Conclusiones

La principal conclusión que se puede obtener de este plan de pruebas es que la realización de tests unitarios, y también de otras pequeñas pruebas no documentadas aquí, durante el desarrollo del software, aumenta en gran medida la calidad de éste, si bien requiere un esfuerzo mucho mayor, para buena parte de los proyectos. Sin embargo, por las características particulares de este proyecto, resultaba imprescindible la realización de dichas pruebas unitarias, siempre que fuera posible, con el fin de detectar el grueso de los fallos del mismo, ya que de hacerse de otra forma se habrían producido gran cantidad de fallos en cada prueba de sistema, haciendo mucho más difícil la detección de los defectos que daban lugar a ellos, especialmente para los defectos situados en clases de utilidades.

En líneas generales, el software cumple con todos sus objetivos, y posee una calidad elevada, ya que todos los fallos detectados han podido ser corregidos. El escaso número de fallos detectados en las pruebas de sistema se explica por la gran cantidad de pruebas unitarias llevadas a cabo, en las cuales se encontró la mayor parte del total de fallos detectados. Es de destacar que ninguno de los fallos detectados en las pruebas de sistema tenía relación directa con código cubierto por las pruebas unitarias, debido a las particularidades de este proyecto, en que la ejecución de cada comando se corresponde siempre con la ejecución de un método concreto. En cualquier caso, la ejecución de las pruebas de sistema resulta imprescindible, aun en el caso de que dichos métodos hubieran podido ser cubiertos por las pruebas unitarias, ya que el enfoque y el objetivo de cada uno de los tipos de pruebas es totalmente diferente (especialmente, los casos de prueba), y sería posible que se hubieran producido fallos no detectables por las pruebas unitarias.

Finalmente, es necesario indicar que durante el desarrollo del software se ha ido probando con éxito todos los casos de disponibilidad o no de cada uno de los comandos, pruebas no incluidas en este documento por su elevado número, si bien cada una de las pruebas de sistema llevadas a cabo comprueba que, al menos, no falla en ese momento la disponibilidad del comando ejecutado.

ANEXO I: clases de Test utilizadas en las pruebas unitarias

CreateModulesTest.java

```
package rooplusplus.roo.addon.utils;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStreamReader;

import org.apache.commons.io.FileUtils;
import org.apache.felix.scr.annotations.Reference;
import org.junit.After;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;
import org.springframework.roo.process.manager.FileManager;
import org.springframework.roo.project.MavenOperations;
import org.springframework.roo.project.ProjectOperations;
import org.springframework.roo.project.packaging.PackagingProviderRegistry;

import rooplusplus.roo.addon.utils.CreateModules;

/**
 * Tests de la clase CreateModules.
 *
 * @author javier.j.menendez
 */
public class CreateModulesTest {

    /**
     * Referencia al MavenOperations del container OSGi.
     */
    @Reference
    private MavenOperations mavenOperations;

    /**
     * Get a reference to the ProjectOperations from the underlying OSGi
     * container. Make sure you are referencing the Roo bundle which contains
     * this service in your add-on pom.xml.
     */
    @Reference
    private ProjectOperations projectOperations;

    /**
     * Referencia al PackagingProviderRegistry del container OSGi.
     */
}
```

```

@Reference
private PackagingProviderRegistry packagingProviderRegistry;

/**
 * Get a reference to the FileManager from the underlying OSGi container.
 * Make sure you are referencing the Roo bundle which contains this service
 * in your add-on pom.xml. Using the Roo file manager instead of
 * java.io.File gives you automatic rollback in case an Exception is thrown.
 */
@Reference
private FileManager fileManager;

/**
 * Código que se ejecutara antes de los tests.
 */
@Before
public void before() {

    new File("target/tests/proyectoCopyResources").mkdir();
    new File("target/tests/proyectoCopyResources/src").mkdir();

}

/**
 * Crea en el directorio "target/tests/" un archivo, para ejecutar las
 * pruebas unitarias.
 *
 * @param nombre Nombre del archivo
 * @param contenido Texto del contenido inicial del archivo
 */
private void creaArchivoDeTest(String nombre, String contenido) {

    File f = new File("target/tests/" + nombre);
    FileWriter fw = null;
    try {
        f.createNewFile();
        fw = new FileWriter(f);
        fw.write(contenido);

    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        if (fw != null) {
            try {
                fw.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}
}

```

```

/**
 * Codigo que se ejecutara tras los tests.
 */
@After
public void after() {

    File dir = new File("target/tests");
    try {
        FileUtils.deleteDirectory(dir);
    } catch (IOException e) {
        e.printStackTrace();
    }
    dir.mkdir();
}

/**
 * Tests del metodo copyResources.
 */
@Test
public void testCopyResources() {

    CreateModules.copyResources("proyectoCopyResources",
        "target/tests/proyectoCopyResources", "rooAddon/", this.getClass()
            .getClassLoader());
    if (new File("target/tests/proyectoCopyResources/src").exists()) {
        Assert.fail("Ha fallado");
    }
    String fileName, comienzo = "target/tests/proyectoCopyResources/";
    BufferedReader in = null;
    try {
        in =
            new BufferedReader(new InputStreamReader(this.getClass()
                .getClassLoader()
                    .getResourceAsStream("rooAddon/" + "textfiles.txt"),
                    "UTF-8"));
        while (in.ready()) {
            fileName = in.readLine();
            if (fileName.equals("pom.xml")) {
                fileName = comienzo + fileName;
            } else {
                fileName = comienzo + "proyectoCopyResources-" + fileName;
            }
            if (!new File(fileName).exists()) {
                Assert.fail("Ha fallado");
            }
        }
    } catch (IOException ex) {
        Assert.fail("Ha fallado");
    } finally {
        if (in != null) {
            try {
                in.close();
            }

```



```
        } catch (IOException e) {
            Assert.fail("Ha fallado");
        }
    }
}
try {
    in =
        new BufferedReader(new InputStreamReader(this.getClass()
            .getClassLoader()
            .getResourceAsStream("rooAddon/" + "dirs.txt"), "UTF-8"));
    while (in.ready()) {
        fileName = in.readLine();
        fileName = comienzo + "proyectoCopyResources-" + fileName;
        if (!new File(fileName).exists()) {
            Assert.fail("Ha fallado");
        }
    }
} catch (IOException ex) {
    Assert.fail("Ha fallado");
} finally {
    if (in != null) {
        try {
            in.close();
        } catch (IOException e) {
            Assert.fail("Ha fallado");
        }
    }
}
}
}
```

FileUtilitiesTest.java

```
package rooplusplus.roo.addon.utils;

import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

import org.apache.commons.io.FileUtils;
import org.junit.After;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;

import rooplusplus.roo.addon.utils.FileUtilities;

/**
 * Tests de la clase FileUtilities.
 *
 * @author javier.j.menendez
 */
```

```

public class FileUtilitiesTest {

    /**
     * Código que se ejecutara antes de los tests.
     */
    @Before
    public void before() {

        creaArchivoDeTest("este_fichero_existe.txt", "Este fichero\nexiste.");
        creaArchivoDeTest("reemplazo_de_texto.txt",
            "Este archivo contiene un texto a ser reemplazado por otro texto.");
        creaArchivoDeTest("archivo_lineas_vacias.txt",
            "Este fichero\ntiene\n \n\n\nalguna \n\t \t\n que otra linea\nvacia\n");
        new File("target/tests/directorio_a_copiar").mkdir();
        new File("target/tests/directorio_a_copiar/subdirectorio").mkdir();
        new File("target/tests/directorio_copiado_que_ya_existia").mkdir();
        creaArchivoDeTest("directorio_a_copiar/archivo1.txt",
            "Contenido del archivo 1.\n");
        creaArchivoDeTest("directorio_a_copiar/subdirectorio/archivo2.txt",
            "Contenido del archivo 2.\n");
    }

    /**
     * Crea en el directorio "target/tests/" un archivo, para ejecutar las
     * pruebas unitarias.
     *
     * @param nombre Nombre del archivo
     * @param contenido Texto del contenido inicial del archivo
     */
    private void creaArchivoDeTest(String nombre, String contenido) {

        File f = new File("target/tests/" + nombre);
        FileWriter fw = null;
        try {
            f.createNewFile();
            fw = new FileWriter(f);
            fw.write(contenido);

        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            if (fw != null) {
                try {
                    fw.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
        }
    }

    /**
     * Código que se ejecutara tras los tests.

```

```

*/
@After
public void after() {

    File dir = new File("target/tests");
    try {
        FileUtils.deleteDirectory(dir);
    } catch (IOException e) {
        e.printStackTrace();
    }
    dir.mkdir();
}

/**
 * Tests del metodo readFileAsStringThrows. El fichero
 * "target/tests/este_fichero_no_existe.txt" no existe, y el fichero
 * "target/tests/este_fichero_existe.txt" contiene el texto
 * "Este fichero\nexiste.".
 */
@Test
public void testReadFileAsStringThrows() {

    try {
        FileUtilities
            .readFileAsStringThrows("target/tests/este_fichero_no_existe.txt");
        Assert.fail("Ha fallado");
    } catch (IOException e) {

    }

    try {
        String str =
            FileUtilities
                .readFileAsStringThrows("target/tests/este_fichero_existe.txt");
        Assert.assertEquals("Este fichero\nexiste.", str);
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    }
}

/**
 * Tests del metodo readFileAsString. El fichero
 * "target/tests/este_fichero_no_existe.txt" no existe, y el fichero
 * "target/tests/este_fichero_existe.txt" contiene el texto
 * "Este fichero\nexiste.".
 */
@Test
public void testReadFileAsString() {

    String str =
        FileUtilities
            .readFileAsString("target/tests/este_fichero_no_existe.txt");
    Assert.assertEquals("", str);
    str =

```

```

        FileUtilities
            .readFileAsString("target/tests/este_fichero_existe.txt");
        Assert.assertEquals("Este fichero\nexiste.", str);
    }

/**
 * Tests del metodo replaceText. El fichero
 * "target/tests/reemplazo_de_texto.txt" contiene el texto
 * "Este archivo contiene un texto a ser reemplazado por otro texto.".
 */
@Test
public void testReplaceText() {

    FileUtilities.replaceText("texto a ser reemplazado",
        "texto tras reemplazarlo", "target/tests/reemplazo_de_texto.txt");
    String str =
        FileUtilities
            .readFileAsString("target/tests/reemplazo_de_texto.txt");
    Assert.assertEquals(
        "Este archivo contiene un texto tras reemplazarlo por otro texto.",
        str);
    FileUtilities.replaceText("texto a ser reemplazado",
        "texto tras reemplazarlo", "target/tests/reemplazo_de_texto.txt");
    str =
        FileUtilities
            .readFileAsString("target/tests/reemplazo_de_texto.txt");
    Assert.assertEquals(
        "Este archivo contiene un texto tras reemplazarlo por otro texto.",
        str);
}

/**
 * Tests del metodo getConfigurationFilePath.
 */
@Test
public void testGetConfigurationFilePath() {

    try {

        System.setProperty("ROOADDON", new File(
            "src/main/resources/RooAddonsData").getAbsolutePath() + "/");
        String str =
            FileUtilities.getConfigurationFilePath("doc/fonts.xml");
        str = new File(str).getCanonicalPath();
        String expected =
            new File("src/main/resources/RooAddonsData/doc/fonts.xml")
                .getCanonicalPath();
        Assert.assertEquals(expected, str);
    } catch (IOException ex) {
        Assert.fail("Ha fallado");
    }
}
}

```

```

/**
 * Tests del metodo removeBlankLines. El archivo
 * "target/tests/archivo_lineas_vacias.txt" contiene algunas lineas vacias.
 */
@Test
public void testRemoveBlankLines() {

    FileUtilities
        .removeBlankLines("target/tests/archivo_lineas_vacias.txt");
    String str =
        FileUtilities
            .readFileAsString("target/tests/archivo_lineas_vacias.txt");
    Assert.assertEquals(
        "Este fichero\n\tiene\n\talguna \n que otra linea\n\tvacia", str);
}

/**
 * Tests del metodo copyDirContent. El archivo
 * "target/tests/directorio_a_copiar/archivo1.txt" contiene el texto
 * "Contenido del archivo 1.\n", y
 * "target/tests/directorio_a_copiar/subdirectorio/archivo2.txt" el texto
 * "Contenido del archivo 2.\n".
 */
@Test
public void testCopyDirContent() {

    FileUtilities.copyDirContent("target/tests/directorio_a_copiar",
        "target/tests/directorio_copiado_que_no_existia");
    String arch1 =
        FileUtilities
            .readFileAsString("target/tests/directorio_copiado_que_no_existia/archivo1.txt");
    String arch2 =
        FileUtilities
            .readFileAsString("target/tests/directorio_copiado_que_no_existia/subdirectorio/archivo2.txt");
    Assert.assertEquals("Contenido del archivo 1.\n", arch1);
    Assert.assertEquals("Contenido del archivo 2.\n", arch2);
    FileUtilities.copyDirContent("target/tests/directorio_a_copiar",
        "target/tests/directorio_copiado_que_ya_existia");
    arch1 =
        FileUtilities
            .readFileAsString("target/tests/directorio_copiado_que_ya_existia/archivo1.txt");
    arch2 =
        FileUtilities
            .readFileAsString("target/tests/directorio_copiado_que_ya_existia/subdirectorio/archivo2.txt");
    Assert.assertEquals("Contenido del archivo 1.\n", arch1);
    Assert.assertEquals("Contenido del archivo 2.\n", arch2);
}

/**
 * Tests del metodo readTextFile.
 */
@Test

```

```

public void testReadTextFile() {

    String str =
        FileUtilities.readTextFile("textfiles.txt", "rooAddon/", this
            .getClass().getClassLoader());
    String expected =
        "pom.xml\r\ncore/pom.xml\r\nndocs/pom.xml\r\nndocs/src/InstallationManual.xml\r\nndocs/src/UserManual.x
ml\r\nndocs/src/assembly/package.xml\r\nnear/pom.xml\r\nnear/src/main/application/META-INF/weblogic-
application.xml\r\npackage/pom.xml\r\npackage/src/assembly/package.xml\r\nscripts/pom.xml\r\nscripts
/src/assembly/compress-zip.xml";
    Assert.assertEquals(expected, str);
}

/**
 * Tests del metodo getFirstLineConfFile.
 */
@Test
public void testGetFirstLineConfFile() {

    try {
        String str = FileUtilities.getFirstLineConfFile("licenses.xml");
        Assert.assertEquals("<licenses>", str);
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    }
    try {
        FileUtilities.getFirstLineConfFile("esto_no_existe.xml");
        Assert.fail("Ha fallado");
    } catch (IOException e) {

    }
}
}

```

PomUtilsTest.java

NOTA: en algunos tests de esta clase se introduce la línea "<?xml version='1.0' encoding='UTF-8' standalone='no'?" al comienzo de el archivo a comparar con el resultado; esto es debido a que esta línea es introducida por las librerías utilizadas para guardar archivos XML, sin que ello signifique que el resultado no es el esperado.

```

package rooplusplus.roo.addon.utils;

import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.TransformerException;
import javax.xml.xpath.XPathExpressionException;

import org.apache.commons.io.FileUtils;
import org.apache.felix.scr.annotations.Reference;
import org.junit.After;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;

```

```

import org.springframework.roo.process.manager.FileManager;
import org.springframework.roo.project.ProjectOperations;
import org.w3c.dom.Document;
import org.w3c.dom.Node;
import org.xml.sax.SAXException;

import rooplusplus.roo.addon.utils.FileUtilities;
import rooplusplus.roo.addon.utils.PomUtils;
import rooplusplus.roo.addon.utils.XmlUtils;

/**
 * Tests de la clase PomUtils.
 *
 * @author javier.j.menendez
 */
public class PomUtilsTest {

    /**
     * Get a reference to the FileManager from the underlying OSGi container.
     * Make sure you are referencing the Roo bundle which contains this service
     * in your add-on pom.xml. Using the Roo file manager instead if
     * java.io.File gives you automatic rollback in case an Exception is thrown.
     */
    @Reference
    private FileManager fileManager;

    /**
     * Get a reference to the ProjectOperations from the underlying OSGi
     * container. Make sure you are referencing the Roo bundle which contains
     * this service in your add-on pom.xml.
     */
    @Reference
    private ProjectOperations projectOperations;

    /**
     *Codigo que se ejecutara antes de los tests.
     */
    @Before
    public void before() {

        String pom =
            FileUtilities.readTextFile("docs/pom.xml", "rooAddon/", this
                .getClass().getClassLoader());
        creaArchivoDeTest("pom.xml", pom);
        pom =
            FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this
                .getClass().getClassLoader());
        creaArchivoDeTest("scope-pom.xml", pom);
        creaArchivoDeTest("properties-pom.xml", pom);
        creaArchivoDeTest("version-pom.xml", pom);
        creaArchivoDeTest("proyecto-core/pom.xml", pom);
        FileUtilities.replaceText("${project.version}", "0.1.0.BUILD-SNAPSHOT",

```

```

        "target/tests/version-pom.xml");
    creaArchivoDeTest("license-plugin-pom.xml", pom);
    pom =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    creaArchivoDeTest("license-pom.xml", pom);
    creaArchivoDeTest("build-plugins-pom.xml", pom);
    XmlUtils.eliminaNodoSiExiste("target/tests/build-plugins-pom.xml",
        "/project/build");
}

/**
 * Crea en el directorio "target/tests/" un archivo, para ejecutar las
 * pruebas unitarias.
 *
 * @param nombre Nombre del archivo
 * @param contenido Texto del contenido inicial del archivo
 */
private void creaArchivoDeTest(String nombre, String contenido) {

    File f = new File("target/tests/" + nombre);
    FileWriter fw = null;
    try {
        f.createNewFile();
        fw = new FileWriter(f);
        fw.write(contenido);

    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        if (fw != null) {
            try {
                fw.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}

/**
 * Codigo que se ejecutara tras los tests.
 */
@After
public void after() {

    File dir = new File("target/tests");
    try {
        FileUtils.deleteDirectory(dir);
    } catch (IOException e) {
        e.printStackTrace();
    }
    dir.mkdir();
}

```



```

}

/**
 * Tests del metodo replaceVariables.
 */
@Test
public void testReplaceVariables() {

    PomUtils.replaceVariables("target/tests/pom.xml", "", "proyecto",
        "1.0", "top.level.package");
    String pom = FileUtilities.readFileAsString("target/tests/pom.xml");
    String expected =
        FileUtilities.readTextFile("docs/pom.xml", "rooAddon/", this
            .getClass().getClassLoader());
    expected =
        expected.replace("${project.parent.groupId}", "top.level.package");
    expected = expected.replace("${project.parent.artifactId}", "proyecto");
    expected = expected.replace("${project.parent.version}", "1.0");
    expected = expected.replace("${project.parent.name}", "proyecto");
    expected = expected.replace("${project.version}", "1.0");
    Assert.assertEquals(expected, pom);
}

/**
 * Tests del metodo addDependencies.
 */
@Test
public void testAddDependencies() {

    String expected =
        FileUtilities.readFileAsString("target/tests/pom.xml");
    String dependency =

"<dependencies>\n<dependency>\n<groupId>groupId</groupId>\n<artifactId>artifactId</artifactId>\n<ver
sion>version</version>\n</dependency>\n</dependencies>\n";
    expected = expected.replace("</project>", dependency + "</project>");
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    try {
        PomUtils.addDependency("target/tests/pom.xml", "groupId",
            "artifactId", "version");
        String str = FileUtilities.readFileAsString("target/tests/pom.xml");
        expected = expected.replace("\r\n", "\n");
        str = str.replace("\r\n", "\n");
        Assert.assertEquals(expected, str);
    } catch (XPathExpressionException e) {
        Assert.fail("Ha fallado");
    } catch (ParserConfigurationException e) {
        Assert.fail("Ha fallado");
    } catch (SAXException e) {
        Assert.fail("Ha fallado");
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    }
}

```

```

    } catch (TransformerException e) {
        Assert.fail("Ha fallado");
    }
}

/**
 * Tests del metodo addDependencyToBuildPlugin.
 */
@Test
public void testAddDependencyToBuildPlugin() {

    String expected =
        FileUtilities.readFileAsString("target/tests/pom.xml");
    String dependency =

"<dependency>\n<groupId>groupId</groupId>\n<artifactId>artifactId</artifactId>\n<version>version</ve
rsion>\n</dependency>";
    expected = expected.replace("\r\n", "\n");
    expected =
        expected
            .replace(
                "<version>1.1.2_01</version>\n\t\t\t\t\t</dependency>\n\t\t\t\t\t</dependencies>",
                "<version>1.1.2_01</version>\n\t\t\t\t\t</dependency>\n\t\t\t\t\t"
                    + dependency + "\n</dependencies>");
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + expected;
    try {
        PomUtils.addDependencyToBuildPlugin("target/tests/pom.xml",
            "docbkk-maven-plugin", "groupId", "artifactId", "version");
        String str = FileUtilities.readFileAsString("target/tests/pom.xml");
        str = str.replace("\r\n", "\n");
        Assert.assertEquals(expected, str);
    } catch (XPathExpressionException e) {
        Assert.fail("Ha fallado");
    } catch (ParserConfigurationException e) {
        Assert.fail("Ha fallado");
    } catch (SAXException e) {
        Assert.fail("Ha fallado");
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (TransformerException e) {
        Assert.fail("Ha fallado");
    }
}

/**
 * Tests del metodo addScope.
 */
@Test
public void testAddScope() {

    String expected =
        FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this

```

```

        .getClass().getClassLoader());
expected =
    "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
    + expected;
expected =
    expected.replace("</plugin>", "<scope>test</scope>\n</plugin>");
expected = expected.replace("\r\n", "\n");
try {
    Document doc =
        XmlUtils.createDocumentFromFile("target/tests/scope-pom.xml");
    XmlUtils.guardaXml(doc, "target/tests/scope-pom.xml");
    PomUtils.addScope("test",
        "/project/build/plugins/plugin/artifactId", doc,
        "target/tests/scope-pom.xml");
    String actual =
        FileUtilities.readFileAsString("target/tests/scope-pom.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
} catch (SAXException e) {
    Assert.fail("Ha fallado");
} catch (IOException e) {
    Assert.fail("Ha fallado");
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (XPathExpressionException e) {
    Assert.fail("Ha fallado");
} catch (TransformerException e) {
    Assert.fail("Ha fallado");
}
}

/**
 * Tests del metodo addProperty, tanto cuando existe el nodo "properties"
 * como cuando no existe. Tambien cuando no existe el pom.xml indicado
 */
@Test
public void testAddProperty() {

    // Cuando no existen properties
    String expected =
        FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this
            .getClass().getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    expected =
        expected
            .replace("</build>",
                "</build>\n<properties>\n<propiedad1>valor1</propiedad1>\n</properties>");
    expected = expected.replace("\r\n", "\n");
    try {
        PomUtils.addProperty("target/tests/properties-pom.xml",
            "propiedad1", "valor1");
    }
}

```

```

String actual =
    FileUtilities
        .readFileAsString("target/tests/properties-pom.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
} catch (IOException e) {
    Assert.fail("Ha fallado");
} catch (XPathExpressionException e) {
    Assert.fail("Ha fallado");
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
}
// Cuando ya existen properties
try {
    PomUtils.addProperty("target/tests/properties-pom.xml",
        "propiedad2", "valor2");
String actual =
    FileUtilities
        .readFileAsString("target/tests/properties-pom.xml");
expected =
    expected.replace("</properties>",
        "<propiedad2>valor2</propiedad2>\n</properties>");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
} catch (IOException e) {
    Assert.fail("Ha fallado");
} catch (XPathExpressionException e) {
    Assert.fail("Ha fallado");
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
}
// Cuando no existe el pom
try {
    PomUtils.addProperty("target/tests/pom-que-no-existe.xml",
        "propiedad2", "valor2");
    Assert.fail("Ha fallado");
} catch (IOException ex) {

} catch (XPathExpressionException e) {
    Assert.fail("Ha fallado");
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
}
}

/**
 * Tests del metodo setVersion.

```

```

*/
@Test
public void testSetVersion() {

    String expected =
        FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this
            .getClass().getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    expected = expected.replace("\r\n", "\n");
    expected =
        expected.replace("<version>${project.version}</version>",
            "<version>3.0</version>");
    PomUtils.setVersion("target/tests/version-pom.xml", "3.0", fileManager);
    String actual =
        FileUtilities.readFileAsString("target/tests/version-pom.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo addLicensePlugin.
 */
@Test
public void testAddLicensePlugin() {

    String expected =
        FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this
            .getClass().getClassLoader());
    String licPl =
        FileUtilities.readTextFile("plugin-license.xml", "RooAddonsData/",
            this.getClass().getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    expected = expected.replace("</plugins>", licPl + "\n</plugins>");
    expected = expected.replace("\r\n", "\n");
    PomUtils.addLicensePlugin("target/tests/license-plugin-pom.xml");
    String actual =
        FileUtilities
            .readFileAsString("target/tests/license-plugin-pom.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo addLicense, tanto el caso de que ya se haya introducido
 * una licencia como el que no.
 */
@Test
public void testAddLicense() {

```

```

// Cuando no se habia introducido licencia
String lic =
    FileUtilities.readTextFile("licenses.xml", "RooAddonsData/", this
        .getClass().getClassLoader());
String expected =
    FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
        .getClassLoader());
expected = expected.replace("</project>", lic + "\n</project>");
expected = expected.replace("\r\n", "\n");
expected =
    "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
PomUtils.addLicense("target/tests/license-pom.xml");
String actual =
    FileUtilities.readFileAsString("target/tests/license-pom.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
PomUtils.addLicense("target/tests/license-pom.xml");
// Cuando ya se habia introducido licencia
actual = FileUtilities.readFileAsString("target/tests/license-pom.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo addLicense, en los 3 casos de que no exista el nodo
 * build, de que exista pero no tenga plugins, y de que si los tenga.
 */
@Test
public void testAddBuildPlugin() {

    // Cuando no existe el nodo build
    String expected =
        FileUtilities
            .readFileAsString("target/tests/build-plugins-pom.xml");
    expected = expected.replace("\r\n", "\n");
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + expected;
    expected =
        expected
            .replace(
                "</project>",
                "<build>\n<plugins>\n<plugin>\n<groupId>groupId</groupId>\n<artifactId>artifactId</artifactId>\n<v
ersion>version</version>\n</plugin>\n</plugins>\n</build>\n</project>");
    try {
        Document doc =
            XmlUtils
                .createDocumentFromFile("target/tests/build-plugins-pom.xml");
        Node plugin = doc.createElement("plugin");
        Node gId = doc.createElement("groupId");
        gId.setTextContent("groupId");
        Node aId = doc.createElement("artifactId");

```

```

aId.setTextContent("artifactId1");
Node version = doc.createElement("version");
version.setTextContent("version1");
plugin.appendChild(gId);
plugin.appendChild(aId);
plugin.appendChild(version);
PomUtils.addBuildPlugin("target/tests/build-plugins-pom.xml",
    plugin);
String actual =
    FileUtilities
        .readFileAsString("target/tests/build-plugins-pom.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
} catch (IOException e) {
    Assert.fail("Ha fallado");
}
}
// Cuando existe el nodo build pero no contiene plugins
XmlUtils.eliminaNodoSiExiste("target/tests/build-plugins-pom.xml",
    "/project/build/plugins");
try {
    Document doc =
        XmlUtils
            .createDocumentFromFile("target/tests/build-plugins-pom.xml");
    Node plugin = doc.createElement("plugin");
    Node gId = doc.createElement("groupId");
    gId.setTextContent("groupId1");
    Node aId = doc.createElement("artifactId");
    aId.setTextContent("artifactId1");
    Node version = doc.createElement("version");
    version.setTextContent("version1");
    plugin.appendChild(gId);
    plugin.appendChild(aId);
    plugin.appendChild(version);
    PomUtils.addBuildPlugin("target/tests/build-plugins-pom.xml",
        plugin);
    String actual =
        FileUtilities
            .readFileAsString("target/tests/build-plugins-pom.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
} catch (IOException e) {
    Assert.fail("Ha fallado");
}
}
// Cuando existe el nodo build y tiene plugins
expected =

```

```

        expected
            .replace(
                "</plugins>",
                "<plugin>\n<groupId>groupId2</groupId>\n<artifactId>artifactId2</artifactId>\n<version>version2</version>\n</plugin>\n</plugins>");
    try {
        Document doc =
            XmlUtils
                .createDocumentFromFile("target/tests/build-plugins-pom.xml");
        Node plugin = doc.createElement("plugin");
        Node gId = doc.createElement("groupId");
        gId.setTextContent("groupId2");
        Node aId = doc.createElement("artifactId");
        aId.setTextContent("artifactId2");
        Node version = doc.createElement("version");
        version.setTextContent("version2");
        plugin.appendChild(gId);
        plugin.appendChild(aId);
        plugin.appendChild(version);
        PomUtils.addBuildPlugin("target/tests/build-plugins-pom.xml",
            plugin);
        String actual =
            FileUtilities
                .readFileAsString("target/tests/build-plugins-pom.xml");
        actual = actual.replace("\r\n", "\n");
        Assert.assertEquals(expected, actual);
    } catch (ParserConfigurationException e) {
        Assert.fail("Ha fallado");
    } catch (SAXException e) {
        Assert.fail("Ha fallado");
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    }
}
}

```

ProjectUtilsTest.java

```

package rooplusplus.roo.addon.utils;

import java.io.BufferedInputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileWriter;
import java.io.IOException;

import org.apache.commons.io.FileUtils;
import org.junit.After;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;

import rooplusplus.roo.addon.utils.FileUtilities;
import rooplusplus.roo.addon.utils.ProjectUtils;

```

Pruebas – Spring Roo Add-Ons para Prototipado Rápido


```

/**
 * Tests de la clase PomUtils.
 *
 * @author javier.j.menendez
 */
public class ProjectUtilsTest {

    /**
     * How big buffer to use to process files.
     */
    public static final int BUFFER_SIZE = 65536;

    /**
     * Codigo que se ejecutara antes de los tests.
     */
    @Before
    public void before() {

        new File("target/tests/proyecto").mkdir();
        new File("target/tests/proyecto/proyecto-core").mkdir();
        new File("target/tests/proyecto/proyecto-otromodulo").mkdir();
        String texto =
            FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this
                .getClass().getClassLoader());
        creaArchivoDeTest("archivoACopiar.txt", texto);
        new File("target/tests/proyecto/proyecto-docs").mkdir();
        new File("target/tests/proyecto/proyecto-doc").mkdir();
        new File("target/tests/proyecto/proyecto-doc/front_pages").mkdir();
        creaArchivoDeTest("proyecto/proyecto-core/archivodetexto.txt",
            "Este archivo contiene texto.");
    }

    /**
     * Crea en el directorio "target/tests/" un archivo, para ejecutar las
     * pruebas unitarias.
     *
     * @param nombre Nombre del archivo
     * @param contenido Texto del contenido inicial del archivo
     */
    private void creaArchivoDeTest(String nombre, String contenido) {

        File f = new File("target/tests/" + nombre);
        FileWriter fw = null;
        try {
            f.createNewFile();
            fw = new FileWriter(f);
            fw.write(contenido);

        } catch (IOException e) {
            e.printStackTrace();
        } finally {

```

```

        if (fw != null) {
            try {
                fw.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}

/**
 * Codigo que se ejecutara tras los tests.
 */
@After
public void after() {

    File dir = new File("target/tests");
    try {
        FileUtils.deleteDirectory(dir);
    } catch (IOException e) {
        e.printStackTrace();
    }
    dir.mkdir();
}

/**
 * Tests del metodo copyTextFile.
 */
@Test
public void testCopyTextFile() {

    ProjectUtils.copyTextFile("target/tests/proyecto", "proyecto",
        "core/pom.xml", "rooAddon/", this.getClass().getClassLoader());
    String expected =
        FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this
            .getClass().getClassLoader());
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-core/pom.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);

}

/**
 * Tests del metodo copyExternTextFile.
 */
@Test
public void testCopyExternTextFile() {

    String expected =
        FileUtilities.readTextFile("core/pom.xml", "rooAddon/", this

```

```

        .getClass().getClassLoader());
expected = expected.replace("\r\n", "\n");
ProjectUtils.copyExternTextFile("target/tests/proyecto", "proyecto",
    "otromodulo/pom.xml", "target/tests/archivoACopiar.txt");
String actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-otromodulo/pom.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo appendTextToFile.
 */
@Test
public void testAppendTextToFile() {

    ProjectUtils.appendTextToFile("Texto introducido en el archivo",
        "target/tests/proyecto", "proyecto", "core/archivodetexto.txt");
String actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-core/archivodetexto.txt");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(
    "Este archivo contiene texto.\nTexto introducido en el archivo",
    actual);
}
}

```

WebModuleUtilsTest.java

```

package rooplusplus.roo.addon.utils;

import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

import org.apache.commons.io.FileUtils;
import org.junit.After;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;

import rooplusplus.roo.addon.utils.FileUtilities;
import rooplusplus.roo.addon.utils.WebModuleUtils;

/**
 * Tests de la clase WebModuleUtils.
 *
 * @author javier.j.menendez
 */
public class WebModuleUtilsTest {

```

```

/**
 * Codigo que se ejecutara antes de los tests.
 */
@Before
public void before() {

    new File("target/tests/proyecto").mkdir();
    new File(
        "target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring")
        .mkdirs();
    creaArchivoDeTest(
        "proyecto/proyecto-web/pom.xml",
        "<project><dependencies><dependency><groupId>mysql</groupId></dependency></dependencies></project>")
;
    new File(
        "target/tests/proyecto/proyecto-web/src/main/webapp/WEB-INF/views")
        .mkdirs();
    creaArchivoDeTest(
        "proyecto/proyecto-web/src/main/webapp/WEB-INF/views/header.jsp",
        "Archivo a ser sustituido");
    creaArchivoDeTest(
        "proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext.xml",
        "<beans>\n<bean
class=\"org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean\"> </bean>\n<bean id
= \"dataSource\"> </bean>\n</beans>");
    new File(
        "target/tests/proyecto2/proyecto2-web/src/main/resources/META-INF/spring")
        .mkdirs();
    new File(
        "target/tests/proyecto2/proyecto2-web/src/main/webapp/WEB-INF/spring")
        .mkdirs();
    creaArchivoDeTest(
        "proyecto2/proyecto2-web/src/main/resources/META-INF/spring/applicationContext.xml",
        "<beans>\n<!--This will automatically locate any and all property files you have-->\n<context:property-placeholder/>\n</beans>");
    creaArchivoDeTest(
        "proyecto2/proyecto2-web/src/main/webapp/WEB-INF/spring/webmvc-config.xml",
        "<beans/>");
    creaArchivoDeTest("proyecto2/proyecto2-web/pom.xml",
        "<project>\n<build>\n<plugins>\n</plugins>\n</build>\n</project>");
    new File(
        "target/tests/renamepersistence/renamepersistence-web/src/main/resources/META-INF")
        .mkdirs();
    creaArchivoDeTest(
        "renamepersistence/renamepersistence-web/src/main/resources/META-INF/persistence.xml",
        "persistence-unit name=\"persistenceUnit\"");
}

/**
 * Crea en el directorio "target/tests/" un archivo, para ejecutar las
 * pruebas unitarias.
 *
 * @param nombre Nombre del archivo
 * @param contenido Texto del contenido inicial del archivo

```

```

*/
private void creaArchivoDeTest(String nombre, String contenido) {

    File f = new File("target/tests/" + nombre);
    FileWriter fw = null;
    try {
        f.createNewFile();
        fw = new FileWriter(f);
        fw.write(contenido);

    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        if (fw != null) {
            try {
                fw.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}

/**
 * Codigo que se ejecutara tras los tests.
 */
@After
public void after() {

    File dir = new File("target/tests");
    try {
        FileUtils.deleteDirectory(dir);
    } catch (IOException e) {
        e.printStackTrace();
    }
    dir.mkdir();
}

/**
 * Tests del metodo copyApplicationPropertiesToWebModule.
 */
@Test
public void testCopyApplicationPropertiesToWebModule() {

    WebModuleUtils.copyApplicationPropertiesToWebModule(
        "target/tests/proyecto", "rooAddon/", "proyecto", this.getClass()
            .getClassLoader());
    String expected =
        FileUtilities
            .readTextFile(
                "web/src/main/resources/META-INF/spring/application.properties",
                "rooAddon/", this.getClass().getClassLoader());
    expected = expected.replace("test", "proyecto");
}

```

```

        expected = expected.replace("\r\n", "\n");
        String actual =
            FileUtilities
                .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/application.properties");
        actual = actual.replace("\r\n", "\n");
        Assert.assertEquals(expected, actual);
    }

/**
 * Tests del metodo copyDatabasePropertiesToWebModule, con cada uno de los
 * sistemas de gestion de bases de datos soportados.
 */
@Test
public void testCopyDatabasePropertiesToWebModule() {

    // MySQL
    WebModuleUtils.copyDatabasePropertiesToWebModule(
        "target/tests/proyecto", "rooAddon/", "proyecto", "database", this
            .getClass().getClassLoader());
    String expected =
        FileUtilities.readTextFile(
            "web/src/main/resources/META-INF/spring/database.properties",
            "rooAddon/", this.getClass().getClassLoader());
    expected = expected.replace("TEST", "database");
    expected = expected.replace("testDS", "proyectoDS");
    expected =
        expected.replace("HIBERNATE_DIALECT_GOES_HERE",
            "org.hibernate.dialect.MySQL5InnoDBDialect");
    expected =
        expected.replace("persistenceUnitTest", "persistenceUnitProyecto");
    expected = expected.replace("\r\n", "\n");
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);

    // SQL Server
    FileUtilities.replaceText("mysql", "net.sourceforge.jtds",
        "target/tests/proyecto/proyecto-web/pom.xml");
    new File(
        "target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties")
        .delete();
    expected =
        expected.replace("MySQL5InnoDBDialect", "SQLServer2008Dialect");
    WebModuleUtils.copyDatabasePropertiesToWebModule(
        "target/tests/proyecto", "rooAddon/", "proyecto", "database", this
            .getClass().getClassLoader());
    actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties");
    actual = actual.replace("\r\n", "\n");

```

```

Assert.assertEquals(expected, actual);
// Oracle
FileUtilities.replaceText("net.sourceforge.jtds", "ojdbc",
    "target/tests/proyecto/proyecto-web/pom.xml");
new File(
    "target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties")
    .delete();
expected = expected.replace("SQLServer2008Dialect", "Oracle10gDialect");
WebModuleUtils.copyDatabasePropertiesToWebModule(
    "target/tests/proyecto", "rooAddon/", "proyecto", "database", this
        .getClass().getClassLoader());
actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
// PostgreSQL
FileUtilities.replaceText("ojdbc", "postgresql",
    "target/tests/proyecto/proyecto-web/pom.xml");
new File(
    "target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties")
    .delete();
expected = expected.replace("Oracle10gDialect", "PostgreSQLDialect");
WebModuleUtils.copyDatabasePropertiesToWebModule(
    "target/tests/proyecto", "rooAddon/", "proyecto", "database", this
        .getClass().getClassLoader());
actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/database.properties");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo copyApplicationContextPlatformToWebModule.
 */
@Test
public void testCopyApplicationContextPlatformToWebModule() {

    String expected =
        FileUtilities.readTextFile("applicationContext-platform.xml",
            "RooAddonsData/", this.getClass().getClassLoader());
    WebModuleUtils.copyApplicationContextPlatformToWebModule(
        "target/tests/proyecto", "proyecto");
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/applicationContext-platform.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

```

```

}

/**
 * Tests del metodo copyApplicationContextSecurityToWebModule.
 */
@Test
public void testCopyApplicationContextSecurityToWebModule() {

    String expected =
        FileUtilities.readTextFile(
            "usermgmt/applicationContext-security.xml", "RooAddonsData/",
            this.getClass().getClassLoader());
    WebModuleUtils.copyApplicationContextSecurityToWebModule(
        "target/tests/proyecto", "proyecto");
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/applicationContext-security.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo copyApplicationContextPortalToWebModule.
 */
@Test
public void testCopyApplicationContextPortalToWebModule() {

    String expected =
        FileUtilities.readTextFile("applicationContext-portal.xml",
            "RooAddonsData/", this.getClass().getClassLoader());
    WebModuleUtils.copyApplicationContextPortalToWebModule(
        "target/tests/proyecto", "proyecto");
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-
INF/spring/applicationContext-portal.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo replaceHeader.
 */
@Test
public void testReplaceHeader() {

    String expected =
        FileUtilities.readTextFile("portal-header.jsp", "RooAddonsData/",
            this.getClass().getClassLoader());
    WebModuleUtils.replaceHeader("target/tests/proyecto", "proyecto");
    String actual =

```



```

        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/webapp/META-INF/views/header.jsp");
        expected = expected.replace("\r\n", "\n");
        actual = actual.replace("\r\n", "\n");
        Assert.assertEquals(expected, actual);
    }

/**
 * Tests del metodo updateAppContextDatabase.
 */
@Test
public void testUpdateAppContextDatabase() {

    String expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n<beans
xmlns:jee=\"http://www.springframework.org/schema/jee\">\n"
        + FileUtilities
            .readTextFile(
                "web/src/main/resources/META-INF/spring/jndi-datasource.xml",
                "rooAddon/", this.getClass().getClassLoader());
    expected =
        expected
        + "\n"
        + FileUtilities
            .readTextFile(
                "web/src/main/resources/META-INF/spring/entity-manager-factory.xml",
                "rooAddon/", this.getClass().getClassLoader());
    expected = expected + "\n</beans>";
    WebModuleUtils.updateAppContextDatabase("target/tests/proyecto",
        "rooAddon/", "proyecto", this.getClass().getClassLoader());
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto/proyecto-web/src/main/resources/META-INF/spring/applicationContext.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo copyJettyConfWithoutDatabase.
 */
@Test
public void testCopyJettyConfWithoutDatabase() {

    WebModuleUtils.copyJettyConfWithoutDatabase("target/tests/proyecto",
        "proyecto", "rooAddon/", this.getClass().getClassLoader());
    String[] dirs =
        {"jetty", "jetty/etc", "jetty/logs", "jetty/webapps",
        "jetty/webapps-plus"};
    for (int i = 0; i < dirs.length; i++) {
        File dir =
            new File("target/tests/proyecto/proyecto-web/src/main/"

```

```

        + dirs[i]);
    if (!dir.exists() || !dir.isDirectory()) {
        Assert.fail("Ha fallado");
    }
}
String expected =
    FileUtilities.readTextFile("web/src/main/jetty/jetty.xml",
        "rooAddon/", this.getClass().getClassLoader());
String actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-web/src/main/jetty/jetty.xml");
expected = expected.replace("\r\n", "\n");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
expected =
    FileUtilities.readTextFile(
        "web/src/main/jetty/etc/realm.properties", "rooAddon/", this
            .getClass().getClassLoader());
actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-
web/src/main/jetty/etc/realm.properties");
expected = expected.replace("\r\n", "\n");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
expected =
    FileUtilities.readTextFile("web/src/main/jetty/etc/webdefault.xml",
        "rooAddon/", this.getClass().getClassLoader());
actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto/proyecto-
web/src/main/jetty/etc/webdefault.xml");
expected = expected.replace("\r\n", "\n");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo addLicenseToWebModule.
 */
@Test
public void testAddLicenseToWebModule() {

    String expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + FileUtilities
                .readFileAsString("target/tests/proyecto2/proyecto2-web/pom.xml");
    expected =
        expected.replace(
            "</plugins>",
            FileUtilities.readTextFile("web/plugin-license.xml",
                "rooAddon/", this.getClass().getClassLoader())
                + "\n</plugins>");
    WebModuleUtils.addLicenseToWebModule("target/tests/proyecto2",

```

```

        "rooAddon/", "proyecto2", this.getClass().getClassLoader());
String actual =
    FileUtilities
        .readFileAsString("target/tests/proyecto2/proyecto2-web/pom.xml");
expected = expected.replace("\r\n", "\n");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo addSourcePlugin.
 */
@Test
public void testAddSourcePlugin() {

    String expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + FileUtilities
            .readFileAsString("target/tests/proyecto2/proyecto2-web/pom.xml");
    expected =
        expected.replace(
            "</plugins>",
            FileUtilities.readTextFile("web/plugin-source.xml",
                "rooAddon/", this.getClass().getClassLoader())
            + "\n</plugins>");
    WebModuleUtils.addSourcePlugin(
        "target/tests/proyecto2/proyecto2-web/pom.xml", "rooAddon/", this
            .getClass().getClassLoader());
    String actual =
        FileUtilities
            .readFileAsString("target/tests/proyecto2/proyecto2-web/pom.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo renamePersistence.
 */
@Test
public void testRenamePersistence() {

    String expected =
        "persistence-unit name=\"persistenceUnitRenamepersistence\"";
    WebModuleUtils.renamePersistence("target/tests/renamepersistence",
        "renamepersistence");
    if (new File(
        "target/tests/renamepersistence/renamepersistence-web/src/main/resources/META-
INF/persistence.xml")
        .exists()) {
        Assert.fail("Ha fallado");
    }
    String actual =

```

```

        FileUtilities
            .readFileAsString("target/tests/renamepersistence/renamepersistence-
web/src/main/resources/META-INF/persistenceInfo.xml");
        Assert.assertEquals(expected, actual);
    }
}

```

XmlUtilsTest.java

NOTA: en algunos tests de esta clase se introduce la línea "<?xml version='1.0' encoding='UTF-8' standalone='no'?" al comienzo de el archivo a comparar con el resultado; esto es debido a que esta línea es introducida por las librerías utilizadas para guardar archivos XML, sin que ello signifique que el resultado no es el esperado.

```

package rooplusplus.roo.addon.utils;

import java.io.File;
import java.io.FileWriter;
import java.io.IOException;

import javax.xml.parsers.ParserConfigurationException;
import javax.xml.transform.TransformerException;
import javax.xml.xpath.XPathExpressionException;

import org.apache.commons.io.FileUtils;
import org.junit.After;
import org.junit.Assert;
import org.junit.Before;
import org.junit.Test;
import org.w3c.dom.Document;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;

import rooplusplus.roo.addon.utils.FileUtilities;
import rooplusplus.roo.addon.utils.XmlUtils;

/**
 * Tests de la clase XmlUtils.
 *
 * @author javier.j.menendez
 */
public class XmlUtilsTest {

    /**
     * Código que se ejecutara antes de los tests.
     */
    @Before
    public void before() {

        String xml =
            FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
                .getClassLoader());
        creaArchivoDeTest("addChildToRoot.xml", xml);
        creaArchivoDeTest("addChildToNode.xml", xml);
    }
}

```

```

    creaArchivoDeTest("changeNodeValue.xml", xml);
    creaArchivoDeTest("setNodeAttribute.xml", xml);
    creaArchivoDeTest("appendFileToRoot.xml", xml);
    creaArchivoDeTest("eliminaNodoSiExiste.xml", xml);
    creaArchivoDeTest("guardaXml.xml", xml);
    creaArchivoDeTest("evaluateXPath.xml", xml);
    creaArchivoDeTest("createDocumentFromFile.xml", xml);
    xml =
        FileUtilities.readTextFile("textfiles.txt", "rooAddon/", this
            .getClass().getClassLoader());
    creaArchivoDeTest("xmlNoValido.xml", xml);
}

/**
 * Crea en el directorio "target/tests/" un archivo, para ejecutar las
 * pruebas unitarias.
 *
 * @param nombre Nombre del archivo
 * @param contenido Texto del contenido inicial del archivo
 */
private void creaArchivoDeTest(String nombre, String contenido) {

    File f = new File("target/tests/" + nombre);
    FileWriter fw = null;
    try {
        f.createNewFile();
        fw = new FileWriter(f);
        fw.write(contenido);

    } catch (IOException e) {
        e.printStackTrace();
    } finally {
        if (fw != null) {
            try {
                fw.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}

/**
 * Codigo que se ejecutara tras los tests.
 */
@After
public void after() {

    File dir = new File("target/tests");
    try {
        FileUtils.deleteDirectory(dir);
    } catch (IOException e) {
        e.printStackTrace();
    }
}

```

```

    }
    dir.mkdir();
}

/**
 * Tests del metodo addChildToRoot, introduciendo nodos tanto sin contenido
 * como con el.
 */
@Test
public void testAddChildToRoot() {

    // Nodo con contenido
    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected =
        expected.replace("</project>",
            "<nombredelnodo>valordelnodo</nombredelnodo>\n</project>");
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + expected;
    XmlUtils.addChildToRoot("target/tests/addChildToRoot.xml",
        "nombredelnodo", "valordelnodo");
    String actual =
        FileUtilities.readFileAsString("target/tests/addChildToRoot.xml");
    expected = expected.replace("\r\n", "\n");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
    // Nodo sin contenido
    XmlUtils.addChildToRoot("target/tests/addChildToRoot.xml", "nodovacio",
        "");
    expected = expected.replace("</project>", "<nodovacio/>\n</project>");
    actual =
        FileUtilities.readFileAsString("target/tests/addChildToRoot.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo addChildToNode, introduciendo nodos tanto sin contenido
 * como con el.
 */
@Test
public void testAddChildToNode() {

    // Nodo con contenido
    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected = expected.replace("</parent>", "<nodovacio/>\n</parent>");
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + expected;

```

```

XmlUtils.addChildToNode("target/tests/addChildToNode.xml",
    "/project/parent", "nodovacio", "");
String actual =
    FileUtilities.readFileAsString("target/tests/addChildToNode.xml");
expected = expected.replace("\r\n", "\n");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
// Nodo sin contenido
XmlUtils.addChildToNode("target/tests/addChildToNode.xml",
    "/project/parent/nodovacio", "nombredelnodo", "valordelnodo");
expected =
    expected
        .replace("<nodovacio/>",
            "<nodovacio>\n<nombredelnodo>valordelnodo</nombredelnodo>\n</nodovacio>");
actual =
    FileUtilities.readFileAsString("target/tests/addChildToNode.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo changeNodeValue.
 */
@Test
public void testChangeNodeValue() {

    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + expected;
    expected = expected.replace("\r\n", "\n");
    expected = expected.replace("groupId.groupId.groupId", "New value");
    XmlUtils.changeNodeValue("target/tests/changeNodeValue.xml",
        "/project/parent/groupId", "New value");
    String actual =
        FileUtilities.readFileAsString("target/tests/changeNodeValue.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo setNodeAttribute.
 */
@Test
public void testSetNodeAttribute() {

    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"

```

```

        + expected;
expected = expected.replace("\r\n", "\n");
expected =
    expected.replace("<modules>", "<modules atributo=\"valor\">");
XmlUtils.setNodeAttribute("target/tests/setNodeAttribute.xml",
    "/project/modules", "atributo", "valor");
String actual =
    FileUtilities.readFileAsString("target/tests/setNodeAttribute.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo appendFileToRoot.
 */
@Test
public void testAppendFileToRoot() {

    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    expected =
        expected.replace(
            "</project>",
            FileUtilities.readTextFile("web/resource-ref.xml", "rooAddon/",
                this.getClass().getClassLoader()) + "\n</project>");
    expected = expected.replace("\r\n", "\n");
    XmlUtils.appendFileToRoot("target/tests/appendFileToRoot.xml",
        "rooAddon/", "web/resource-ref.xml", this.getClass()
            .getClassLoader());
    String actual =
        FileUtilities.readFileAsString("target/tests/appendFileToRoot.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo eliminaNodoSiExiste, tanto si el nodo a eliminar existe
 * como si no.
 */
@Test
public void testEliminaNodoSiExiste() {

    // Si el nodo a eliminar existe.
    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;

```



```

expected = expected.replace("\r\n", "\n");
expected = expected.replace("<packaging>pom</packaging>\n\t", "");
XmlUtils.eliminaNodoSiExiste("target/tests/eliminaNodoSiExiste.xml",
    "/project/packaging");
String actual =
    FileUtilities
        .readFileAsString("target/tests/eliminaNodoSiExiste.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
// Si el nodo a eliminar no existe.
XmlUtils.eliminaNodoSiExiste("target/tests/eliminaNodoSiExiste.xml",
    "/project/packaging");
actual =
    FileUtilities
        .readFileAsString("target/tests/eliminaNodoSiExiste.xml");
actual = actual.replace("\r\n", "\n");
Assert.assertEquals(expected, actual);
}

/**
 * Tests del metodo guardaXml.
 */
@Test
public void testGuardaXml() {

    String expected =
        FileUtilities.readTextFile("pom.xml", "rooAddon/", this.getClass()
            .getClassLoader());
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    expected = expected.replace("\r\n", "\n");
    expected =
        expected.replace("</project>", "<nodo>valor</nodo>\n</project>");
    Document doc;
    try {
        doc = XmlUtils.createDocumentFromFile("target/tests/guardaXml.xml");
        Node node = doc.createElement("nodo");
        node.setTextContent("valor");
        doc.getDocumentElement().appendChild(node);
        XmlUtils.guardaXml(doc, "target/tests/guardaXml.xml");
        String actual =
            FileUtilities.readFileAsString("target/tests/guardaXml.xml");
        actual = actual.replace("\r\n", "\n");
        Assert.assertEquals(expected, actual);
    } catch (ParserConfigurationException e) {
        Assert.fail("Ha fallado");
    } catch (SAXException e) {
        Assert.fail("Ha fallado");
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (TransformerException e) {
        Assert.fail("Ha fallado");
    }
}

```

```

    }
}

/**
 * Tests del metodo evaluateXPath, tanto cuando se usa un XPath que apunta a
 * nodos existentes como cuando no.
 */
@Test
public void testEvaluateXPath() {

    Document doc;
    try {
        // Cuando el XPath se refiere a nodos existentes
        doc =
            XmlUtils
                .createDocumentFromFile("target/tests/evaluateXPath.xml");
        NodeList list =
            XmlUtils.evaluateXPath("/project/modules/module", doc);
        if (list.getLength() != 6) {
            Assert.fail("Ha fallado");
        }
        String[] valores =
            {"core", "docs", "scripts", "web", "ear", "package"};
        for (int i = 0; i < 6; i++) {
            if (!list.item(i).getNodeName().equals("module")
                || !list.item(i).getTextContent()
                    .equals("${project}-" + valores[i])) {
                Assert.fail("Ha fallado");
            }
        }
        // Cuando el XPath se refiere a nodos no existentes
        list = XmlUtils.evaluateXPath("/este/nodo/no/existe", doc);
        if (list.getLength() != 0) {
            Assert.fail("Ha fallado");
        }
    } catch (ParserConfigurationException e) {
        Assert.fail("Ha fallado");
    } catch (SAXException e) {
        Assert.fail("Ha fallado");
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (XPathExpressionException e) {
        Assert.fail("Ha fallado");
    }
}

/**
 * Tests del metodo createDocumentFromFile, tanto en el caso de que el
 * archivo exista, como el de que no, como el de que no contenga XML valido.
 */
@Test
public void testCreateDocumentFromFile() {

```

```

// Cuando el archivo existe y contiene XML valido
try {
    Document doc =
        XmlUtils
            .createDocumentFromFile("target/tests/createDocumentFromFile.xml");
    XmlUtils.guardaXml(doc, "target/tests/resultado.xml");
    String expected =
        FileUtilities
            .readFileAsString("target/tests/createDocumentFromFile.xml");
    expected =
        "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
        + expected;
    expected = expected.replace("\r\n", "\n");
    String actual =
        FileUtilities.readFileAsString("target/tests/resultado.xml");
    actual = actual.replace("\r\n", "\n");
    Assert.assertEquals(expected, actual);
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
} catch (IOException e) {
    Assert.fail("Ha fallado");
} catch (TransformerException e) {
    Assert.fail("Ha fallado");
}
}
// Cuando el archivo no existe
try {
    Document doc =
        XmlUtils
            .createDocumentFromFile("target/tests/archivoQueNoExiste.xml");
    Assert.fail("Ha fallado");
} catch (ParserConfigurationException e) {
    Assert.fail("Ha fallado");
} catch (SAXException e) {
    Assert.fail("Ha fallado");
} catch (IOException e) {
}

}
// Cuando el archivo no contiene XML valido
try {
    Document doc =
        XmlUtils.createDocumentFromFile("target/tests/xmlNoValido.xml");
    Assert.fail("Ha fallado");
} catch (ParserConfigurationException e) {

} catch (SAXException e) {

} catch (IOException e) {
    Assert.fail("Ha fallado");
}
}
}

```

```

/**
 * Tests del metodo createDocumentFromString, tanto cuando la cadena
 * contiene XML valido como cuando no lo contiene, en diferentes
 * posibilidades en este ultimo caso (no es XML, tiene varias raices, no se
 * cierra algun elemento).
 */
@Test
public void testCreateDocumentFromString() {

    // Cuando el String contiene XML valido
    try {
        String expected = "<raiz> <nodol/> </raiz>";
        Document doc = XmlUtils.createDocumentFromString(expected);
        expected =
            "<?xml version=\"1.0\" encoding=\"UTF-8\" standalone=\"no\"?>\n"
            + expected;
        XmlUtils.guardaXml(doc, "target/tests/guardado.xml");
        String actual =
            FileUtilities.readFileAsString("target/tests/guardado.xml");
        actual = actual.replace("\r\n", "\n");
        Assert.assertEquals(expected, actual);
    } catch (SAXException e) {
        Assert.fail("Ha fallado");
    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (ParserConfigurationException e) {
        Assert.fail("Ha fallado");
    } catch (TransformerException e) {
        Assert.fail("Ha fallado");
    }
    // Cuando el String contiene XML con varias raices
    try {
        Document doc =
            XmlUtils
                .createDocumentFromString("<raiz1> <nodol/> </raiz1> <raiz2> <nodol/>
</raiz2>");
        Assert.fail("Ha fallado");
    } catch (SAXException e) {

    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (ParserConfigurationException e) {

    }
    // Cuando el String contiene XML con nodos sin cerrar
    try {
        Document doc = XmlUtils.createDocumentFromString("<raiz> <nodol/>");
        Assert.fail("Ha fallado");
    } catch (SAXException e) {

    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (ParserConfigurationException e) {

```

```

    }
    // Cuando el String no contiene XML
    try {
        Document doc = XmlUtils.createDocumentFromString("Esto no es XML.");
        Assert.fail("Ha fallado");
    } catch (SAXException e) {

    } catch (IOException e) {
        Assert.fail("Ha fallado");
    } catch (ParserConfigurationException e) {

    }
}
}
}

```

ANEXO II: Resources del JAR de Roo++

A continuación aparece el contenido de los diferentes archivos que se encuentran en el directorio *resources/rooAddon*, dentro del JAR de Roo++. Los archivos que pertenecen a algún módulo, tal como *core* o *web*, se encuentran agrupados según dicho módulo al que pertenecen.

dirs.txt

```

core
core/src
core/src/main
core/src/main/java
core/src/main/resources
core/src/test
core/src/test/java
core/src/test/resources
docs
docs/src
docs/src/assembly
docs/src/docbkx
docs/src/fonts
docs/src/front_pages
docs/src/images
ear
ear/src
ear/src/main
ear/src/main/application
ear/src/main/application/META-INF
package
package/src
package/src/assembly
scripts
scripts/src
scripts/src/assembly
scripts/src/main
scripts/src/main/resources

```

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>groupId.groupId.groupId</groupId>
    <artifactId>artifactIdartifactId</artifactId>
    <version>versionversion</version>
  </parent>
  <groupId>${project.groupId}</groupId>
  <artifactId>${project.artifactId}</artifactId>
  <packaging>pom</packaging>
  <version>${project.version}</version>
  <name>${project.name}</name>
  <modules>
    <module>${project}-core</module>
    <module>${project}-docs</module>
    <module>${project}-scripts</module>
    <module>${project}-web</module>
    <module>${project}-ear</module>
    <module>${project}-package</module>
  </modules>
</project>
```

textfiles.txt

```
pom.xml
core/pom.xml
docs/pom.xml
docs/src/InstallationManual.xml
docs/src/UserManual.xml
docs/src/assembly/package.xml
ear/pom.xml
ear/src/main/application/META-INF/weblogic-application.xml
package/pom.xml
package/src/assembly/package.xml
scripts/pom.xml
scripts/src/assembly/compress-zip.xml
```

core

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>${project.parent.groupId}</groupId>
    <artifactId>${project.parent.artifactId}</artifactId>
    <version>${project.parent.version}</version>
  </parent>
  <groupId>${project.parent.groupId}</groupId>
  <artifactId>${project.parent.artifactId}-core</artifactId>
```

```

<version>${project.version}</version>
<name>${project.parent.name}-core</name>
<packaging>jar</packaging>
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-source-plugin</artifactId>
      <version>2.1.1</version>
      <executions>
        <execution>
          <id>attach-sources</id>
          <phase>verify</phase>
          <goals>
            <goal>jar-no-fork</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>
</project>

```

docs

pom.xml

```

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>${project.parent.groupId}</groupId>
    <artifactId>${project.parent.artifactId}</artifactId>
    <version>${project.parent.version}</version>
  </parent>
  <groupId>${project.parent.groupId}</groupId>
  <artifactId>${project.parent.artifactId}-docs</artifactId>
  <version>${project.version}</version>
  <name>${project.parent.name}-docs</name>
  <packaging>pom</packaging>
  <build>
    <plugins>
      <plugin>
        <groupId>com.agilejava.docbkk</groupId>
        <artifactId>docbkk-maven-plugin</artifactId>
        <version>2.0.8</version>
        <executions>
          <execution>
            <id>generate-help-pdf</id>
            <goals>
              <goal>generate-pdf</goal>
            </goals>
            <phase>generate-resources</phase>
            <configuration>
              <foCustomization>${
{basedir}/src/docbkk/custom.xml</foCustomization>
              <paperType>A4</paperType>
              <imgSrcPath>${basedir}/src/</imgSrcPath>
            </configuration>
          </execution>
        </executions>
      </plugin>
    </plugins>
    <configuration>
      <sourceDirectory>src</sourceDirectory>
      <xincludeSupported>true</xincludeSupported>
      <includes>*Manual.xml</includes>
      <targetDirectory>target/help-pdf</targetDirectory>
    </configuration>
    <dependencies>
      <dependency>
        <groupId>net.sf.docbook</groupId>
        <artifactId>docbook-xml</artifactId>
        <version>5.0-all</version>
        <classifier>resources</classifier>
        <type>zip</type>
        <scope>runtime</scope>
      </dependency>
    </dependencies>
  </build>
</project>

```

```

        </dependency>
        <dependency>
            <groupId>com.sun.media</groupId>
            <artifactId>jai_codec</artifactId>
            <version>1.1.2_01</version>
        </dependency>
    </dependencies>
</plugin>
<plugin>
    <groupId>com.agilejava.docbkx</groupId>
    <artifactId>docbkx-fop-support</artifactId>
    <version>2.0.11</version>
    <executions>
        <execution>
            <phase>generate-sources</phase>
            <goals>
                <goal>generate</goal>
            </goals>
            <configuration>
                <ansi>true</ansi>
            </configuration>
        </execution>
    </executions>
</plugin>
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-assembly-plugin</artifactId>
    <version>2.4</version>
    <configuration>
        <descriptors>
            <descriptor>src/assembly/package.xml</descriptor>
        </descriptors>
        <appendAssemblyId>>false</appendAssemblyId>
    </configuration>
    <executions>
        <execution>
            <id>generate-am-help</id>
            <phase>package</phase>
            <goals>
                <goal>attached</goal>
            </goals>
        </execution>
    </executions>
</plugin>
</plugins>
</build>
</project>

```

src/InstallationManual.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- <!DOCTYPE book PUBLIC "-//OASIS//DTD DocBook XML V4.5//EN" -->
<!-- "http://www.oasis-open.org/docbook/xml/4.5/docbookx.dtd" -->
<book version="5.0" xml:id="help" xmlns="http://docbook.org/ns/docbook"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:ns5="http://www.w3.org/1998/Math/MathML"
    xmlns:ns4="http://www.w3.org/1999/xhtml"
    xmlns:ns3="http://www.w3.org/2000/svg"
    xmlns:ns="http://docbook.org/ns/docbook">
    <title>Title</title>
    <subtitle>Subtitle</subtitle>
    <chapter>
        <title>Chapter</title>
        <para>.</para>
        <section>
            <title>Section</title>
            <para>.</para>
        </section>
    </chapter>
</book>

```

src/UserManual.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- <!DOCTYPE book PUBLIC "-//OASIS//DTD DocBook XML V4.5//EN" -->
<!-- "http://www.oasis-open.org/docbook/xml/4.5/docbookx.dtd" -->
<book version="5.0" xml:id="help" xmlns="http://docbook.org/ns/docbook"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:ns5="http://www.w3.org/1998/Math/MathML"
    xmlns:ns4="http://www.w3.org/1999/xhtml"
    xmlns:ns3="http://www.w3.org/2000/svg"
    xmlns:ns="http://docbook.org/ns/docbook">
    <title>Title</title>
    <subtitle>Subtitle</subtitle>
    <chapter>
        <title>Chapter</title>
        <para>.</para>
        <section>
            <title>Section</title>
            <para>.</para>
        </section>
    </chapter>
</book>

```


src/assembly/package.xml

```
<assembly
  xmlns="http://maven.apache.org/plugins/maven-assembly-plugin/assembly/1.1.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/plugins/maven-assembly-plugin/assembly/1.1.2
http://maven.apache.org/xsd/assembly-1.1.2.xsd">
  <id>zip</id>
  <includeBaseDirectory>false</includeBaseDirectory>
  <formats>
    <format>zip</format>
  </formats>
  <fileSets>
    <fileSet>
      <directory>target</directory>
      <outputDirectory></outputDirectory>
      <excludes>
        <exclude>dependency-maven-plugin-markers</exclude>
      </excludes>
      <includes>
        <include>*.pdf</include>
      </includes>
    </fileSet>
  </fileSets>
</assembly>
```

ear

pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>${project.parent.groupId}</groupId>
    <artifactId>${project.parent.artifactId}</artifactId>
    <version>${project.parent.version}</version>
  </parent>
  <groupId>${project.parent.groupId}</groupId>
  <artifactId>${project.parent.artifactId}-ear</artifactId>
  <version>${project.version}</version>
  <name>${project.parent.name}-ear</name>
  <packaging>ear</packaging>
  <dependencies>
    <dependency>
      <groupId>${project.groupId}</groupId>
      <artifactId>${project.parent.artifactId}-web</artifactId>
      <version>${project.version}</version>
      <type>war</type>
    </dependency>
  </dependencies>
  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-ear-plugin</artifactId>
        <version>2.5</version>
        <configuration>
          <finalName>${project.parent.artifactId}</finalName>
          <modules>
            <webModule>
              <groupId>${project.groupId}</groupId>
              <artifactId>${project.parent.artifactId}-web</artifactId>
              <bundleFileName>${project.parent.artifactId}-web.war</bundleFileName>
              <contextRoot>/${project.parent.artifactId}-web</contextRoot>
            </webModule>
          </modules>
        </configuration>
      </plugin>
    </plugins>
  </build>
</project>
```

src/main/application/META-INF/weblogic-application.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<weblogic-application xmlns="http://www.bea.com/ns/weblogic/10"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.bea.com/ns/weblogic/10
http://xmlns.oracle.com/weblogic/weblogic-application/1.0/weblogic-application.xsd">
  <prefer-application-packages>
    <package-name>antlr.*</package-name>
    <package-name>org.apache.commons.*</package-name>
    <package-name>org.apache.xmlbeans.*</package-name>
    <package-name>org.springframework.*</package-name>
    <package-name>org.hibernate.*</package-name>
    <package-name>javax.persistence.*</package-name>
  </prefer-application-packages>
```

```

</prefer-application-packages>
</weblogic-application>

```

package

pom.xml

```

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>${project.parent.groupId}</groupId>
        <artifactId>${project.parent.artifactId}</artifactId>
        <version>${project.parent.version}</version>
    </parent>
    <groupId>${project.parent.groupId}</groupId>
    <artifactId>${project.parent.artifactId}-package</artifactId>
    <version>${project.version}</version>
    <name>${project.parent.name}-package</name>
    <packaging>pom</packaging>
    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-dependency-plugin</artifactId>
                <executions>
                    <execution>
                        <id>copy-apps</id>
                        <phase>generate-sources</phase>
                        <goals>
                            <goal>copy</goal>
                        </goals>
                        <configuration>
                            <artifactItems>
                                <artifactItem>
                                    <groupId>${
{project.groupId}</groupId>
                                    <artifactId>${
{project.parent.artifactId}-web</artifactId>
                                    <version>${
{project.version}</version>
                                    <type>war</type>
                                    <destFileName>${
{project.parent.artifactId}-web.war</destFileName>
                                </artifactItem>
                                <artifactItem>
                                    <groupId>${
{project.groupId}</groupId>
                                    <artifactId>${
{project.parent.artifactId}-scripts</artifactId>
                                    <version>${
{project.version}</version>
                                    <type>zip</type>
                                    <destFileName>${
{project.parent.artifactId}-scripts.zip</destFileName>
                                </artifactItem>
                                <artifactItem>
                                    <groupId>${
{project.groupId}</groupId>
                                    <artifactId>${
{project.parent.artifactId}-docs</artifactId>
                                    <version>${
{project.version}</version>
                                    <type>zip</type>
                                    <destFileName>${
{project.parent.artifactId}-docs.zip</destFileName>
                                </artifactItem>
                                <artifactItem>
                                    <groupId>${
{project.groupId}</groupId>
                                    <artifactId>${
{project.parent.artifactId}-ear</artifactId>
                                    <version>${
{project.version}</version>
                                    <type>ear</type>
                                    <destFileName>${
{project.parent.artifactId}-web.ear</destFileName>
                                </artifactItem>
                            </artifactItems>
                        </outputDirectory>
                    <stripVersion>true</stripVersion>
                </configuration>
            </execution>
        </executions>
    </plugin>
    <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-assembly-plugin</artifactId>

```

```

        <version>2.4</version>
        <configuration>
            <descriptors>
                <descriptor>src/assembly/package.xml</descriptor>
            </descriptors>
            <appendAssemblyId>>true</appendAssemblyId>
        </configuration>
    </executions>
        <execution>
            <id>generate-package</id>
            <phase>package</phase>
            <goals>
                <goal>attached</goal>
            </goals>
        </execution>
    </executions>
</plugin>
</plugins>
</build>
</project>

```

src/assembly/package.xml

```

<assembly
  xmlns="http://maven.apache.org/plugins/maven-assembly-plugin/assembly/1.1.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/plugins/maven-assembly-plugin/assembly/1.1.2
http://maven.apache.org/xsd/assembly-1.1.2.xsd">
  <id>zip</id>
  <includeBaseDirectory>false</includeBaseDirectory>
  <formats>
    <format>zip</format>
  </formats>
  <fileSets>
    <fileSet>
      <directory>target/package</directory>
      <outputDirectory></outputDirectory>
      <excludes>
        <exclude>dependency-maven-plugin-markers</exclude>
      </excludes>
      <includes>
        <include>**/*</include>
      </includes>
    </fileSet>
  </fileSets>
</assembly>

```

scripts

pom.xml

```

<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>${project.parent.groupId}</groupId>
    <artifactId>${project.parent.artifactId}</artifactId>
    <version>${project.parent.version}</version>
  </parent>
  <groupId>${project.parent.groupId}</groupId>
  <artifactId>${project.parent.artifactId}-scripts</artifactId>
  <version>${project.version}</version>
  <name>${project.parent.name}-scripts</name>
  <packaging>pom</packaging>
  <build>
    <finalName>${project.parent.artifactId}-scripts</finalName>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-assembly-plugin</artifactId>
        <version>2.4</version>
        <configuration>
          <descriptors>
            <descriptor>src/assembly/compress-zip.xml</descriptor>
          </descriptors>
          <appendAssemblyId>>false</appendAssemblyId>
        </configuration>
        <executions>
          <execution>
            <goals>
              <goal>attached</goal>
            </goals>
            <phase>package</phase>
          </execution>
        </executions>
      </plugin>
    </plugins>
  </build>
</project>

```

src/assembly/compress-zip.xml

```
<assembly
  xmlns="http://maven.apache.org/plugins/maven-assembly-plugin/assembly/1.1.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/plugins/maven-assembly-plugin/assembly/1.1.2
http://maven.apache.org/xsd/assembly-1.1.2.xsd">
  <id>zip</id>
  <formats>
    <format>zip</format>
  </formats>
  <includeBaseDirectory>false</includeBaseDirectory>
  <fileSets>
    <fileSet>
      <directory>src/main/resources</directory>
      <outputDirectory></outputDirectory>
    </fileSet>
  </fileSets>
</assembly>
```

web

plugin-license.xml

```
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>license-maven-plugin</artifactId>
<version>1.0</version>
<configuration>
  <excludedScopes>test,system,provided</excludedScopes>
  <generateBundle>false</generateBundle>
  <groupByLicense>true</groupByLicense>
  <useMissingFile>true</useMissingFile>
  <useRepositoryMissingFiles>true</useRepositoryMissingFiles>
  <deployMissingFile>true</deployMissingFile>
  <force>false</force>
  <licenseMerges>
    <licenseMerge><![CDATA[The Apache Software License, Version 2.0|Apache
License, Version 2.0|Apache License Version 2.0|Apache 2]]></licenseMerge>
    <licenseMerge><![CDATA[GNU LESSER GENERAL PUBLIC LICENSE|GNU Lesser General
Public Licence|GNU Lesser General Public License]]></licenseMerge>
    <licenseMerge><![CDATA[Eclipse Public License v1.0|Eclipse Public License - v
1.0]]></licenseMerge>
  </licenseMerges>
</configuration>
<executions>
  <execution>
    <id>add-third-party</id>
    <goals>
      <goal>add-third-party</goal>
    </goals>
    <phase>process-sources</phase>
  </execution>
</executions>
</plugin>
```

plugin-source.xml

```
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-source-plugin</artifactId>
<version>2.1.1</version>
<executions>
  <execution>
    <id>attach-sources</id>
    <phase>verify</phase>
    <goals>
      <goal>jar-no-fork</goal>
    </goals>
  </execution>
</executions>
</plugin>
```

resource-ref.xml

```
<resource-ref>
  <description>ConnectionPool DataSource Reference</description>
  <res-ref-name>jdbc/testDS</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
</resource-ref>
```

src/main/jetty/jetty-database.xml

```
<!-- DataSources CIDI-->
  <New id="testDS" class="org.mortbay.jetty.plus.naming.Resource">
```

```

        <Arg>jdbc/testDS</Arg>
        <Arg>
            <New class="org.apache.commons.dbcp.BasicDataSource">
                <Set name="driverClassName">databaseDriver</Set>
                <Set name="url">databaseUrl</Set>
                <Set name="username">user</Set>
                <Set name="password">password</Set>
            </New>
        </Arg>
    </New>

```

src/main/jetty/jetty.xml

```

    <?xml version="1.0"?>
<!DOCTYPE Configure PUBLIC "-//Mort Bay Consulting//DTD Configure//EN"
"http://jetty.mortbay.org/configure.dtd">
<!-- ===== -->
<!-- Configure the Jetty Server -->
<!-- Documentation of this file format can be found at: -->
<!-- http://docs.codehaus.org/display/JETTY/jetty.xml -->
<!-- ===== -->
<Configure id="Server" class="org.mortbay.jetty.Server">
    <!-- ===== -->
    <!-- Server Thread Pool -->
    <!-- ===== -->
    <Set name="ThreadPool">
        <New class="org.mortbay.thread.QueuedThreadPool">
            <Set name="minThreads">10</Set>
            <Set name="maxThreads">200</Set>
            <Set name="lowThreads">20</Set>
            <Set name="SpawnOrShrinkAt">2</Set>
        </New>
        <!--
        Optional Java 5 bounded threadpool with job queue <New
        class="org.mortbay.thread.concurrent.ThreadPool"> <Set
        name="corePoolSize">50</Set> <Set name="maximumPoolSize">50</Set>
        </New>
        -->
    </Set>

    <!-- ===== -->
    <!-- Set connectors -->
    <!-- ===== -->
    <!-- One of each type! -->
    <!-- ===== -->
    <!--
    Use this connector for many frequently idle connections and for
    threadless continuations.
    -->
    <Call name="addConnector">
        <Arg>
            <New class="org.mortbay.jetty.nio.SelectChannelConnector">
                <Set name="host">
                    <SystemProperty name="COMPUTERNAME" />
                </Set>
                <Set name="port">
                    <SystemProperty name="jetty.port" default="8080" />
                </Set>
                <Set name="maxIdleTime">30000</Set>
                <Set name="Acceptors">2</Set>
                <Set name="statsOn">false</Set>
                <Set name="confidentialPort">8443</Set>
                <Set name="lowResourcesConnections">5000</Set>
                <Set name="lowResourcesMaxIdleTime">5000</Set>
            </New>
        </Arg>
    </Call>
    <!--
    Use this connector if NIO is not available. <Call name="addConnector">
    <Arg> <New class="org.mortbay.jetty.bio.SocketConnector"> <Set
    name="port">8081</Set> <Set name="maxIdleTime">50000</Set> <Set
    name="lowResourceMaxIdleTime">1500</Set> </New> </Arg> </Call>
    -->

    <!-- ===== -->
    <!-- To add a HTTPS SSL listener -->
    <!-- see jetty-ssl.xml to add an ssl connector. use -->
    <!-- java -jar start.jar etc/jetty.xml etc/jetty-ssl.xml -->
    <!-- ===== -->

    <!-- ===== -->
    <!-- Set up global session ID manager -->
    <!-- ===== -->
    <!--
    <Set name="sessionIdManager"> <New
    class="org.mortbay.jetty.servlet.HashSessionIdManager"> <Set
    name="workerName">node1</Set> </New> </Set>
    -->

    <!-- ===== -->
    <!-- Set handler Collection Structure -->
    <!-- ===== -->

```

```

        <Set name="handler">
            <New id="Handlers" class="org.mortbay.jetty.handler.HandlerCollection">
                <Set name="handlers">
                    <Array type="org.mortbay.jetty.Handler">
                        <Item>
                            <New id="Contexts"
class="org.mortbay.jetty.handler.ContextHandlerCollection" />
                        </Item>
                        <Item>
                            <New id="DefaultHandler"
class="org.mortbay.jetty.handler.DefaultHandler" />
                        </Item>
                        <Item>
                            <New id="RequestLog"
class="org.mortbay.jetty.handler.RequestLogHandler" />
                        </Item>
                    </Array>
                </Set>
            </New>
        </Set>

<!-- ===== -->
<!-- Configure the context deployer -->
<!-- A context deployer will deploy contexts described in -->
<!-- configuration files discovered in a directory. -->
<!-- The configuration directory can be scanned for hot -->
<!-- deployments at the configured scanInterval. -->
<!-- This deployer is configured to deploy contexts configured -->
<!-- in the $JETTY_HOME/contexts directory -->
<!-- ===== -->
        <Call name="addLifeCycle">
            <Arg>
                <New class="org.mortbay.jetty.deployer.ContextDeployer">
                    <Set name="contexts">
                        <Ref id="Contexts" />
                    </Set>
                    <Set name="configurationDir">
                        <SystemProperty name="jetty.home" default="src/main/jetty"
/>/contexts</Set>
                    <Set name="scanInterval">1</Set>
                </New>
            </Arg>
        </Call>

<!-- ===== -->
<!-- Configure the webapp deployer. -->
<!-- A webapp deployer will deploy standard webapps discovered -->
<!-- in a directory at startup, without the need for additional -->
<!-- configuration files. It does not support hot deploy or -->
<!-- non standard contexts (see ContextDeployer above). -->
<!-- This deployer is configured to deploy webapps from the -->
<!-- $JETTY_HOME/webapps directory -->
<!-- Normally only one type of deployer need be used. -->
<!-- ===== -->
        <Call name="addLifeCycle">
            <Arg>
                <New class="org.mortbay.jetty.deployer.WebAppDeployer">
                    <Set name="contexts">
                        <Ref id="Contexts" />
                    </Set>
                    <Set name="webAppDir">
                        <SystemProperty name="jetty.home"
default="src/main/jetty"/>/webapps</Set>
                    <Set name="parentLoaderPriority">>false</Set>
                    <Set name="extract">>true</Set>
                    <Set name="allowDuplicates">>false</Set>
                    <Set name="defaultsDescriptor">
                        <SystemProperty name="jetty.home" default="src/main/jetty"
/>/etc/webdefault.xml</Set>
                </New>
            </Arg>
        </Call>

<!-- ===== -->
<!-- Configurations for WebAppContexts -->
<!-- Sequence of configurations to enable Plus features. -->
<!-- ===== -->
        <Array id="plusConfig" type="java.lang.String">
            <Item>org.mortbay.jetty.webapp.WebInfConfiguration
            </Item>
            <Item>org.mortbay.jetty.plus.webapp.EnvConfiguration
            </Item>
            <Item>org.mortbay.jetty.plus.webapp.Configuration
            </Item>
            <Item>org.mortbay.jetty.webapp.JettyWebXmlConfiguration
            </Item>
            <Item>org.mortbay.jetty.webapp.TagLibConfiguration
            </Item>
        </Array>

<!-- ===== -->
<!-- Deploy all webapps in webapps-plus -->
        <Call name="addLifeCycle">

```

```

        <Arg>
            <New class="org.mortbay.jetty.deployer.WebAppDeployer">
                <Set name="contexts">
                    <Ref id="Contexts" />
                </Set>
                <Set name="webAppDir">
                    <SystemProperty name="jetty.home" default="src/main/jetty"
/>/webapps-plus</Set>
                <Set name="parentLoaderPriority">false</Set>
                <Set name="extract">true</Set>
                <Set name="allowDuplicates">false</Set>
                <Set name="defaultsDescriptor">
                    <SystemProperty name="jetty.home" default="src/main/jetty"
/>/etc/webdefault.xml</Set>
                <Set name="configurationClasses">
                    <Ref id="plusConfig" />
                </Set>
            </New>
        </Arg>
    </Call>

<!-- ===== -->
<!-- Configure Authentication Realms -->
<!-- Realms may be configured for the entire server here, or -->
<!-- they can be configured for a specific web app in a context -->
<!-- configuration (see $(jetty.home)/contexts/test.xml for an -->
<!-- example). -->
<!-- ===== -->
    <Set name="UserRealms">
        <Array type="org.mortbay.jetty.security.UserRealm">
            <Item>
                <New class="org.mortbay.jetty.security.HashUserRealm">
                    <Set name="name">Test Realm</Set>
                    <Set name="config">
                        <SystemProperty name="jetty.home"
default="src/main/jetty" />/etc/realm.properties</Set>
                    <Set name="refreshInterval">0</Set>
                </New>
            </Item>
        </Array>
    </Set>

<!-- ===== -->
<!-- Configure Request Log -->
<!-- Request logs may be configured for the entire server here, -->
<!-- or they can be configured for a specific web app in a -->
<!-- contexts configuration (see $(jetty.home)/contexts/test.xml -->
<!-- for an example). -->
<!-- ===== -->
    <Ref id="RequestLog">
        <Set name="requestLog">
            <New id="RequestLogImpl" class="org.mortbay.jetty.NCSARequestLog">
                <Set name="filename">
                    <SystemProperty name="jetty.logs"
default="src/main/jetty/logs" />/yyyy_mm_dd.request.log</Set>
                <Set name="filenameDateFormat">yyyy_MM_dd</Set>
                <Set name="retainDays">90</Set>
                <Set name="append">true</Set>
                <Set name="extended">true</Set>
                <Set name="logCookies">false</Set>
                <Set name="LogTimeZone">GMT</Set>
            </New>
        </Set>
    </Ref>

<!-- ===== -->
<!-- extra options -->
<!-- ===== -->
    <Set name="stopAtShutdown">true</Set>
    <Set name="sendServerVersion">true</Set>
    <Set name="sendDateHeader">true</Set>
    <Set name="gracefulShutdown">1000</Set>
</Configure>

```

src/main/jetty/etc/realm.properties

```

#
# This file defines users passwords and roles for a HashUserRealm
#
# The format is
# <username>: <password>[,<rolename> ...]
#
# Passwords may be clear text, obfuscated or checksummed. The class
# org.mortbay.util.Password should be used to generate obfuscated
# passwords or password checksums
#
# If DIGEST Authentication is used, the password must be in a recoverable
# format, either plain text or OBF:.
#
jetty: MD5:164c88b302622e17050af52c89945d44,user
admin: CRYPT:adlks.kc.1Ug,server-administrator,content-administrator,admin
other: OBF:1xmk1w261u9rlw1clxmq
plain: plain
user: password

```

```
# This entry is for digest auth. The credential is a MD5 hash of username:realmname:password
digest: MD5:6e120743ad67abfbc385bc2bb754e297
```

src/main/jetty/etc/webdefault.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- ===== -->
<!-- This file contains the default descriptor for web applications. -->
<!-- ===== -->
<!-- The intent of this descriptor is to include jetty specific or common -->
<!-- configuration for all webapps. If a context has a webdefault.xml -->
<!-- descriptor, it is applied before the contexts own web.xml file -->
<!-- ===== -->
<!-- A context may be assigned a default descriptor by: -->
<!-- + Calling WebApplicationContext.setDefaultsDescriptor -->
<!-- + Passed an arg to addWebApplications -->
<!-- ===== -->
<!-- This file is used both as the resource within the jetty.jar (which is -->
<!-- used as the default if no explicit defaults descriptor is set) and it -->
<!-- is copied to the etc directory of the Jetty distro and explicitly -->
<!-- by the jetty.xml file. -->
<!-- ===== -->
<web-app
  xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-
app_2_5.xsd"
  metadata-complete="true"
  version="2.5">
  <description>
    Default web.xml file.
    This file is applied to a Web application before it's own WEB_INF/web.xml file
  </description>

  <!-- ===== -->
  <!-- Context params to control Session Cookies -->
  <!-- ===== -->
  <!-- UNCOMMENT TO ACTIVATE -->
  <context-param>
    <param-name>org.mortbay.jetty.servlet.SessionDomain</param-name>
    <param-value>127.0.0.1</param-value>
  </context-param>

  <context-param>
    <param-name>org.mortbay.jetty.servlet.SessionPath</param-name>
    <param-value></param-value>
  </context-param>

  <context-param>
    <param-name>org.mortbay.jetty.servlet.MaxAge</param-name>
    <param-value>-1</param-value>
  </context-param>
  -->

  <context-param>
    <param-name>org.mortbay.jetty.webapp.NoTLDJarPattern</param-name>
    <param-value>start.jar|ant-*.jar|dojo-*.jar|jetty-*.jar|jsp-api-*.jar|junit-*.jar|
servlet-api-*.jar|dnsns.jar|rt.jar|jsse.jar|tools.jar|sunpkcs11.jar|sunjce_provider.jar|
xerces.jar</param-value>
  </context-param>

  <!-- ===== -->
  <!-- The default servlet. -->
  <!-- This servlet, normally mapped to /, provides the handling for static -->
  <!-- content, OPTIONS and TRACE methods for the context. -->
  <!-- The following initParameters are supported: -->
  <!-- ===== -->
  <!-- acceptRanges If true, range requests and responses are -->
  <!-- supported -->
  <!-- ===== -->
  <!-- dirAllowed If true, directory listings are returned if no -->
  <!-- welcome file is found. Else 403 Forbidden. -->
  <!-- ===== -->
  <!-- welcomeServlets If true, attempt to dispatch to welcome files -->
  <!-- that are servlets, if no matching static -->
  <!-- resources can be found. -->
  <!-- ===== -->
  <!-- redirectWelcome If true, redirect welcome file requests -->
  <!-- else use request dispatcher forwards -->
  <!-- ===== -->
  <!-- gzip If set to true, then static content will be served -->
  <!-- as gzip content encoded if a matching resource is -->
  <!-- found ending with ".gz" -->
  <!-- ===== -->
  <!-- resoureBase Can be set to replace the context resource base -->
  <!-- ===== -->
  <!-- relativeResourceBase Set with a pathname relative to the base of the -->
  <!-- servlet context root. Useful for only serving -->
  <!-- static content from only specific subdirectories. -->
```



```

<!-- useFileMappedBuffer -->
<!-- If set to true (the default), a memory mapped -->
<!-- file buffer will be used to serve static content -->
<!-- when using an NIO connector. Setting this value -->
<!-- to false means that a direct buffer will be used -->
<!-- instead. If you are having trouble with Windows -->
<!-- file locking, set this to false. -->
<!-- cacheControl If set, all static content will have this value -->
<!-- set as the cache-control header. -->
<!-- maxCacheSize Maximum size of the static resource cache -->
<!-- maxCachedFileSize Maximum size of any single file in the cache -->
<!-- maxCachedFiles Maximum number of files in the cache -->
<!-- cacheType "nio", "bio" or "both" to determine the type(s) -->
<!-- of resource cache. A bio cached buffer may be used -->
<!-- by nio but is not as efficient as a nio buffer. -->
<!-- An nio cached buffer may not be used by bio. -->
<!-- ----- -->
</servlet>
<servlet-name>default</servlet-name>
<servlet-class>org.mortbay.jetty.servlet.DefaultServlet</servlet-class>
<init-param>
  <param-name>acceptRanges</param-name>
  <param-value>>true</param-value>
</init-param>
<init-param>
  <param-name>dirAllowed</param-name>
  <param-value>>true</param-value>
</init-param>
<init-param>
  <param-name>welcomeServlets</param-name>
  <param-value>>false</param-value>
</init-param>
<init-param>
  <param-name>redirectWelcome</param-name>
  <param-value>>false</param-value>
</init-param>
<init-param>
  <param-name>maxCacheSize</param-name>
  <param-value>256000000</param-value>
</init-param>
<init-param>
  <param-name>maxCachedFileSize</param-name>
  <param-value>10000000</param-value>
</init-param>
<init-param>
  <param-name>maxCachedFiles</param-name>
  <param-value>1000</param-value>
</init-param>
<init-param>
  <param-name>cacheType</param-name>
  <param-value>both</param-value>
</init-param>
<init-param>
  <param-name>gzip</param-name>
  <param-value>>true</param-value>
</init-param>
<init-param>
  <param-name>useFileMappedBuffer</param-name>
  <param-value>>false</param-value>
</init-param>
<!--
<init-param>
  <param-name>cacheControl</param-name>
  <param-value>max-age=3600,public</param-value>
</init-param>
-->
<load-on-startup>0</load-on-startup>
</servlet>

<servlet-mapping> <servlet-name>default</servlet-name> <url-pattern>/</url-pattern> </servlet-
mapping>

<!-- ===== -->
<!-- JSP Servlet -->
<!-- This is the jasper JSP servlet from the jakarta project -->
<!-- ----- -->
<!-- The JSP page compiler and execution servlet, which is the mechanism -->
<!-- used by Glassfish to support JSP pages. Traditionally, this servlet -->
<!-- is mapped to URL patterh "*.jsp". This servlet supports the -->
<!-- following initialization parameters (default values are in square -->
<!-- brackets): -->
<!-- ----- -->
<!-- checkInterval If development is false and reloading is true, -->
<!-- background compiles are enabled. checkInterval -->
<!-- is the time in seconds between checks to see -->
<!-- if a JSP page needs to be recompiled. [300] -->
<!-- -->
<!-- compiler Which compiler Ant should use to compile JSP -->
<!-- pages. See the Ant documentation for more -->

```

```

<!--                                     information. [javac]                                     -->
<!--                                     -->
<!-- classdebuginfo                     Should the class file be compiled with -->
<!--                                     debugging information? [true]                -->
<!--                                     -->
<!-- classpath                           What class path should I use while compiling -->
<!--                                     generated servlets? [Created dynamically    -->
<!--                                     based on the current web application]        -->
<!--                                     Set to ? to make the container explicitly set -->
<!--                                     this parameter.                               -->
<!--                                     -->
<!-- development                         Is Jasper used in development mode (will check -->
<!--                                     for JSP modification on every access)? [true] -->
<!--                                     -->
<!-- enablePooling                       Determines whether tag handler pooling is      -->
<!--                                     enabled [true]                               -->
<!--                                     -->
<!-- fork                                 Tell Ant to fork compiles of JSP pages so that -->
<!--                                     a separate JVM is used for JSP page compiles -->
<!--                                     from the one Tomcat is running in. [true]    -->
<!--                                     -->
<!-- ieClassId                           The class-id value to be sent to Internet   -->
<!--                                     Explorer when using <jsp:plugin> tags.      -->
<!--                                     [clsid:8AD9C840-044E-11D1-B3E9-00805F499D93]    -->
<!--                                     -->
<!-- javaEncoding                       Java file encoding to use for generating java -->
<!--                                     source files. [UTF-8]                          -->
<!--                                     -->
<!-- keepgenerated                       Should we keep the generated Java source code -->
<!--                                     for each page instead of deleting it? [true]   -->
<!--                                     -->
<!-- logVerbosityLevel                  The level of detailed messages to be produced -->
<!--                                     by this servlet. Increasing levels cause the  -->
<!--                                     generation of more messages. Valid values are -->
<!--                                     FATAL, ERROR, WARNING, INFORMATION, and DEBUG. -->
<!--                                     [WARNING]                                       -->
<!--                                     -->
<!-- mappedfile                          Should we generate static content with one   -->
<!--                                     print statement per input line, to ease      -->
<!--                                     debugging? [false]                             -->
<!--                                     -->
<!-- reloading                           Should Jasper check for modified JSPs? [true] -->
<!--                                     -->
<!-- suppressSmap                        Should the generation of SMAP info for JSR45 -->
<!--                                     debugging be suppressed? [false]             -->
<!--                                     -->
<!-- dumpSmap                            Should the SMAP info for JSR45 debugging be   -->
<!--                                     dumped to a file? [false]                     -->
<!--                                     False if suppressSmap is true                 -->
<!--                                     -->
<!-- scratchdir                          What scratch directory should we use when  -->
<!--                                     compiling JSP pages? [default work directory  -->
<!--                                     for the current web application]              -->
<!--                                     -->
<!-- tagpoolMaxSize                     The maximum tag handler pool size [5]      -->
<!--                                     -->
<!-- xpoweredBy                          Determines whether X-Powered-By response     -->
<!--                                     header is added by generated servlet [false]  -->
<!--                                     -->
<!-- If you wish to use Jikes to compile JSP pages: -->
<!-- Set the init parameter "compiler" to "jikes". Define -->
<!-- the property "-Dbuild.compiler.emacs=true" when starting Jetty -->
<!-- to cause Jikes to emit error messages in a format compatible with -->
<!-- Jasper. -->
<!-- If you get an error reporting that jikes can't use UTF-8 encoding, -->
<!-- try setting the init parameter "javaEncoding" to "ISO-8859-1". -->
<!--                                     -->
<!--                                     ----- -->
<servlet id="jsp">
  <servlet-name>jsp</servlet-name>
  <servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class>
  <init-param>
    <param-name>logVerbosityLevel</param-name>
    <param-value>DEBUG</param-value>
  </init-param>
  <init-param>
    <param-name>fork</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param>
    <param-name>xpoweredBy</param-name>
    <param-value>>false</param-value>
  </init-param>
  <!--
  <init-param>
    <param-name>classpath</param-name>
    <param-value>?</param-value>
  </init-param>
  -->
  <load-on-startup>0</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>jsp</servlet-name>
  <url-pattern>*.jsp</url-pattern>
  <url-pattern>*.jspx</url-pattern>
  <url-pattern>*.jspf</url-pattern>
  <url-pattern>*.jsp</url-pattern>
  <url-pattern>*.jsp</url-pattern>
</servlet-mapping>

```

```

    <url-pattern>*.JSP</url-pattern>
    <url-pattern>*.JSPF</url-pattern>
    <url-pattern>*.JSPX</url-pattern>
    <url-pattern>*.XSP</url-pattern>
</servlet-mapping>

<!-- ===== -->
<!-- Dynamic Servlet Invoker. -->
<!-- This servlet invokes anonymous servlets that have not been defined -->
<!-- in the web.xml or by other means. The first element of the pathInfo -->
<!-- of a request passed to the invoker is treated as a servlet name for -->
<!-- an existing servlet, or as a class name of a new servlet. -->
<!-- This servlet is normally mapped to /servlet/* -->
<!-- This servlet support the following initParams: -->
<!-- ----- -->
<!-- nonContextServlets      If false, the invoker can only load -->
<!--                          servlets from the contexts classloader. -->
<!--                          This is false by default and setting this -->
<!--                          to true may have security implications. -->
<!-- ----- -->
<!-- verbose                  If true, log dynamic loads -->
<!-- ----- -->
<!-- *                        All other parameters are copied to the -->
<!--                          each dynamic servlet as init parameters -->
<!-- ----- -->
<!-- Uncomment for dynamic invocation
<servlet>
  <servlet-name>invoker</servlet-name>
  <servlet-class>org.mortbay.jetty.servlet.Invoker</servlet-class>
  <init-param>
    <param-name>verbose</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param>
    <param-name>nonContextServlets</param-name>
    <param-value>>false</param-value>
  </init-param>
  <init-param>
    <param-name>dynamicParam</param-name>
    <param-value>anyValue</param-value>
  </init-param>
  <load-on-startup>0</load-on-startup>
</servlet>

<servlet-mapping> <servlet-name>invoker</servlet-name> <url-pattern>/servlet/*</url-pattern>
</servlet-mapping>
-->

<!-- ===== -->
<session-config>
  <session-timeout>30</session-timeout>
</session-config>

<!-- ===== -->
<!-- Default MIME mappings -->
<!-- The default MIME mappings are provided by the mime.properties -->
<!-- resource in the org.mortbay.jetty.jar file. Additional or modified -->
<!-- mappings may be specified here -->
<!-- ----- -->
<!-- UNCOMMENT TO ACTIVATE
<mime-mapping>
  <extension>mysuffix</extension>
  <mime-type>mymime/type</mime-type>
</mime-mapping>
-->

<!-- ===== -->
<welcome-file-list>
  <welcome-file>index.html</welcome-file>
  <welcome-file>index.htm</welcome-file>
  <welcome-file>index.jsp</welcome-file>
</welcome-file-list>

<!-- ===== -->
<locale-encoding-mapping-list>
  <locale-encoding-mapping><locale>ar</locale><encoding>ISO-8859-6</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>be</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>bg</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>ca</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>cs</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>da</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>de</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>el</locale><encoding>ISO-8859-7</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>en</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
  <locale-encoding-mapping><locale>es</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>

```

```

    <locale-encoding-mapping><locale>et</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>fi</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>fr</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>hr</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>hu</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>is</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>it</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>iw</locale><encoding>ISO-8859-8</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>ja</locale><encoding>Shift_JIS</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>ko</locale><encoding>EUC-KR</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>lt</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>lv</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>mk</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>nl</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>no</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>pl</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>pt</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>ro</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>ru</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>sh</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>sk</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>sl</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>sq</locale><encoding>ISO-8859-2</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>sr</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>sv</locale><encoding>ISO-8859-1</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>tr</locale><encoding>ISO-8859-9</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>uk</locale><encoding>ISO-8859-5</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>zh</locale><encoding>GB2312</encoding></locale-encoding-
mapping>
    <locale-encoding-mapping><locale>zh_TW</locale><encoding>Big5</encoding></locale-encoding-
mapping>
  </locale-encoding-mapping-list>

  <security-constraint>
    <web-resource-collection>
      <web-resource-name>Disable TRACE</web-resource-name>
      <url-pattern>/</url-pattern>
      <http-method>TRACE</http-method>
    </web-resource-collection>
    <auth-constraint/>
  </security-constraint>
</web-app>

```

src/main/resources/META-INF/spring/application-properties-bean.xml

```

<bean id="applicationPropertiesBean"
class="org.springframework.beans.factory.config.PropertiesFactoryBean">
  <property name="locations">
    <list>
      <!-- El orden establece la prioridad -->
      <value>classpath*:META-INF/spring/*.properties</value>
      <value>classpath:META-INF/spring/*.properties</value>
      <value>classpath:META-INF/*.properties</value>
    </list>
  </property>
  <property name="ignoreResourceNotFound" value="true"/>
</bean>

```

src/main/resources/META-INF/spring/application.properties

```

# Properties server URL settings for remote access
REMOTE.HOST=http://localhost:8080
APP.NAME=test-web

```

```
# External I18N resources
home.i18n.resources=WEB-INF/i18n/messages
```

src/main/resources/META-INF/spring/applicationContext-security.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xmlns:p="http://www.springframework.org/schema/p"
       xmlns:tx="http://www.springframework.org/schema/tx"
       xmlns:security="http://www.springframework.org/schema/security"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-3.1.xsd
http://www.springframework.org/schema/security
http://www.springframework.org/schema/security/spring-security-3.1.xsd">

    <!-- HTTP security configurations-->
    <security:http entry-point-ref="casEntryPoint"
        access-denied-page="/casFailure" use-expressions="true">

        <security:logout invalidate-session="true" logout-url="/j_spring_security_logout"
logout-success-url="logoutUrl?url=http://hostname:portNumber/appName" />

        <!-- Configure these elements to secure URIs in your application -->
        <security:intercept-url pattern="/resources/**" access="permitAll" />
        <security:intercept-url pattern="/static/**" access="permitAll" />
        <security:intercept-url pattern="/remoting/**" access="isAuthenticated()" />
        <security:intercept-url pattern="/**" access="hasRole('$
{application.security.role}')" />
        <security:custom-filter ref="casFilter" after="CAS_FILTER" />
        <security:custom-filter ref="singleLogoutFilter" before="CAS_FILTER"/>
        <security:http-basic />

        <security:custom-filter position="CAS_FILTER" ref="casFilter" />
    </security:http>

    <security:authentication-manager alias="authenticationManager">
        <security:authentication-provider ref="casAuthenticationProvider">
            </security:authentication-provider>
        </security:authentication-manager>

        <bean id="singleLogoutFilter" class="org.jasig.cas.client.session.SingleSignOutFilter"/>

        <bean id="serviceProperties"
            class="org.springframework.security.cas.ServiceProperties">
            <property name="service"
                value="ServiceUrl"/>
            <property name="sendRenew" value="false"/>
        </bean>

        <bean id="casFilter"
            class="org.springframework.security.cas.web.CasAuthenticationFilter">
            <property name="authenticationManager" ref="authenticationManager"/>
        </bean>

        <bean id="casEntryPoint"
            class="org.springframework.security.cas.web.CasAuthenticationEntryPoint">
            <property name="loginUrl" value="LoginUrl"/>
            <property name="serviceProperties" ref="serviceProperties"/>
        </bean>

        <bean id="casAuthenticationProvider"
            class="org.springframework.security.cas.authentication.CasAuthenticationProvider">
            <property name="userDetailsService" ref="userService"/>
            <property name="serviceProperties" ref="serviceProperties" />
            <property name="ticketValidator">
                <bean class="org.jasig.cas.client.validation.Cas20ServiceTicketValidator">
                    <constructor-arg index="0" value="ticketValidatorUrl" />
                </bean>
            </property>
            <property name="key" value="idAuthProvider"/>
        </bean>

        <security:user-service id="userService">
    </security:user-service>
</beans>
```

src/main/resources/META-INF/spring/database.properties

```
#Datasource JNDI
jndi.datasource=java:comp/env/jdbc/testDS

#Persistence Unit
database.persistence.unit=persistenceUnitTest

#Hibernate properties
database.hibernate.default_schema=TEST
database.hibernate.dialect=HIBERNATE_DIALECT_GOES_HERE
database.hibernate.hbm2ddl.auto=validate
database.hibernate.ejb.naming_strategy=org.hibernate.cfg.ImprovedNamingStrategy
database.hibernate.connection.charset=UTF-8
database.hibernate.jdbc.fetch_size=100
```

```
database.hibernate.jdbc.batch_size=100
```

src/main/resources/META-INF/spring/entity-manager-factory.xml

```
<bean class="org.springframework.orm.jpa.LocalContainerEntityManagerFactoryBean"
id="entityManagerFactory">
    <property name="persistenceXmlLocation" value="classpath:META-INF/persistenceInfo.xml"/>
    <property name="persistenceUnitName" value="${database.persistence.unit}"/>
    <property name="dataSource" ref="dataSource"/>
    <property name="jpaPropertyMap">
        <map>
            <entry key="hibernate.default_schema" value="${
{database.hibernate.default_schema}"/>
            <entry key="hibernate.dialect" value="${database.hibernate.dialect}"/>
            <entry key="hibernate.hbm2ddl.auto" value="${
{database.hibernate.hbm2ddl.auto}"/>
            <entry key="hibernate.ejb.naming_strategy" value="${
{database.hibernate.ejb.naming_strategy}"/>
            <entry key="hibernate.connection.charset" value="${
{database.hibernate.connection.charset}"/>
            <entry key="hibernate.jdbc.fetch_size" value="${
{database.hibernate.jdbc.fetch_size}"/>
            <entry key="hibernate.jdbc.batch_size" value="${
{database.hibernate.jdbc.batch_size}"/>
        </map>
    </property>
</bean>
```

src/main/resources/META-INF/spring/jndi-datasource.xml

```
<jee:jndi-lookup id="dataSource" jndi-name="${jndi.datasource}"/>
```

src/main/resources/META-INF/spring/message-source.xml

```
<bean class="org.springframework.context.support.ReloadableResourceBundleMessageSource"
id="messageSource" p:fallbackToSystemLocale="false" p:basenames="${home.il18n.resources},WEB-
INF/il18n/application"/>
```

src/main/resources/META-INF/spring/property-placeholder-configurer.xml

```
<context:property-placeholder properties-ref="applicationPropertiesBean" order="1" ignore-resource-
not-found="true" ignore-unresolvable="true"/>
```

src/main/tomcat/context.xml

```
<?xml version='1.0' encoding='utf-8'?>
<Context>
    <Resource name="jdbc/testDS"
        auth="Container"
        type="javax.sql.DataSource"
        username="user"
        password="password"
        driverClassName="databaseDriver"
        url="databaseUrl"
        maxActive="3" maxIdle="1"/>
</Context>
```

src/main/tomcat/server.xml

```
<Server port="8010" shutdown="SHUTDOWN">
    <GlobalNamingResources>
        <Resource name="UserDatabase" auth="Container"
            type="org.apache.catalina.UserDatabase"
            description="User database that can be updated and saved"
            factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
            pathname="conf/tomcat-users.xml" />
    </GlobalNamingResources>
    <Service name="Catalina">
        <Connector port="8080" URIEncoding="UTF-8" keepAliveTimeout="1800000"
maxKeepAliveRequests="30000" maxThreads="300" />
        <Connector port="8088" keepAliveTimeout="1800000" maxKeepAliveRequests="30000"
maxThreads="300" />
        <Engine name="Catalina" defaultHost="localhost">
            <Valve className="org.apache.catalina.valves.AccessLogValve"
                resolveHosts="false" pattern="%t-ip:%a-protocol:%H-localPort:%p-path:%U-time:%D ms"/>
            <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
                resourceName="UserDatabase" />
            <Host name="localhost" appBase="webapps" />
        </Engine>
    </Service>
</Server>
```

src/main/webapp/WEB-INF/casfailure-view.xml

```
<definition name="casFailure" extends="public">
  <put-attribute name="body" value="/WEB-INF/views/casfailure.jspx" />
</definition>
```

src/main/webapp/WEB-INF/views/casfailure.jspx

```
<div xmlns:c="http://java.sun.com/jsp/jstl/core"
      xmlns:util="urn:jsptagdir:/WEB-INF/tags/util"
      xmlns:fn="http://java.sun.com/jsp/jstl/functions"
      xmlns:spring="http://www.springframework.org/tags"
      xmlns:jsp="http://java.sun.com/JSP/Page"
      version="2.0">

  <jsp:directive.page
import="org.springframework.web.servlet.handler.SimpleMappingExceptionResolver"/>
  <jsp:directive.page import="org.springframework.security.access.AccessDeniedException"/>
  <jsp:directive.page import="org.springframework.security.core.AuthenticationException"/>
  <jsp:directive.page import="org.springframework.security.web.WebAttributes"/>
  <jsp:directive.page contentType="text/html; charset=UTF-8" />
  <jsp:output omit-xml-declaration="yes" />

    <div id="casfailure_body">
      <div id="cas_errors">
        <span class="errors">
          <spring:message code="label.login.error" text="Error" />
          <jsp:scriptlet>
            AuthenticationException authException =
(AuthenticationException) session.getAttribute(WebAttributes.AUTHENTICATION_EXCEPTION);
pageContext.setAttribute("session_exception_code",
authException != null ? authException.getMessage() : "", PageContext.PAGE_SCOPE);
pageContext.setAttribute("session_exception_message",
authException != null ? authException.getLocalizedMessage() : "", PageContext.PAGE_SCOPE);
          </jsp:scriptlet>
          <c:if test="{not empty session_exception_code}">
            <spring:message code="{session_exception_code}" text="{session_exception_message}" />
          </c:if>
        </span>
        <br/>
        <span class="errors">
          <spring:message code="label.login.error" text="Error" />
          <jsp:scriptlet>
            AccessDeniedException accessException =
(AccessDeniedException) request.getAttribute(WebAttributes.ACCESS_DENIED_403);
            if (accessException == null) {
              accessException=(AccessDeniedException) request.getAttribute(SimpleMappingExceptionResolver.DEFAULT_EXCEPTION_ATTRIBUTE);
            }
            pageContext.setAttribute("request_exception_code",
accessException != null ? accessException.getMessage() : "", PageContext.PAGE_SCOPE);
            pageContext.setAttribute("request_exception_message",
accessException != null ? accessException.getLocalizedMessage() : "", PageContext.PAGE_SCOPE);
          </jsp:scriptlet>
          <c:if test="{not empty request_exception_code}">
            <spring:message code="{request_exception_code}" text="{request_exception_message}" />
          </c:if>
        </span>
      </div>
    </div>
  </div>
</div>
```

ANEXO III: Contenido del directorio de configuración

A continuación se muestra el contenido por defecto de cada uno de los ficheros de texto contenidos en el directorio de configuración de Roo++, que se encuentra disponible dentro del directorio *resources/RooAddonsData* del JAR de este software.

applicationContext-platfowm.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE beans PUBLIC "-//SPRING//DTD BEAN//EN" "http://www.springframework.org/dtd/spring-beans.dtd">

<beans>

  <bean id="fileEventType" class="com.devdaily.springtest1.bean.FileEventType">
    <property name="eventType" value="10"/>
    <property name="description" value="A sample description here"/>
  </bean>

  <bean id="fileEventDao" class="com.devdaily.springtest1.dao.FileEventDao">
    <property name="dataSource" ref="basicDataSource"/>
  </bean>

</beans>
```

```

<bean id="basicDataSource" class="org.apache.commons.dbcp.BasicDataSource">
  <property name="driverClassName" value="com.mysql.jdbc.Driver" />
  <property name="url" value="jdbc:mysql://localhost/my_database" />
  <property name="username" value="my_username" />
  <property name="password" value="my_password" />
  <property name="initialSize" value="3" />
  <property name="maxActive" value="10" />
</bean>
</beans>

```

applicationContext-portal.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE beans PUBLIC "-//SPRING/DTD BEAN//EN" "http://www.springframework.org/dtd/spring-
beans.dtd">

```

```

<beans>

  <bean id="fileEventType" class="com.devdaily.springtest1.bean.FileEventType">
    <property name="eventType" value="10"/>
    <property name="description" value="A sample description here"/>
  </bean>

  <bean id="fileEventDao" class="com.devdaily.springtest1.dao.FileEventDao">
    <property name="dataSource" ref="basicDataSource"/>
  </bean>

  <bean id="basicDataSource" class="org.apache.commons.dbcp.BasicDataSource">
    <property name="driverClassName" value="com.mysql.jdbc.Driver" />
    <property name="url" value="jdbc:mysql://localhost/my_database" />
    <property name="username" value="my_username" />
    <property name="password" value="my_password" />
    <property name="initialSize" value="3" />
    <property name="maxActive" value="10" />
  </bean>
</beans>

```

breadcrumb-dependency.xml

```

<dependency>
  <groupId>com.empresa.platform</groupId>
  <artifactId>platform-readcrumb-module</artifactId>
  <version>${platform.version}</version>
</dependency>

```

breadcrumbs-beans.xml

```

<!-- Bean para las migas de pan -->
<bean id="breadcrumb" class="com.breadcrumbs.Breadcrumbs"
  init-method="configureControls" scope="singleton">
</bean>

```

breadcrumbs-url.txt

<http://www.empresa.com/breadcrumbs>

context-dataload.sql

```

INSERT INTO CONTEXT_ENTITY VALUES(1,1,'contexto por defecto','default', 1);
COMMIT;

```

context-dependencies.txt

```

com.empresa.usermgmt
usermgmt-contexts
1.0

```

context-messages.properties

```

# Modulo de configuracion de contextos
portal.configuration.context.module.name=Contexts
portal.configuration.context.module.description=Contexts

```

context-portal.properties

```

portal.configuration.context.module.id=Contexts

```



```
portal.configuration.context.module.name=Contexts
portal.configuration.context.module.description=Contexts
portal.configuration.context.module.url=${portal.application.url}/contexts
portal.configuration.context.module.role=ROLE_ADMIN
```

licenses.xml

```
<licenses>
  <license>
    <name>Software license</name>
    <url>${license.url}softwarelicense/license.txt</url>
    <distribution>repo</distribution>
    <comments>This is a software license</comments>
  </license>
</licenses>
```

locations-application-properties.xml

messages.properties

```
application_name=Aplicación

#Informacion de los modulos de la aplicacion en el menu general
application.name=Nombre de la Aplicación
application.description=Descripción de la aplicación

# Modulo de configuracion de la aplicacion
application.configuration.label.name=Administración
application.configuration.label.description=Menú de administración

#Casfailed
label.authorization.failure.title=Error de autenticación
label.authorization.failure.access=Error de acceso
label.global.volver=Volver
label.global.salir=Salir
label.global.footer=Texto del footer

portal.configuration.default.module.name=Administrar
portal.configuration.default.module.description=Módulo de administración
```

mssql.txt

```
1.2.4
SQLServer2008Dialect
```

mysql.txt

```
5.1.18
MySQL5InnoDBDialect
```

oracle.txt

```
16
Oracle10gDialect
```

oracle-sid.txt

```
XE
```

parent.txt

```
groupId
artifactId
version
```

platform.txt

```
groupId
version
module-name
```

plugin-license.xml

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
```

```

<artifactId>license-maven-plugin</artifactId>
<version>1.0</version>
<configuration>
<licenseName>license</licenseName>
<licenseResolver>${license.url}</licenseResolver>
<bundleLicensePath>META-INF/LICENSE.txt</bundleLicensePath>
<generateBundle>>false</generateBundle>
</configuration>
<executions>
<execution>
<id>update-project-license</id>
<goals>
<goal>update-project-license</goal>
</goals>
<phase>process-sources</phase>
</execution>
</executions>
</plugin>

```

portal-header.jspx

```

<div id="header" xmlns:jsp="http://java.sun.com/JSP/Page"
xmlns:fn="http://java.sun.com/jsp/jstl/functions"
xmlns:c="http://java.sun.com/jsp/jstl/core"
xmlns:spring="http://www.springframework.org/tags"
xmlns:portal="http://www.empresa.com/portal" version="2.0">

<jsp:directive.page import="org.springframework.web.servlet.support.RequestContextUtils" />
<jsp:directive.page import="org.springframework.context.i18n.LocaleContextHolder" />
<jsp:directive.page import="com.empresa.portal.handler.PortalClientHandler" />

<jsp:scriptlet>
pageContext.setAttribute("applications", PortalClientHandler.getApplications(),
PageContext.PAGE_SCOPE);
pageContext.setAttribute("configuration", PortalClientHandler.getConfiguration(),
PageContext.PAGE_SCOPE);
pageContext.setAttribute("currentLocale", LocaleContextHolder.getLocale(),
PageContext.PAGE_SCOPE);
pageContext.setAttribute("requestUri", org.springframework.security.web.util.UrlUtils
.buildFullRequestUrl(pageContext.getRequest().getScheme(),
pageContext.getRequest().getServerName(),
pageContext.getRequest().getServerPort(),
pageContext.getAttribute("javax.servlet.forward.request_uri", 2).toString(),
null), 2);
</jsp:scriptlet>

<portal:portal-header applications="${applications}" currentLocale="${currentLocale}"
configuration="${configuration}" requestUri="${requestUri}" />

```

portal-url.txt

http://www.empresa.com/portal

portal.properties

```

# Beans que proporcionan la información de la aplicación
portal.application.provider.name=portalApplicationProvider
portal.configuration.provider.name=portalConfigurationProvider

# Properties de configuración del portal
portal.application.id=${APP.NAME}
portal.application.url=${REMOTE.HOST}/${APP.NAME}
portal.application.priority=1
portal.application.default.name=${APP.NAME}
portal.application.default.description=${APP.NAME}

portal.default.module.id=Default
portal.default.module.url=${portal.application.url}/index

portal.configuration.logout.url=/j_spring_security_logout
portal.configuration.user.url=${portal.application.url}/userInfo
portal.configuration.security.role=ROLE_ADMIN

portal.configuration.default.module.id=Configuration
portal.configuration.default.module.url=${portal.application.url}/index
portal.configuration.default.module.role=ROLE_ADMIN

portal.default.module.default.name=Default
portal.default.module.default.description=Default

```

portal.txt

```

com.empresa.portal
portal-client
1.0

```

postgresql.txt

```
9.0-801.jdbc4
PostgreSQLDialect
```

usermgmt.txt

```
localhost
8080
http://localhost:9443/cas/login
http://localhost:9443/cas
an id for this auth provider only
http://localhost:9443/cas/logout
```

usermgmt_application.properties

```
# Role security
application.security.role=ROLE_ADMIN
```

usermgmt_users.txt

```
joe
joe
ROLE_USER
mary
mary
ROLE_USER, ROLE_ADMIN
```

contextBeans/portal/nodesToAdd/nodeToAdd.txt

```
/
beans/bean[@id='portalConfigurationProvider']/property[@name='configuration']/bean[@class='com.organ
izacion.portal.Configuration']/property[@name='configurationModules']/list
ref
bean
contextConfigurationModule
```

contextBeans/portal/portal_bean.xml

```
<bean id="contextConfigurationModule" class="com.organization.portal.ConfigurationModule">
    <property name="id" value="{portal.configuration.context.module.id}" />
    <property name="name" value="portal.configuration.context.module.name" />
    <property name="description"
value="portal.configuration.context.module.description" />
    <property name="accessAuthority" value="$
{portal.configuration.context.module.role}" />
    <property name="url" value="{portal.configuration.context.module.url}" />
</bean>
```

contextBeans/security/authenticationSuccessHandler.txt

```
contextAuthenticationSuccessHandler
```

contextBeans/security/security_bean_1.xml

```
<!-- Bean LogoutSuccessHandler -->
<bean id="contextLogoutSuccessHandler"
class="com.organization.usermgmt.context.ContextLogoutSuccessHandler">
    <property name="defaultTargetUrl"
value="{cas.securityContext.casAuthenticationFilterEntryPoint.logoutUrl}" />
    <property name="alwaysUseDefaultTargetUrl" value="true" />
</bean>
```

contextBeans/security/security_bean_2.xml

```
<!-- Bean AuthenticationSuccessHandler -->
<bean id="contextAuthenticationSuccessHandler"
class="com.organizacion.usermgmt.context.ContextAuthenticationSuccessHandler">
    <property name="contextHandler" ref="contextHandler" />
    <property name="defaultTargetUrl" value="/" />
    <property name="accessDeniedHandler">
        <bean id="accessDeniedHandler"
class="org.springframework.security.web.access.AccessDeniedHandlerImpl">
            <property name="errorPage" value="/casFailure" />
        </bean>
    </property>
</bean>
```

contextHeader/add-to-portal-header.txt

```
contextName="${contextName}"
```

contextHeader/directive-pages.jspx

```
<jsp:directive.page import="com.organizacion.context.handler.ContextSessionHandler" />  
<jsp:directive.page import="com.organizacion.platform.context.vo.ContextVO" />
```

contextHeader/scriptlets.jspx

```
<jsp:scriptlet>ContextVO context = ContextSessionHandler.retrieveContext(request);  
    String contextName = (context == null) ? "" : context.getName();  
    pageContext.setAttribute("contextName", contextName,  
PageContext.PAGE_SCOPE);</jsp:scriptlet>
```

contextViews/change.jspx

```
<div id="contextChange" class="body_info" xmlns:jsp="http://java.sun.com/JSP/Page"  
    xmlns:fn="http://java.sun.com/jsp/jstl/functions"  
    xmlns:c="http://java.sun.com/jsp/jstl/core"  
    xmlns:spring="http://www.springframework.org/tags"  
    xmlns:context="http://www.empresa.com/platform/contexts/change">  
<jsp:directive.page contentType="text/html; charset=UTF-8" />  
<jsp:output omit-xml-declaration="yes" />  
  
    <fieldset>  
        <legend>  
            <spring:message  
                code="label_contextentity"  
                htmlEscape="false" var="label" />  
            <spring:message arguments="{label}" code="entity_change"  
                htmlEscape="false" />  
        </legend>  
        <context:change showChangeDefaultContext="true"  
            showChangeSessionContext="true" />  
    </fieldset>  
</div>
```

contextViews/create.jspx

```
<div id="contextCreate" class="body_info" xmlns:jsp="http://java.sun.com/JSP/Page"  
    xmlns:fn="http://java.sun.com/jsp/jstl/functions"  
    xmlns:c="http://java.sun.com/jsp/jstl/core"  
    xmlns:spring="http://www.springframework.org/tags"  
    xmlns:context="http://www.empresa.com/platform/contexts"  
    version="2.0">  
<jsp:directive.page contentType="text/html; charset=UTF-8" />  
<jsp:output omit-xml-declaration="yes" />  
  
    <fieldset>  
        <legend>  
            <spring:message  
                code="label_contextentity"  
                htmlEscape="false" var="label" />  
            <spring:message arguments="{label}" code="entity_create"  
                htmlEscape="false" />  
        </legend>  
        <context:create />  
    </fieldset>  
</div>
```

contextViews/list.jspx

```
<div class="body_info" id="contextList" xmlns:jsp="http://java.sun.com/JSP/Page"  
    xmlns:fn="http://java.sun.com/jsp/jstl/functions"  
    xmlns:c="http://java.sun.com/jsp/jstl/core"  
    xmlns:spring="http://www.springframework.org/tags"  
    xmlns:context="http://www.empresa.com/platform/contexts"  
    version="2.0">  
<jsp:directive.page contentType="text/html; charset=UTF-8" />  
<jsp:output omit-xml-declaration="yes" />  
  
    <context:list />  
</div>
```

contextViews/show.jspx

```
<div id="contextShow" class="body_info" xmlns:jsp="http://java.sun.com/JSP/Page"
```

```

xmlns:fn="http://java.sun.com/jsp/jstl/functions"
xmlns:c="http://java.sun.com/jsp/jstl/core"
xmlns:spring="http://www.springframework.org/tags"
xmlns:context="http://www.empresa.com/platform/contexts"
version="2.0">
<jsp:directive.page contentType="text/html;charset=UTF-8" />
<jsp:output omit-xml-declaration="yes" />

    <fieldset>
        <legend>
            <spring:message code="legend_basic_data" />
        </legend>
        <context:show />
    </fieldset>

</div>

```

contextViews/update.jspx

```

<div id="contextUpdate" class="body_info" xmlns:jsp="http://java.sun.com/JSP/Page"
xmlns:fn="http://java.sun.com/jsp/jstl/functions"
xmlns:c="http://java.sun.com/jsp/jstl/core"
xmlns:spring="http://www.springframework.org/tags"
xmlns:context="http://www.empresa.com/platform/contexts"
version="2.0">
<jsp:directive.page contentType="text/html;charset=UTF-8" />
<jsp:output omit-xml-declaration="yes" />

    <fieldset>
        <legend>
            <spring:message
                code="label_contextentity"
                htmlEscape="false" var="label" />
            <spring:message arguments="{label}" code="entity_update"
                htmlEscape="false" />
        </legend>
        <context:update />
    </fieldset>

</div>

```

doc/docbkx/custom.xsl

```

<xsl:stylesheet xmlns:t="http://nwalsh.com/docbook/xsl/template/1.0"
xmlns:param="http://nwalsh.com/docbook/xsl/template/1.0/param"
xmlns:fo="http://www.w3.org/1999/XSL/Format"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">

    <xsl:import href="urn:docbkx:stylesheet" />
    <!--<xsl:import href="cover.xsl" />-->

    <!-- Configuracion global -->
    <xsl:param name="hyphenate">false</xsl:param>
    <xsl:param name="headers.on.blank.pages">0</xsl:param>
    <xsl:param name="footers.on.blank.pages">0</xsl:param>

    <xsl:param name="l10n.gentext.default.language">es</xsl:param>

    <!-- Configuracion de las paginas -->
    <xsl:param name="page.margin.top">10mm</xsl:param>
    <xsl:param name="page.margin.right">25mm</xsl:param>
    <xsl:param name="page.margin.bottom">20mm</xsl:param>
    <xsl:param name="page.margin.left">25mm</xsl:param>
    <xsl:param name="region.before.extent">10mm</xsl:param>
    <xsl:param name="region.after.extent">10mm</xsl:param>
    <xsl:param name="body.start.indent">15mm</xsl:param>
    <xsl:param name="body.margin.top">15mm</xsl:param>
    <xsl:param name="body.margin.bottom">10mm</xsl:param>

    <!-- Tipografia -->
    <xsl:param name="title.font.family">Georgia</xsl:param>
    <xsl:param name="body.font.family">Calibri</xsl:param>
    <xsl:param name="dingbat.font.family">Calibri</xsl:param>
    <xsl:param name="body.font.master">10</xsl:param>
    <xsl:param name="alignment">justify</xsl:param>
    <xsl:param name="line-height">normal</xsl:param>

    <!-- Cabecera -->
    <xsl:param name="header.rule" select="0"></xsl:param>

    <xsl:attribute-set name="header.content.properties">
        <xsl:attribute name="font-family">Georgia</xsl:attribute>
        <xsl:attribute name="font-size">7pt</xsl:attribute>
    </xsl:attribute-set>

    <xsl:template name="header.content">
        <xsl:param name="pageclass" select="'" />
        <xsl:param name="sequence" select="'" />
        <xsl:param name="position" select="'" />
        <xsl:param name="gentext-key" select="'" />
        <fo:block>
            <xsl:choose>

```

```

        <xsl:when
            test="($pageclass='body' or $pageclass='front') and $position
= 'left'">
            <fo:block>
                <xsl:value-of
                    select="concat(/book/title,
',/book/bookinfo/releaseinfo)" />
            </fo:block>
            <fo:block>
                <xsl:value-of
                    select="ancestor-or-self::book/subtitle" />
            </fo:block>
        </xsl:when>
        <xsl:when
            test="$pageclass='body' and $position = 'right'">
            <fo:block>
                <xsl:apply-templates select="."
                    mode="titleabbrev.markup" />
            </fo:block>
        </xsl:when>
    </xsl:choose>
</fo:block>
</xsl:template>

<!-- Pie de pagina -->
<xsl:param name="footer.rule" select="0"></xsl:param>
<xsl:attribute-set name="footer.content.properties">
    <xsl:attribute name="font-family">Georgia</xsl:attribute>
    <xsl:attribute name="font-size">9pt</xsl:attribute>
</xsl:attribute-set>
<xsl:template name="footer.content">
    <xsl:param name="pageclass" select="'" />
    <xsl:param name="sequence" select="'" />
    <xsl:param name="position" select="'" />
    <xsl:param name="gentext-key" select="'" />
    <fo:block>
        <xsl:choose>
            <xsl:when test="$position = 'right'">
                <fo:page-number />
            </xsl:when>
        </xsl:choose>
    </fo:block>
</xsl:template>

<!-- Propiedades de los títulos de los capitulos -->
<xsl:attribute-set name="component.title.properties">
    <xsl:attribute name="font-size">26</xsl:attribute>
    <xsl:attribute name="font-family">Georgia</xsl:attribute>
    <xsl:attribute name="space-before.minimum">60pt</xsl:attribute>
    <xsl:attribute name="space-before.optimum">60pt</xsl:attribute>
    <xsl:attribute name="space-before.maximum">60pt</xsl:attribute>
    <xsl:attribute name="space-after.minimum">20pt</xsl:attribute>
    <xsl:attribute name="space-after.optimum">20pt</xsl:attribute>
    <xsl:attribute name="space-after.maximum">20pt</xsl:attribute>
    <xsl:attribute name="border-bottom">
        0.5pt solid black
    </xsl:attribute>
    <xsl:attribute name="padding-bottom">10pt</xsl:attribute>
    <xsl:attribute name="padding-top">150pt</xsl:attribute>
    <xsl:attribute name="text-align">end</xsl:attribute>
</xsl:attribute-set>

<!-- Quitar Capitulo1, Capitulo2 ... -->
<xsl:param name="chapter.autolabel" select="0"></xsl:param>

<!-- Secciones -->
<xsl:attribute-set name="section.title.properties">
    <xsl:attribute name="space-after.minimum">5mm</xsl:attribute>
    <xsl:attribute name="space-after.optimum">5mm</xsl:attribute>
    <xsl:attribute name="space-after.maximum">5mm</xsl:attribute>
</xsl:attribute-set>

<xsl:attribute-set name="section.title.level1.properties">
    <xsl:attribute name="font-size">16</xsl:attribute>
    <xsl:attribute name="border-bottom">
        0.5pt solid black
    </xsl:attribute>
    <xsl:attribute name="space-before.minimum">15mm</xsl:attribute>
    <xsl:attribute name="space-before.optimum">15mm</xsl:attribute>
    <xsl:attribute name="space-before.maximum">15mm</xsl:attribute>
</xsl:attribute-set>

<xsl:attribute-set name="section.title.level2.properties">
    <xsl:attribute name="font-size">14</xsl:attribute>
    <xsl:attribute name="border-bottom">
        0.5pt solid black
    </xsl:attribute>
    <xsl:attribute name="margin-left">15mm</xsl:attribute>
    <xsl:attribute name="space-before.minimum">10mm</xsl:attribute>
    <xsl:attribute name="space-before.optimum">10mm</xsl:attribute>
    <xsl:attribute name="space-before.maximum">10mm</xsl:attribute>
</xsl:attribute-set>

<xsl:attribute-set name="section.title.level3.properties">
    <xsl:attribute name="font-size">12</xsl:attribute>
    <xsl:attribute name="margin-left">15mm</xsl:attribute>
    <xsl:attribute name="space-before.minimum">10mm</xsl:attribute>

```

```

        <xsl:attribute name="space-before.optimum">10mm</xsl:attribute>
        <xsl:attribute name="space-before.maximum">10mm</xsl:attribute>
    </xsl:attribute-set>

<!-- Propiedades de los objetos (figuras) -->
<xsl:attribute-set name="formal.object.properties">
    <xsl:attribute name="space-before.minimum">10mm</xsl:attribute>
    <xsl:attribute name="space-before.optimum">10mm</xsl:attribute>
    <xsl:attribute name="space-before.maximum">10mm</xsl:attribute>
    <xsl:attribute name="space-after.minimum">10mm</xsl:attribute>
    <xsl:attribute name="space-after.optimum">10mm</xsl:attribute>
    <xsl:attribute name="space-after.maximum">10mm</xsl:attribute>
</xsl:attribute-set>

<!-- Titulos de los objetos -->
<xsl:attribute-set name="formal.title.properties">
    <xsl:attribute name="font-size">10pt</xsl:attribute>
    <xsl:attribute name="hyphenate">>false</xsl:attribute>
    <xsl:attribute name="font-style">
        <xsl:choose>
            <xsl:when test="self::figure">italic</xsl:when>
            <xsl:otherwise>inherit</xsl:otherwise>
        </xsl:choose>
    </xsl:attribute>
    <xsl:attribute name="font-weight">
        <xsl:choose>
            <xsl:when test="self::figure">normal</xsl:when>
            <xsl:otherwise>inherit</xsl:otherwise>
        </xsl:choose>
    </xsl:attribute>
</xsl:attribute-set>

<!-- Propiedades de las listas -->
<xsl:attribute-set name="list.item.spacing">
    <xsl:attribute name="space-before.optimum">2.5mm</xsl:attribute>
    <xsl:attribute name="space-before.minimum">2.5mm</xsl:attribute>
    <xsl:attribute name="space-before.maximum">2.5mm</xsl:attribute>
</xsl:attribute-set>
<xsl:attribute-set name="list.block.spacing">
    <xsl:attribute name="space-before.optimum">5mm</xsl:attribute>
    <xsl:attribute name="space-before.minimum">5mm</xsl:attribute>
    <xsl:attribute name="space-before.maximum">5mm</xsl:attribute>
    <xsl:attribute name="space-after.optimum">5mm</xsl:attribute>
    <xsl:attribute name="space-after.minimum">5mm</xsl:attribute>
    <xsl:attribute name="space-after.maximum">5mm</xsl:attribute>
</xsl:attribute-set>
<!-- Ancho entre el punto y el texto -->
<xsl:param name="itemizedlist.label.width">1.0em</xsl:param>

<!-- Placeholder templates -->
<xsl:template name="front.cover"></xsl:template>
<xsl:template name="back.cover"></xsl:template>

<!-- Niveles del índice a mostrar -->
<xsl:param name="toc.section.depth">4</xsl:param>

<!-- Para generar número de etiquetas -->
<xsl:param name="section.autolabel.max.depth">8</xsl:param>
<xsl:param name="section.autolabel" select="1"></xsl:param>
<xsl:param name="section.label.includes.component.label"
    select="0">
</xsl:param>

<!-- Quita el título de las notas (Nota)-->
<xsl:param name="admon.textlabel" select="0"></xsl:param>

<!-- Propiedades de las notas -->
<xsl:attribute-set name="admonition.properties">
    <xsl:attribute name="background-color">#E0E0E0</xsl:attribute>
    <xsl:attribute name="padding-top">15pt</xsl:attribute>
    <xsl:attribute name="padding-bottom">15pt</xsl:attribute>
    <xsl:attribute name="padding-left">15pt</xsl:attribute>
    <xsl:attribute name="padding-right">15pt</xsl:attribute>
    <xsl:attribute name="margin-top">0pt</xsl:attribute>
    <xsl:attribute name="margin-left">0pt</xsl:attribute>
    <xsl:attribute name="margin-right">0pt</xsl:attribute>
    <xsl:attribute name="margin-bottom">0pt</xsl:attribute>
</xsl:attribute-set>

<!-- Propiedades de los márgenes de las notas -->
<xsl:attribute-set name="nongraphical.admonition.properties">
    <xsl:attribute name="space-before.minimum">0.8em</xsl:attribute>
    <xsl:attribute name="space-before.optimum">1em</xsl:attribute>
    <xsl:attribute name="space-before.maximum">1.2em</xsl:attribute>
    <xsl:attribute name="space-after.minimum">0.8em</xsl:attribute>
    <xsl:attribute name="space-after.optimum">1em</xsl:attribute>
    <xsl:attribute name="space-after.maximum">1.2em</xsl:attribute>
    <xsl:attribute name="margin-left">0in</xsl:attribute>
    <xsl:attribute name="margin-right">0in</xsl:attribute>
</xsl:attribute-set>

<!-- Márgenes entre los índices -->
<xsl:attribute-set name="toc.margin.properties">
    <xsl:attribute name="space-before.minimum">2em</xsl:attribute>
    <xsl:attribute name="space-before.optimum">2em</xsl:attribute>

```

```

        <xsl:attribute name="space-before.maximum">2em</xsl:attribute>
        <xsl:attribute name="space-after.minimum">2em</xsl:attribute>
        <xsl:attribute name="space-after.optimum">2em</xsl:attribute>
        <xsl:attribute name="space-after.maximum">2em</xsl:attribute>
    </xsl:attribute-set>

    <!-- Título de la tabla de contenidos -->
    <xsl:template name="table.of.contents.titlepage" priority="1">
        <fo:block font-size="16pt" font-family="Georgia"
            space-before="0in" space-before.conditionality="retain"
            space-after="20pt" border-left="1pt solid black" padding-top="40pt"
            padding-left="15pt" padding-bottom="0pt">
            <xsl:call-template name="gentext">
                <xsl:with-param name="key" select="'TableofContents'" />
            </xsl:call-template>
        </fo:block>
    </xsl:template>

    <xsl:template name="list.of.figures.titlepage" priority="1">
        <fo:block font-size="16pt" font-family="Georgia"
            space-before="0in" space-before.conditionality="retain"
            space-after="20pt" border-left="1pt solid black" padding-top="40pt"
            padding-left="15pt" padding-bottom="0pt">
            <xsl:call-template name="gentext">
                <xsl:with-param name="key" select="'ListofFigures'" />
            </xsl:call-template>
        </fo:block>
    </xsl:template>

    <!-- INDICES -->

    <!-- Para quitar el índice y lista de figuras -->
    <!-- <xsl:param name="generate.toc" />-->

    <xsl:attribute-set name="toc.line.properties">
        <xsl:attribute name="font-family">
            <xsl:choose>
                <xsl:when test="self::section">Calibri</xsl:when>
                <xsl:otherwise>Georgia</xsl:otherwise>
            </xsl:choose>
        </xsl:attribute>
        <xsl:attribute name="padding-top">
            <xsl:choose>
                <xsl:when test="self::chapter">20pt</xsl:when>
                <xsl:otherwise>2pt</xsl:otherwise>
            </xsl:choose>
        </xsl:attribute>
        <xsl:attribute name="border-bottom">
            <xsl:choose>
                <xsl:when test="self::chapter">
                    0.5pt solid black
                </xsl:when>
                <xsl:otherwise>0pt solid black</xsl:otherwise>
            </xsl:choose>
        </xsl:attribute>
        <xsl:attribute name="font-size">10</xsl:attribute>
    </xsl:attribute-set>

    <!-- Separador del índice entre el título y la numeración -->
    <xsl:param name="autotoc.label.separator">.</xsl:param>
    <xsl:param name="generate.section.toc.level" select="0"></xsl:param>

    <!-- Genera las entradas del índice. Sólo está cambiado puntos por espacios -->
    <xsl:template name="toc.line">
        <xsl:param name="toc-context" select="NOTANODE" />

        <xsl:variable name="id">
            <xsl:call-template name="object.id" />
        </xsl:variable>

        <xsl:variable name="label">
            <xsl:apply-templates select="." mode="label.markup" />
        </xsl:variable>

        <fo:block xsl:use-attribute-sets="toc.line.properties">
            <fo:inline keep-with-next.within-line="always">
                <fo:basic-link internal-destination="{ $id }">
                    <xsl:if test="$label != ''">
                        <xsl:copy-of select="$label" />
                        <xsl:value-of select="$autotoc.label.separator" />
                    </xsl:if>
                    <xsl:apply-templates select="."
                        mode="titleabbrev.markup" />
                </fo:basic-link>
            </fo:inline>
            <fo:inline keep-together.within-line="always">
                <xsl:text></xsl:text>
                <fo:leader leader-pattern="space"
                    leader-pattern-width="0cm" leader-alignment="reference-area"
                    keep-with-next.within-line="always" />
                <xsl:text></xsl:text>
                <fo:basic-link internal-destination="{ $id }">
                    <fo:page-number-citation ref-id="{ $id }" />
                </fo:basic-link>
            </fo:inline>
        </fo:block>
    </xsl:template>
</xsl:stylesheet>

```


doc/fonts.xml

```
<font>
  <font>
    <name>DejaVuSans-Bold</name>
    <style>normal</style>
    <weight>normal</weight>
    <embedFile>${basedir}/src/fonts/DejaVuSans-Bold.ttf</embedFile>
    <metricsFile>${project.build.directory}/fonts/DejaVuSans-Bold-
metrics.xml</metricsFile>
  </font>
  <font>
    <name>DejaVuSans</name>
    <style>normal</style>
    <weight>normal</weight>
    <embedFile>${basedir}/src/fonts/DejaVuSans.ttf</embedFile>
    <metricsFile>${project.build.directory}/fonts/DejaVuSans-metrics.xml</metricsFile>
  </font>
  <font>
    <name>DejaVuSerif</name>
    <style>normal</style>
    <weight>normal</weight>
    <embedFile>${basedir}/src/fonts/DejaVuSerif.ttf</embedFile>
    <metricsFile>${project.build.directory}/fonts/DejaVuSerif-metrics.xml</metricsFile>
  </font>
</font>
</fonts>
```

layout/load-scripts.jspx

```
<spring:url value="/resources/scripts/script.js" var="script_url" />
<script src="${script_url}" type="text/javascript"><!-- required for FF3 and Opera --></script>
<spring:url value="/resources/scripts/script-esp.js" var="script_esp_url" />
<script src="${script_esp_url}" type="text/javascript"><!-- required for FF3 and Opera
--></script>
<spring:url value="/resources/styles/roundbox.css" var="roundbox_url" />
<link rel="stylesheet" type="text/css" href="${roundbox_url}" />
<spring:url value="/resources/styles/estilo.css" var="estilo_url" />
<link rel="stylesheet" type="text/css" href="${estilo_url}" />
```

layout/classes/estilo.properties

styleSheet=resources/styles/estilo.css

layout/classes/style.properties

styleSheet=resources/styles/style.css

layout/layouts/alreves.jspx

```
<html xmlns:jsp="http://java.sun.com/JSP/Page" xmlns:c="http://java.sun.com/jsp/jstl/core"
xmlns:tiles="http://tiles.apache.org/tags-tiles" xmlns:spring="http://www.springframework.org/tags"
xmlns:util="urn:jsptagdir:/WEB-INF/tags/util" >

  <jsp:output doctype-root-element="HTML" doctype-system="about:legacy-compat" />

  <jsp:directive.page contentType="text/html; charset=UTF-8" />
  <jsp:directive.page pageEncoding="UTF-8" />

  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=8" />

    <util:load-scripts />

    <spring:message code="application name" var="app name" htmlEscape="false"/>
    <title><spring:message code="welcome_h3" arguments="${app_name}" /></title>
  </head>

  <body class="tundra spring">
    <div id="wrapper">
      <tiles:insertAttribute name="header" ignore="true" />
      <div id="main">
        <tiles:insertAttribute name="menu" ignore="true" />
        <tiles:insertAttribute name="footer" ignore="true" />
        <tiles:insertAttribute name="body" />
      </div>
    </div>
  </body>
</html>
```

layout/scripts/script-esp.js

```
document.writeln("Aplicacion de Ejemplo" );
```

layout/scripts/script.js

```
document.writeln("Example Application");
```

layout/styles/estilo.css

```
/* main elements */
body,div,td {
    font-family: Georgia, "Times New Roman", Times, serif;
    font-size: 12px;
    color: #666;
}
body {
    background-color: #fff;
    text-align: center;
}
#header {
    margin-bottom: 1em;
}
#wrapper {
    width:800px;
    min-width: 800px;
    max-width: 800px;
    margin-right: auto;
    margin-left: auto;

    /* fix max-width incompatibility in IE6 */
    width:expression(document.body.clientWidth > 800? "800px": "auto" );

    overflow:hidden;
    display:block;
}
/* header and footer elements */
#main {
    overflow:hidden;
    display:box;
}
#menu {
    background: #eee;
    position:relative;
    float:left;
    left:0px;
    width:220px;
    margin-right:15px;
}
#menu ul{
    list-style: none;
    margin: 0;
    padding: 0;
}
#menu ul li{
    padding: 0px;
}
#menu a, #menu h2 {
    display: block;
    margin: 0;
    padding: 2px 6px;
    color:#EEEEEF;
}
#menu h2 {
    color: #fff;
    background: #648C1D;
    text-transform: uppercase;
    font-weight:bold;
    font-size: 12px;
}
#menu a {
    color: #555555;
    background: #fefefe;
    text-decoration: none;
    padding: 2px 12px;
}
#menu a:hover {
    color: #648C1D;
    background: #fff;
}
#footer {
    background:#fff;
    border:none;
}
```

```

        margin-top:15px;
        border-top:1px solid #999999;
    }
#footer .new {
    float:left;
}
#footer a:link {color: #7db223;}
.quicklinks {
    clear:both;
    padding-bottom: 15px
}
.quicklinks span {
    float: right;
}
table.navigation {
    border: 0px;
}
label {
    width:100px;
    float:left;
    margin-left: 5px;
    margin-top: 0px;
}
input {
    height:20px;
}
input, textarea, select {
    border:1px solid #B3B3B3;
}
input.image {
    border: none;
    height: auto;
    vertical-align: middle;
}
submit {
    height:25px;
}
div {
    text-align: left;
}
div .box {
    display:block;
    margin-left:105px;
}
/* menu elements*/
a.menu, a.menu:link, a.menu:visited {display:block; width:150px; height:25px;}
/* text styles */
h1,h2,h3 {
    font-family: Helvetica, sans-serif;
    color: #7db223;
}
h1 {
    font-size: 20px;
    line-height: 26px;
}
h2 {
    font-size: 18px;
    line-height: 20px;
}
h3 {
    font-size: 15px;
    line-height: 21px;
    color:#555;
}
h4 {
    font-size: 14px;
    line-height: 20px;
}
.errors {
    color: red;
    font-weight: bold;
    display: block;
    margin-left: 105px;
}
a {
    text-decoration: underline;
    font-size: 12px;
}

```

```

}
a img {
  border: 0 none;
  vertical-align: middle;
}
tr:nth-child(odd) {
  background-color: #FFFFFF;
}
tr:nth-child(even) {
  background-color: #E0E0E0;
}
a:link {
  color: #7db223;
}
a:hover {
  color: #456314;
}
a:active {
  color: #7db223;
}
a:visited {
  color: #7db223;
}
li {
  padding-top: 5px;
  text-align: left;
}
ul li {
  margin: 0 0 0.25em 0;
  padding: 0;
}
/* table elements */
table {
  background: #E0E0E0;
  margin: 2px 0 0 0;
  border: 1px solid #BBBBBB;
  border-collapse: collapse;
  width: 100%
}
table table {
  margin: -5px 0;
  border: 0px solid #e0e7d3;
  width: 100%;
}
table td,table th {
  padding: 2px;
  border: 1px solid #CCCCCC;
}
table td form {
  text-align:center;
  vertical-align: middle;
  margin: 0px;
}
table th {
  font-size: 0.9em;
  text-align: left;
  font-weight: bold;
  color: #777;
  background: #888;
}
table thead {
  font-weight: bold;
  font-style: italic;
  background-color: #AAAAAA;
}
table a:link {color: #303030;}
.utilbox {width: 18px;}
caption {
  caption-side: top;
  width: auto;
  text-align: left;
  font-size: 12px;
  color: #848f73;
  padding-bottom: 4px;
}
fieldset {
  background: #e0e7d3;
  padding: 8px;
  padding-bottom: 22px;
}

```

```

        border: none;
        width: 560px;
    }
    fieldset label {
        width: 70px;
        float: left;
        margin-top: 1.7em;
        margin-left: 20px;
    }
    fieldset textfield {
        margin: 3px;
        height: 20px;
        background: #7d3e0e;
    }
    fieldset textarea {
        margin: 3px;
        height: 165px;
        background: #7d3e0e;
    }
    fieldset input {
        margin: 4px;
        height: 30px;
        background: #7d3e0e;
    }
    fieldset table {
        width: 100%;
    }
    fieldset th {
        padding-left: 85px;
    }
    .table-buttons {
        background-color:#eee;
        border:none;
    }
    .table-buttons td {
        border:none;
    }
    .submit input {
        border: 1px solid #CCCCCC;
        color:#111111;
        padding:2px 7px;
        font-size:11px;
        text-transform:uppercase;
        font-weight:bold;
        height:24px;
    }
    .updated {
        background:#ecf1e5;
        font-size:11px;
        margin-left:2px;
        border:4px solid #ecf1e5;
    }
    .updated td {
        padding:2px 8px;
        font-size:11px;
        color:#AAAAAA;
    }
    .dijitArrowButton {
        height: 20px;
    }
    .dijitTextArea{
        min-height:5.5em !important;
        max-height:22em !important;
        overflow-y: auto !important;
        max-width: 175px;
    }
    .RichTextEditable{
        min-height:18em !important;
        max-height:18em !important;
    }
    .flag {
        height: 16px;
        width: 22px;
    }

```

layout/styles/style.css

```

/* main elements */
body,div,td {
    font-family: Georgia, "Times New Roman", Times, serif;

```

```

        font-size: 26px;
        color: #666;
    }
body {
    background-color: #aaa;
    text-align: center;
}
#header {
    margin-bottom: 1em;
}
#wrapper {
    width:800px;
    min-width: 800px;
    max-width: 800px;
    margin-right: auto;
    margin-left: auto;

    /* fix max-width incompatibility in IE6 */
    width:expression(document.body.clientWidth > 800? "800px": "auto" );

    overflow:hidden;
    display:block;
}
/* header and footer elements */
#main {
    overflow:hidden;
    display:box;
}
#menu {
    background: #eee;
    position:relative;
    float:left;
    left:0px;
    width:220px;
    margin-right:15px;
}
#menu ul{
    list-style: none;
    margin: 0;
    padding: 0;
}
#menu ul li{
    padding: 0px;
}
#menu a, #menu h2 {
    display: block;
    margin: 0;
    padding: 2px 6px;
    color:#FFFFFF;
}
#menu h2 {
    color: #fff;
    background: #648C1D;
    text-transform: uppercase;
    font-weight:bold;
    font-size: 26px;
}
#menu a {
    color: #666666;
    background: #efefef;
    text-decoration: none;
    padding: 2px 24px;
}
#menu a:hover {
    color: #648C1D;
    background: #fff;
}
#footer {
    background:#fff;
    border:none;
    margin-top:15px;
    border-top:1px solid #999999;
}
#footer .new {
    float:left;
}
#footer a:link {color: #7db223;}
.quicklinks {
    clear:both;
    padding-bottom: 15px
}

```

```

.quicklinks span {
    float: right;
}
table.navigation {
    border: 0px;
}
label {
    width:100px;
    float:left;
    margin-left: 5px;
    margin-top: 0px;
}
input {
    height:28px;
}
input, textarea, select {
    border:1px solid #B3B3B3;
}
input.image {
    border: none;
    height: auto;
    vertical-align: middle;
}
submit {
    height:25px;
}
div {
    text-align: left;
}
div .box {
    display:block;
    margin-left:105px;
}
/* menu elements*/
a.menu, a.menu:link, a.menu:visited {display:block; width:150px; height:25px;}
/* text styles */
h1,h2,h3 {
    font-family: Helvetica, sans-serif;
    color: #7db223;
}
h1 {
    font-size: 26px;
    line-height: 36px;
}
h2 {
    font-size: 22px;
    line-height: 30px;
}
h3 {
    font-size: 20px;
    line-height: 24px;
    color:#555;
}
h4 {
    font-size: 18px;
    line-height: 20px;
}
.errors {
    color: red;
    font-weight: bold;
    display: block;
    margin-left: 105px;
}
a {
    text-decoration: underline;
    font-size: 20px;
}
a img {
    border: 0 none;
    vertical-align: middle;
}
tr:nth-child(odd) {
    background-color: #000000;
}
tr:nth-child(even) {
    background-color: #E0E0E0;
}

```

```

a:link {
    color: #7db223;
}
a:hover {
    color: #456314;
}
a:active {
    color: #7db223;
}
a:visited {
    color: #7db223;
}
li {
    padding-top: 5px;
    text-align: left;
}
ul li {
    margin:0 0 0.25em 0;
    padding:0;
}
/* table elements */
table {
    background: #EEEEEE;
    margin: 2px 0 0 0;
    border: 1px solid #BBBBBB;
    border-collapse: collapse;
    width: 100%
}
table table {
    margin: -5px 0;
    border: 0px solid #e0e7d3;
    width: 100%;
}
table td,table th {
    padding: 2px;
    border: 1px solid #CCCCCC;
}
table td form {
    text-align:center;
    vertical-align: middle;
    margin: 0px;
}
table th {
    font-size: 0.9em;
    text-align: left;
    font-weight: bold;
    color: #FFF;
    background: #999;
}
table thead {
    font-weight: bold;
    font-style: italic;
    background-color: #BBBBBB;
}
table a:link {color: #030303;}
.utilbox {width: 28px;}
caption {
    caption-side: top;
    width: auto;
    text-align: left;
    font-size: 12px;
    color: #848f73;
    padding-bottom: 4px;
}
fieldset {
    background: #e0e7d3;
    padding: 8px;
    padding-bottom: 22px;
    border: none;
    width: 560px;
}
fieldset label {
    width: 70px;
    float: left;
    margin-top: 1.7em;
    margin-left: 20px;
}
fieldset textfield {
    margin: 3px;
    height: 20px;
}

```



```

        background: #e0e7d3;
    }
    fieldset textarea {
        margin: 3px;
        height: 165px;
        background: #e0e7d3;
    }
    fieldset input {
        margin: 3px;
        height: 20px;
        background: #e0e7d3;
    }
    fieldset table {
        width: 100%;
    }
    fieldset th {
        padding-left: 25px;
    }
    .table-buttons {
        background-color:#fff;
        border:none;
    }
    .table-buttons td {
        border:none;
    }
    .submit input {
        border: 1px solid #BBBBBB;
        color:#777777;
        padding:2px 7px;
        font-size:11px;
        text-transform:uppercase;
        font-weight:bold;
        height:24px;
    }
    .updated {
        background:#ecf1e5;
        font-size:11px;
        margin-left:2px;
        border:4px solid #ecf1e5;
    }
    .updated td {
        padding:2px 8px;
        font-size:11px;
        color:#888888;
    }
    .dijitArrowButton {
        height: 20px;
    }
    .dijitTextArea{
        min-height:5.5em !important;
        max-height:22em !important;
        overflow-y: auto !important;
        max-width: 175px;
    }
    .RichTextEditable{
        min-height:18em !important;
        max-height:18em !important;
    }
    .flag {
        height: 11px;
        width: 16px;
    }

```

layout/tags/box/roundbox.tagx

```

<jsp:root xmlns:c="http://java.sun.com/jsp/jstl/core"
    xmlns:fn="http://java.sun.com/jsp/jstl/functions"
    xmlns:page="urn:jsptagdir:/WEB-INF/tags/form"
    xmlns:form="http://www.springframework.org/tags/form"
    xmlns:jsp="http://java.sun.com/JSP/Page"
    xmlns:spring="http://www.springframework.org/tags"
    version="2.0">
    <jsp:output omit-xml-declaration="yes"/>

    <jsp:directive.attribute name="id" type="java.lang.String" required="false" description="The
    identifier for this tag (do not change!)" />
    <jsp:directive.attribute name="cssClass" type="java.lang.String" required="true"
    description="The css class for this element" />
    <jsp:directive.attribute name="render" type="java.lang.Boolean" required="false"
    description="Indicate if the contents of this tag and all enclosed tags should be rendered (default
    'true')" />

    <c:if test="!${empty render or render}">

```

```

<c:choose>
  <c:when test="{empty id}">
    <div class="bl-${cssClass}">
      <div class="br-${cssClass}">
        <div class="tl-${cssClass}">
          <div class="tr-${cssClass}">
            <jsp:doBody />
          </div>
        </div>
      </div>
    </div>
  </c:when>
  <c:otherwise>
    <div id="bl-${id}" class="bl-${cssClass}">
      <div id="br-${id}" class="br-${cssClass}">
        <div id="tl-${id}" class="tl-${cssClass}">
          <div id="tr-${id}" class="tr-${cssClass}">
            <jsp:doBody />
          </div>
        </div>
      </div>
    </div>
  </c:otherwise>
</c:choose>
</c:if>
</jsp:root>

```

usermgmt/application.properties

```

# Remote security
AUTHENTICATION.REQUIRED=true
# Autenticacion para servicios remotos
REMOTE.USER.USERNAME=user
REMOTE.USER.PASSWORD=password
PASSWORD.ENCRYPT.TYPE=SIMPLE
# Properties file with server URL settings for remote access
cas.securityContext.service.url=${REMOTE.HOST}/${APP.NAME}
cas.securityContext.serviceProperties.service=${REMOTE.HOST}/${APP.NAME}/j_spring_cas_security_check
cas.securityContext.ticketValidator.casServerUrlPrefix=${REMOTE.USERMODULE.HOST}/${USERMGMT.NAME}
cas.securityContext.casAuthenticationFilterEntryPoint.loginUrl=${
cas.securityContext.ticketValidator.casServerUrlPrefix}/login
cas.securityContext.casAuthenticationFilterEntryPoint.logoutUrl=${
cas.securityContext.ticketValidator.casServerUrlPrefix}/j_spring_security_logout
cas.securityContext.remoteUsersService=${REMOTE.USERMODULE.HOST}/${
USERMGMT.NAME}/remoting/remoteUsersService
cas.securityContext.remotePrincipalService=${USERMGMT.NAME}/remoting/remotePrincipalService
cas.securityContext.remoteUserDetailsService=${REMOTE.USERMODULE.HOST}/${
USERMGMT.NAME}/remoting/remoteUserDetailsService
cas.securityContext.remoteAuthenticationManagerService=${REMOTE.USERMODULE.HOST}/${
USERMGMT.NAME}/remoting/remoteAuthenticationService
# Role security
application.security.role=ROLE_ADMIN

```

usermgmt/applicationContext-security.xml

usermgmt/artifactItems.txt

```

usermgmt-application
war
${usermgmt.delivery}.war
usermgmt-doc
zip
${usermgmt.delivery}-docs.zip
usermgmt-scripts
zip
${usermgmt.delivery}-scripts.zip

```

usermgmt/changePassword.jspx

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<div id="user_password" xmlns:jsp="http://java.sun.com/JSP/Page"
  xmlns:userInfo="http://www.empresa.com/usermgmt/userInfo"
  xmlns:spring="http://www.springframework.org/tags"
  class="body_info"
  version="2.0">
  <jsp:output omit-xml-declaration="yes" />

  <fieldset>
    <legend>
      <spring:message
        code="label usermgmt user username"
        arguments="{userPasswordChange.username}" />
    </legend>
    <userInfo:changePassword />
  </fieldset>
</div>

```

usermgmt/userInfo.jspx

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<div id="user_info" xmlns:jsp="http://java.sun.com/JSP/Page"
    xmlns:userInfo="http://www.empresa.com/usermgmt/userInfo"
    xmlns:spring="http://www.springframework.org/tags"
    class="body_info"
    version="2.0">
    <jsp:output omit-xml-declaration="yes" />

    <fieldset>
        <legend>
            <spring:message
                code="label usermgmt_user_username"
                arguments="{user.usērnāmē}" />
        </legend>
        <userInfo:userInfo />
    </fieldset>
</div>
```

usermgmt/usermgmt.txt

```
com.empresa.usermgmt
usermgmt-client
1.0
http://cas.empresa.com:8080
appname
```