Migrazione alla 4.6.X

Prerequisiti

Java

Java 8 o superiore.

Application Server

Tomcat

1. https://tomcat.apache.org/download-90.cgi
2. https://tomcat.apache.org/download-80.cgi

Non sono supportate versioni 10+, basate su **jakarta**

Wildfly

1. wildfly 20.0.1.Final https://www.wildfly.org/downloads/

Database

Oracle

Geoweb 4.6.0 utilizza il driver oracle **ojdbc8 v 21.5.0.0**.

In base alla matrice di compatibilità ufficiale, per utilizzare questa versione di Geoweb è necessario assicurarsi che la versione del database sia **'12.2 or 12cR2' o superiore**.

Matrice di compatibilità *qui*

Postgre

Geoweb 4.6.0 utilizza il driver **postgresql v 42.6.0**. In base alla matrice di compatibilità ufficiale, per utilizzare questa versione di Geoweb è necessario assicurarsi che la versione del database sia **"PostgreSQL 8.2 or newer"**.

Matrice di compatibilità *qui*.

MsSqlServer

Geoweb 4.6.0 utilizza il driver **mssql-jdbc v 12.2.0.jre8**.

In base alla matrice di compatibilità ufficiale, per utilizzare questa versione di Geoweb è necessario assicurarsi che la versione del database sia **SQL Server 2014**.

Matrice di compatibilità qui

MongoDB

La minima versione testata è **MongoDB 4.2**. In caso contrario l'applicativo segnalerà un errore del tipo:

```
....ERROR .... - Server at 127.0.0.1:27017 reports wire version 4, but this
version of the driver requires at least 6 (MongoDB 3.6).
```

Alfresco

I vecchi componenti CMIS continuano a lavorare, come prima, con **Alfresco 5**. I nuovi componenti CMS, necessitano di **Alfresco 7**.

SharePoint

Passaggi da seguire

- Creare un repository ex-novo ex-novo o modificare quello esistente intervenendo su tutti i file necessari come descritto nella parte Migrazione repository
- Apportare le opportune modifiche nello schema dei metadati, come descritto nella parte Migrazione metadati
- leggere le Note Migrazione

Migrazione repository

Repository git nuovi

I nuovi repository git, di progetto o di commessa, dovrebbero utilizzare fin da subito il nuovo template di progetto **gw-webapp-template**. La prima cosa da fare per creare un progetto di base nel proprio pc cliccare sul seguente link e avviare il download della cartella ZIP

makegproj-4.6.x.zip

.

Repository git preesistenti

Per tutti i progetti git in corso:

- va adeguato il pom.xml.
- vanno sostituiti specifici file nel vecchio template di progetto. file interessati:
    - spring-security.xml
    - dispatcher-servlet.xml
    - web.xml
    - configuration.properties
    - log4j.properties, sostituito da log4j2.properties

Questi file potrebbero comunque necessitare di ulteriori integrazioni dovute a personalizzazioni a suo tempo seguite sul cliente.

Note

log4j.properties va proprio eliminato. Se configurato male può causare la non partenza del server, con errore sul log standard error del server.

```
log4j:ERROR Could not find value for key log4j.appender.access
log4j:ERROR Could not instantiate appender named "access".
```

pom.xml

Per le versioni 4.6.X è stato creato un apposito template **gw-webapp-template 1.0.0**, che sostituisce il vecchio template **gw-webapp-template 1.0.0**. Il template va generato tramite

makegproj-4.6.x.zip

.

Al contempo sono state fatti vari refactor di vario genere sui maven module del framework:

- refactor nome dei moduli
    - tutti i moduli ora partono per 'gw-'
    - tutti name in minuscolo
    - key-word staccate con '-'
    - niente parole ridondanti nei name dei moduli maven: *plugin, service, utils*
- *artifactId* (*non il groupId*) del pom generale geowebframework passa da *<artifactId>com.geowebframework</artifactId>* a *<artifactId>geowebframework</artifactId>* (interessa solo gli sviluppatori che devono generare nuovi maven plugin)
- moduli integrati nel repository git di *geowebframework*
- moduli tirati fuori dal repository git di *geowebframework*, in quanto di prodotto/commessa

Per i progetti preesistenti modificare il pom.xml utilizzando i nuovi *artifactId*, e lasciando inalterato il *groupId*.

Tabella riepilogativa

NOME MODULO MAVEN (artifactId)		spostato da git geowebframework	MOVED/DELETED	Deprecated	NOTE
VECCHIO	NUOVO				
calendar	gw-advanced				
cde	gw-cde	SI	(git CDE)		specifico CDE, da spostare su repo di prodotto
classificationplugin	gw-classification				
gwApproval	gw-approvals				
furnitureplugin	gw-furnitures	SI	(git SPACE)		specifico SPACE, da spostare su repo di prodotto
googleStreetView	gw-google-street-view				
dataservice	gw-data				
-	gw-cms				versione 1.0.0. Aggiunto per ospitare i servizi cms rest
gwCsvConnector	gw-csv-connector		MOVED		integrato in git geowebframework, versionamento corrente 2.0.0
gwindex	gw-index			SI	pagine di accesso specifiche per il cde, da spostare su repo di prodotto
gwMnemonicCode	gw-mnemonic-code				
gwProvisioning	gw-provisioning				usato solo da Università Padova, ma generale (provisioning dei loro utenti, copiati sul nostro DB)
gwRabbitApi	gw-rabbit-api				
gwRabbitConsumer	gw-rabbit-consumer				
gwRabbitPublisher	gw-rabbit-publisher				
gwScenari	gw-scenarios				
gwXktVisualizer	gw-xkt-visualizer				
hierarchicalFilterMenu			MOVED		integrato in git geowebframework, ora senza versionamento specifico
indexPortal	gw-index-portal			SI	pagine di accesso specifiche per vecchie commesse
keycloak-userspi	gw-keycloak-userspi				intesa Cloud, rimane in comune per ora ma non più usato con il nuovo KEYCLOACK
metadataservice	gw-metadata				
portalApi	gw-portal-api			SI	
portalObjects	gw-portal-commons			SI	

NOME MODULO MAVEN (artifactId)		spostato da git geowebframework	MOVED/DELETED	Deprecated	NOTE
VECCHIO	NUOVO				
printMapFlow	-		MOVED		usato solo in SPACE (Intesa) per ora, ma di carattere generale: accorpato in gw-print-map-legend, senza versionamento sprcifico
printMapLegend	gw-print-map-legend				
restapi	gw-rest-api	SI			rimosse chiamate del modulo EQUIPMENT (@Deprecated), tenuto per spostare servizi interoperabilità
smartplatformplugin	gw-smartplatform	SI	MOVED	SI	widget (indicatori in mappa auto aggiornnanti, serie storiche, etc..) e funzionalità per integrazione dati sensori provenienti dalla smartplatform
thematism	gw-thematism				
ThreeDVisualizer	gw-3d-visualizer				
transfer-objects	gw-commons + gw-commons-web				splittato in due moduli gw-commons + gw-commons-web
umplugin	gw-upload-manager				
webadmin	gw-webadmin				
webclient	gw-webclient				
workflowservice	gw-workflow				
xeokitUtils	gw-xeokit	SI	DELETED	SI	cancellato, mai usato nel CDE (sostituito da rabbit-consumer)
webclientTemplate	gw-webapp-template				nuovo template 1.0.0

excel censimento moduli:

https://docs.google.com/spreadsheets/d/1kqvL8uERY4FJDFaPrl44kl7L1QYpds4pR55I1MhUGv8/edit#gid=0

Esempi

Esempio 1:

```xml
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>calendar</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
...
<dependency>
    <groupId>com.geowebframework</groupId>
```

```xml
    <artifactId>ThreeDVisualizer</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
...
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>umplugin</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
```

diventa:

```xml
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>gw-advanced</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
...
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>gw-3d-visualizer</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
...
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>gw-upload-manager</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
```

Esempio 2, da rimuovere:

```xml
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>printMapFlow</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
...
<dependency>
    <groupId>com.geowebframework</groupId>
    <artifactId>hierarchicalFilterMenu</artifactId>
    <version>${com.geowebframework.version}</version>
</dependency>
```

spring-security.xml

Localizzazione: *WEB-INF/spring-security.xml*

Nel nuovo template di progetto **gw-webapp-template**, sono resi disponibili i seguenti template di *spring-security*.xml*:

- spring-security.xml-base - NO KEYCLOAK
- spring-security-keycloak.xml - KEYCLOAK
- spring-security-deprecated.xml - NO KEYCLOAK, ma con altri tag http, meccanismi di autenticazione, ora deprecati, o di uso particolare:

  - `<http pattern="/oauth/*"`

    usati dalle app e servizi rest, legati alla parte oauth2

  - `<http pattern="/services/**"`

    usati dalle app e servizi rest, legati alla parte oauth2

  - preAuthFilter

    `<beans:bean id="preAuthFilter"`

  - preAuthAuthenticationProviderExt

    `<beans:bean id="preAuthAuthenticationProviderExt"`

  - oauth2

    ```
    <beans:bean id="oauthAuthenticationEntryPoint"
    class="org.springframework.security.oauth2.provider.error.OAuth2Au
    thenticationEntryPoint">
    ```

  - LDAP

    ```
    <beans:bean id="adAuthenticationProvider"
    class="org.springframework.security.ldap.authentication.ad.ActiveD
    irectoryLdapAuthenticationProvider">
    ```

Quando usare uno specifico template

- spring-security.xml Nei casi **dove c'è un ID (Identity Provider)** esterno o comunque messo a diposizione dal cliente (i cui *bean* di configurazione dovranno eventualmente essere integrati)
- spring-security-keycloak.xml Nei casi **dove non c'è un ID (Identity Provider)** del cliente, o nei nostri ambienti interni, dove l'ID scelto è per l'appunto *keycloak*
- spring-security-deprecated.xml **MAI**, usarlo solo come riferimento per integrare negli altri due template solo la parte eventualmente necessaria

spring-security-base.xml

spring-security.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans
  xmlns="http://www.springframework.org/schema/security"
  xmlns:beans="http://www.springframework.org/schema/beans"
  xmlns:context="http://www.springframework.org/schema/context"
  xmlns:oauth="http://www.springframework.org/schema/security/oauth2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="
      http://www.springframework.org/schema/security
http://www.springframework.org/schema/security/spring-security.xsd
      http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
      http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context.xsd
      http://www.springframework.org/schema/security/oauth2
http://www.springframework.org/schema/security/spring-security-oauth2.x
sd
  "
>

    <context:component-scan base-
package="com.geowebframework.webclient.security" />

    <beans:import resource="classpath:metadata-config-spring-
security.xml" />

    <beans:import resource="classpath*:META-INF/*.beandef-security.xml"
/>

    <beans:bean id="messageSource"
class="com.geowebframework.metadataservice.messagesource.GwReloadableRe
sourceBundleMessageSource">
        <beans:property name="basenames">
            <beans:list>
                <beans:value>classpath*:locale/messages*</beans:value>
<!-- load all .properties files contained under folder locale, which
name starts with 'messages'. Works even for plug-in -->
            </beans:list>
        </beans:property>
        <beans:property name="defaultEncoding" value="UTF-8"/>
        <beans:property name="useCodeAsDefaultMessage"
value="false"/><!-- Set whether to use the message code as default
message instead  of throwing a NoSuchMessageException. Useful for
development and debugging. Default is "false".  -->
    </beans:bean>

    <!-- PropertyPlaceholderConfigurer still needed here,
GwPropertyPlaceholderConfigurer not used -->
    <beans:bean
```

```xml
class="org.springframework.beans.factory.config.PropertyPlaceholderConf
igurer">
        <beans:property name="locations"
value="classpath:configuration.properties"/>
    </beans:bean>


    <!-- Add disable-url-rewriting set to true to prevent URL rewriting
of jsessionid -->
    <http auto-config="true" use-expressions="true" disable-url-
rewriting="true">
        <csrf request-matcher-ref="csrfMatcher"/>

        <!--
         issue #589 set response header to prevent clickjacking attacks
         it solves browser console error:
             Refused to display 'http://localhost:8081/' in a frame
because it set 'X-Frame-Options' to 'deny'.
        -->
        <headers>
            <frame-options policy="SAMEORIGIN" />
            <header name="X-Content-Security-Policy" value="default-src
inline-script eval-script; frame-ancestors 'self'"/>
            <hsts include-subdomains="true" max-age-seconds="31536000"
/>
        </headers>


        <!-- IMPORTANT -->
        <!-- Patterns are always evaluated in the order they are
defined. Thus it is important that more specific patterns are defined
higher in the list than less specific patterns. -->

        <intercept-url pattern="/login*" access="permitAll" /> <!--
cover also 'loginPage' -->
        <intercept-url pattern="/forgotPassword*" access="permitAll" />
        <intercept-url pattern="/resetPassword*" access="permitAll" />
        <intercept-url pattern="/forceChangePassword*"
access="permitAll" />
        <intercept-url pattern="/accessdenied*" access="permitAll" />
        <intercept-url pattern="/emptyMetadataCache" access="permitAll"
/>
        <intercept-url pattern="/emptyClassificationCache"
access="permitAll" />
        <intercept-url pattern="/emptyMnemonicCodeCache"
access="permitAll" />
        <intercept-url pattern="/livenessProbe" access="permitAll" />
<!-- kubernetes support -->
        <intercept-url pattern="/readinessProbe" access="permitAll" />
<!-- kubernetes support -->

        <!-- set permitAll for all used resources inside login pages -
```

```
->
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/debug/dojo/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/debug/dijit/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/debug/dojox/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/compressed/dojo/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/compressed/dijit/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/compressed/dojox/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/css/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/icons/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/fontawesome/**" access="permitAll" /> <!-- issue #509 -->
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/images/favicon.png" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/images/${images.login.form.logo:gw4_enterprise.png}"
access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/${images.IndexBackground:IndexBackground.jpg}"
access="permitAll" />

        <!-- for all other resources should to be isAuthenticated() or
isAnonymous() -->
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/**" access="isAuthenticated()" />

        <intercept-url pattern="/index*" access="isAuthenticated()" />

        <!-- @@@ANONYMOUS_USER@@@ -->
```

```xml
        <!--
         -to enable anonymous access: comment 1 and decomment 2
          write the right group name inside 'anonymous' tag, in
'granted-authority' attribute
          (if omitted, the default role is ROLE_ANONYMOUS)
        -->
        <!--1-->
        <intercept-url pattern="/**" access="isAuthenticated()" />
        <!--2
      <intercept-url pattern="/**" access="isAuthenticated() or
isAnonymous()" />
      <anonymous username="" granted-authority="ROLE_ANONYMOUS" />-->


        <!--
         defaults tag attributes:
             password-parameter="password"
             username-parameter="username"
             login-processing-url="login"
        -->
        <form-login
          login-page="/loginPage.html"
          default-target-url="/index.html"
          authentication-failure-handler-
ref="gwAuthenticationFailureHandler"

         />
        <!--
         alternative to authentication-failure-handler-ref:
             authentication-failure-url="/loginPage.html?error" />
        -->

        <!--STANDARD OR CUSTOM LOGOUT, decomment the one to be used-->

        <!--STANDARD LOGOUT-->
        <!--choose one between-->
        <!--<logout invalidate-session="true" logout-url="/gwlogout"
logout-success-url="/" />-->
        <!--<logout invalidate-session="true" logout-url="/gwlogout"
logout-success-url="/loginPage.html?logout" />-->
        <logout invalidate-session="true" logout-url="/gwlogout"
logout-success-url="/loginPage.html?logout" />

        <!--CUSTOM LOGOUT-->
        <!--
      <logout success-handler-ref="customLogoutSuccessHandler" />
      -->
    </http>

    <!--CUSTOM LOGOUT-->
    <!--
```

```
 <beans:bean id="customLogoutSuccessHandler"
class="com.geowebframework.webclient.controller.CustomLogoutSuccessHand
ler" />
 -->

    <beans:bean id="csrfMatcher"
class="org.springframework.security.web.util.matcher.OrRequestMatcher">
        <beans:constructor-arg>
            <beans:list value-
type="org.springframework.security.web.util.matcher.RequestMatcher">
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="login*"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/*/groovyAction/generic/**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/*/groovyAction/detail/**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/*/groovyAction/list/**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/editDetail/*/insertDetail**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
```

```xml
                    </beans:bean>
                    <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                        <beans:constructor-arg name="pattern"
value="/rest/editDetail/*/updateDetail**"/>
                        <beans:constructor-arg name="httpMethod"
value="POST"/>
                    </beans:bean>
                    <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                        <beans:constructor-arg name="pattern"
value="/rest/editDetail/*/deleteDetail**"/>
                        <beans:constructor-arg name="httpMethod"
value="POST"/>
                    </beans:bean>
                    <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                        <beans:constructor-arg name="pattern"
value="/rest/saveAllMultipleGeometryEdit**"/>
                        <beans:constructor-arg name="httpMethod"
value="POST"/>
                    </beans:bean>
                </beans:list>
            </beans:constructor-arg>
        </beans:bean>

    <beans:bean id="filterMultipartResolver"
class="org.springframework.web.multipart.commons.CommonsMultipartResolv
er">
        <beans:property name="maxUploadSize"
value="${commonsMultipartResolver.maxUploadSize:209715200}" />
    </beans:bean>


    <beans:bean id="userLoginLogger"
class="com.geowebframework.webclient.security.LogFileLogger"/>
    <beans:bean id="authenticationSuccessEventListner"
class="com.geowebframework.webclient.security.AuthenticationSuccessEven
tListner">
        <beans:constructor-arg ref = "userLoginLogger"/>
    </beans:bean>

    <beans:bean id="gwAuthenticationFailureHandler"
class="com.geowebframework.webclient.security.GwAuthenticationFailureHa
ndler"/>


    <beans:bean id="metadataSource"
```

```xml
class="org.apache.commons.dbcp.BasicDataSource"
      destroy-method="close">
        <beans:property name="driverClassName"
value="${jdbcmetadata.driverClassName}" />
        <beans:property name="url" value="${jdbcmetadata.url}" />
        <beans:property name="username"
value="${jdbcmetadata.username}" />
        <beans:property name="password"
value="${jdbcmetadata.password}" />
        <beans:property name="maxActive" value="2" />
        <beans:property name="minIdle" value="1" />
        <beans:property name="maxIdle" value="2" />
        <beans:property name="validationQuery"
value="${jdbcmetadata.validationQuery}"/>
    </beans:bean>

    <beans:bean id="metadataSqlSessionFactory"
class="org.mybatis.spring.SqlSessionFactoryBean">
        <beans:property name="dataSource" ref="metadataSource" />
        <beans:property name="configLocation" value="classpath:mybatis-
config.xml" />
    </beans:bean>


    <beans:bean id="customUserDetailsService"
class="com.geowebframework.webclient.security.CustomUserDetailsService"
>
        <beans:property name="usersByUsernameQuery" value="select *
from GWM_USERS where username = ?"/>
        <beans:property name="authoritiesByUsernameQuery" value="select
GWM_USER_GROUPS.*, usergroup as authority from GWM_USER_GROUPS where
username =?" />
        <beans:property name="dataSource">
            <beans:ref bean="metadataSource"></beans:ref>
        </beans:property>
    </beans:bean>

    <beans:bean id="gwUserDetailsDao"
class="com.geowebframework.webclient.security.GwUserDetailsDaoImpl"
></beans:bean>

    <beans:bean id="limitLoginAuthenticationProvider"
class="com.geowebframework.webclient.security.LimitLoginAuthenticationP
rovider">
        <beans:property name="userDetailsService"
ref="customUserDetailsService" />
        <beans:property name="passwordEncoder" ref="passwordEncoder" />
    </beans:bean>
```

```
                <authentication-manager alias="authenticationManager">

                        <authentication-provider
ref="limitLoginAuthenticationProvider"></authentication-provider>

                </authentication-manager>

        </beans:beans>
```

spring-security-keycloak.xml

spring-security-keycloak.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans:beans
    xmlns="http://www.springframework.org/schema/security"
    xmlns:beans="http://www.springframework.org/schema/beans"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:oauth="http://www.springframework.org/schema/security/oauth2"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="
        http://www.springframework.org/schema/security
http://www.springframework.org/schema/security/spring-security.xsd
        http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context.xsd
        http://www.springframework.org/schema/security/oauth2
http://www.springframework.org/schema/security/spring-security-oauth2.x
sd
    "
>

    <context:component-scan base-
package="com.geowebframework.webclient.security" />

    <beans:import resource="classpath:metadata-config-spring-
security.xml" />

    <beans:import resource="classpath*:META-INF/*.beandef-security.xml"
/>

    <beans:bean id="messageSource"
class="com.geowebframework.metadataservice.messagesource.GwReloadableRe
sourceBundleMessageSource">
        <beans:property name="basenames">
            <beans:list>
                <beans:value>classpath*:locale/messages*</beans:value>
```

```xml
<!-- load all .properties files contained under folder locale, which
name starts with 'messages'. Works even for plug-in -->
            </beans:list>
        </beans:property>
        <beans:property name="defaultEncoding" value="UTF-8"/>
        <beans:property name="useCodeAsDefaultMessage"
value="false"/><!-- Set whether to use the message code as default
message instead  of throwing a NoSuchMessageException. Useful for
development and debugging. Default is "false".  -->
    </beans:bean>

    <!-- @@@KEYCLOAK@@@ STARTS -->
    <beans:bean id="adapterDeploymentContext"
class="org.keycloak.adapters.springsecurity.AdapterDeploymentContextFac
toryBean">
        <beans:constructor-arg value="/WEB-INF/keycloak.json" />
    </beans:bean>

    <beans:bean id="keycloakAuthenticationEntryPoint"
class="org.keycloak.adapters.springsecurity.authentication.KeycloakAuth
enticationEntryPoint">
        <beans:constructor-arg name="adapterDeploymentContext"
ref="adapterDeploymentContext" />
    </beans:bean>

    <beans:bean id="gwKeycloakAuthenticationProvider"
class="com.geowebframework.keycloak.authentication.GwKeycloakAuthentica
tionProvider" >
        <beans:constructor-arg name="gwKeycloakUserDetailsService"
ref="gwKeycloakUserDetailsService" />
    </beans:bean>

    <beans:bean id="gwKeycloakUserDetailsService"
class="com.geowebframework.keycloak.userdetails.GwKeycloakUserDetailsSe
rvice">
        <beans:property name="dataSource">
            <beans:ref bean="metadataSource"></beans:ref>
        </beans:property>
    </beans:bean>

    <beans:bean id="httpSessionManager"
class="org.keycloak.adapters.springsecurity.management.HttpSessionManag
er" />
    <beans:bean id="keycloakPreAuthActionsFilter"
class="org.keycloak.adapters.springsecurity.filter.KeycloakPreAuthActio
nsFilter" >
        <beans:constructor-arg ref="httpSessionManager"/>
    </beans:bean>
```

```xml
    <beans:bean id="keycloakAuthenticationProcessingFilter"
class="org.keycloak.adapters.springsecurity.filter.KeycloakAuthenticati
onProcessingFilter">
        <beans:constructor-arg name="authenticationManager"
ref="authenticationManager" />
    </beans:bean>

    <beans:bean id="keycloakLogoutHandler"
class="org.keycloak.adapters.springsecurity.authentication.KeycloakLogo
utHandler">
        <beans:constructor-arg ref="adapterDeploymentContext" />
    </beans:bean>

    <beans:bean id="logoutFilter"
class="org.springframework.security.web.authentication.logout.LogoutFil
ter">
        <beans:constructor-arg name="logoutSuccessUrl" value="/" />
        <beans:constructor-arg name="handlers">
            <beans:list>
                <beans:ref bean="keycloakLogoutHandler" />
                <beans:bean
class="org.springframework.security.web.authentication.logout.SecurityC
ontextLogoutHandler" />
            </beans:list>
        </beans:constructor-arg>
        <beans:property name="logoutRequestMatcher">
            <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                <beans:constructor-arg name="pattern"
value="/gwlogout**" />
                <beans:constructor-arg name="httpMethod" value="POST"
/>
            </beans:bean>
        </beans:property>
    </beans:bean>
    <!-- @@@KEYCLOAK@@@ ENDS -->


    <!-- Add disable-url-rewriting set to true to prevent URL rewriting
of jsessionid -->
    <http auto-config="false" use-expressions="true" disable-url-
rewriting="true" entry-point-ref="keycloakAuthenticationEntryPoint">
        <csrf disabled="true"/>
        <!--  <csrf request-matcher-ref="csrfMatcher"/>  -->

        <!--
          issue #589 set response header to prevent clickjacking attacks
          it solves browser console error:
             Refused to display 'http://localhost:8081/' in a frame
because it set 'X-Frame-Options' to 'deny'.
```

```xml
        -->
        <headers>
            <frame-options policy="SAMEORIGIN" />
            <header name="X-Content-Security-Policy" value="default-src
inline-script eval-script; frame-ancestors 'self'"/>
            <hsts include-subdomains="true" max-age-seconds="31536000"
/>
        </headers>

        <!-- @@@KEYCLOAK@@@ -->
        <custom-filter ref="keycloakPreAuthActionsFilter"
before="LOGOUT_FILTER" />
        <custom-filter ref="keycloakAuthenticationProcessingFilter"
before="FORM_LOGIN_FILTER" />

        <!-- IMPORTANT -->
        <!-- Patterns are always evaluated in the order they are
defined. Thus it is important that more specific patterns are defined
higher in the list than less specific patterns. -->

        <intercept-url pattern="/login*" access="permitAll" /> <!--
cover also 'loginPage' -->
        <intercept-url pattern="/forgotPassword*" access="permitAll" />
        <intercept-url pattern="/resetPassword*" access="permitAll" />
        <intercept-url pattern="/forceChangePassword*"
access="permitAll" />
        <intercept-url pattern="/accessdenied*" access="permitAll" />
        <intercept-url pattern="/emptyMetadataCache" access="permitAll"
/>
        <intercept-url pattern="/emptyClassificationCache"
access="permitAll" />
        <intercept-url pattern="/emptyMnemonicCodeCache"
access="permitAll" />
        <intercept-url pattern="/livenessProbe" access="permitAll" />
<!-- kubernetes support -->
        <intercept-url pattern="/readinessProbe" access="permitAll" />
<!-- kubernetes support -->

        <!-- set permitAll for all used resources inside login pages -
->
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/debug/dojo/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/debug/dijit/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/debug/dojox/**" access="permitAll" />
```

```xml
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/compressed/dojo/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/compressed/dijit/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/compressed/dojox/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/css/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/icons/**" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/fontawesome/**" access="permitAll" /> <!-- issue #509 -->
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/images/favicon.png" access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/images/${images.login.form.logo:gw4_enterprise.png}"
access="permitAll" />
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/${images.IndexBackground:IndexBackground.jpg}"
access="permitAll" />

        <!-- for all other resources should to be isAuthenticated() or
isAnonymous() -->
        <intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevis
ion() }/**" access="isAuthenticated()" />

        <intercept-url pattern="/index*" access="isAuthenticated()" />

        <!-- @@@ANONYMOUS_USER@@@ -->
        <!--
         -to enable anonymous access: comment 1 and decomment 2
          write the right group name inside 'anonymous' tag, in
'granted-authority' attribute
          (if omitted, the default role is ROLE_ANONYMOUS)
      -->
        <!--1-->
        <intercept-url pattern="/**" access="isAuthenticated()" />
        <!--2
     <intercept-url pattern="/**" access="isAuthenticated() or
isAnonymous()" />
      <anonymous username="" granted-authority="ROLE_ANONYMOUS" />-->
```

```xml
            <!-- @@@KEYCLOAK@@@ -->
        <custom-filter ref="logoutFilter" position="LOGOUT_FILTER" />

    </http>

    <beans:bean id="csrfMatcher"
class="org.springframework.security.web.util.matcher.OrRequestMatcher">
        <beans:constructor-arg>
            <beans:list value-
type="org.springframework.security.web.util.matcher.RequestMatcher">
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="login*"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/*/groovyAction/generic/**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/*/groovyAction/detail/**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/*/groovyAction/list/**"/>
                    <beans:constructor-arg name="httpMethod"
value="POST"/>
                </beans:bean>
                <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                    <beans:constructor-arg name="pattern"
value="/rest/editDetail/*/insertDetail**"/>
                    <beans:constructor-arg name="httpMethod"
```

```xml
value="POST"/>
                    </beans:bean>
                    <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                        <beans:constructor-arg name="pattern"
value="/rest/editDetail/*/updateDetail**"/>
                        <beans:constructor-arg name="httpMethod"
value="POST"/>
                    </beans:bean>
                    <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                        <beans:constructor-arg name="pattern"
value="/rest/editDetail/*/deleteDetail**"/>
                        <beans:constructor-arg name="httpMethod"
value="POST"/>
                    </beans:bean>
                    <beans:bean
class="org.springframework.security.web.util.matcher.AntPathRequestMatc
her">
                        <beans:constructor-arg name="pattern"
value="/rest/saveAllMultipleGeometryEdit**"/>
                        <beans:constructor-arg name="httpMethod"
value="POST"/>
                    </beans:bean>
            </beans:list>
        </beans:constructor-arg>
    </beans:bean>

    <beans:bean id="filterMultipartResolver"
class="org.springframework.web.multipart.commons.CommonsMultipartResolv
er">
        <beans:property name="maxUploadSize"
value="${commonsMultipartResolver.maxUploadSize:209715200}" />
    </beans:bean>

    <!-- TODO TEST -->
    <beans:bean id="userLoginLogger"
class="com.geowebframework.webclient.security.LogFileLogger"/>
    <beans:bean id="authenticationSuccessEventListner"
class="com.geowebframework.webclient.security.AuthenticationSuccessEven
tListner">
        <beans:constructor-arg ref = "userLoginLogger"/>
    </beans:bean>
    <!-- TODO TEST -->
    <beans:bean id="gwAuthenticationFailureHandler"
class="com.geowebframework.webclient.security.GwAuthenticationFailureHa
ndler"/>
```

```xml
    <beans:bean id="metadataSource"
class="org.apache.commons.dbcp.BasicDataSource"
        destroy-method="close">
        <beans:property name="driverClassName"
value="${jdbcmetadata.driverClassName}" />
        <beans:property name="url" value="${jdbcmetadata.url}" />
        <beans:property name="username"
value="${jdbcmetadata.username}" />
        <beans:property name="password"
value="${jdbcmetadata.password}" />
        <beans:property name="maxActive" value="2" />
        <beans:property name="minIdle" value="1" />
        <beans:property name="maxIdle" value="2" />
        <beans:property name="validationQuery"
value="${jdbcmetadata.validationQuery}"/>
    </beans:bean>

    <beans:bean id="metadataSqlSessionFactory"
class="org.mybatis.spring.SqlSessionFactoryBean">
        <beans:property name="dataSource" ref="metadataSource" />
        <beans:property name="configLocation" value="classpath:mybatis-
config.xml" />
    </beans:bean>

    <authentication-manager alias="authenticationManager">

        <!-- @@@KEYCLOAK@@@ -->
        <authentication-provider ref="gwKeycloakAuthenticationProvider"
/>

    </authentication-manager>

</beans:beans>
```

Modifiche

E' stato modificato il blocco

```xml
<http auto-config="true" use-expressions="true" disable-url-
rewriting="true">...</http>
```

in particolare sono stati aggiunte le parti:

```xml
...
<intercept-url pattern="/livenessProbe" access="permitAll" />
<intercept-url pattern="/readinessProbe" access="permitAll" />
```

```
..
<intercept-url pattern="/resources/#{
T(com.geowebframework.metadataservice.registry.GwReleaseInfos).getRevision()
}/fontawesome/**" access="permitAll" /> <!-- issue #509 -->
```

e modificate queste:

```
        <!--
         password-parameter="password"
         username-parameter="username"
         login-processing-url="login"
    -->
        <form-login
          login-page="/loginPage.html"
          default-target-url="/index.html"
          authentication-failure-handler-
ref="gwAuthenticationFailureHandler"

      />
        <!--     authentication-failure-url="/loginPage.html?error" /> -->




        <!--STANDARD OR CUSTOM LOGOUT, decomment the one to be used-->
        <!--STANDARD LOGOUT-->
        <!--<logout logout-success-url="/" /> or <logout logout-success-
url="/loginPage.html?logout" />-->
        <logout
          invalidate-session="true"
          logout-url="/gwlogout"
          logout-success-url="/loginPage.html?logout"
      />
        <!--CUSTOM LOGOUT-->
        <!--
    <logout success-handler-ref="customLogoutSuccessHandler" />
      -->
```

Sono stati inoltre rimossi bean ora conservati in **spring-security-deprecated.xml**

dispatcher-servlet.xml

Localizzazione: *WEB-INF/dispatcher-servlet.xml*

dispatcher-servlet.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:mvc="http://www.springframework.org/schema/mvc"
   xmlns:context="http://www.springframework.org/schema/context"
   xmlns:mybatis="http://mybatis.org/schema/mybatis-spring"
xmlns:util="http://www.springframework.org/schema/util"
   xmlns:aop="http://www.springframework.org/schema/aop"
   xmlns:task="http://www.springframework.org/schema/task"
xmlns:ehcache="http://ehcache-spring-annotations.googlecode.com/svn/sch
ema/ehcache-spring"
   xmlns:mongo="http://www.springframework.org/schema/data/mongo"
   xsi:schemaLocation="
       http://www.springframework.org/schema/mvc
http://www.springframework.org/schema/mvc/spring-mvc.xsd
       http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans.xsd
       http://www.springframework.org/schema/context
http://www.springframework.org/schema/context/spring-context.xsd
       http://mybatis.org/schema/mybatis-spring
http://mybatis.org/schema/mybatis-spring.xsd
       http://www.springframework.org/schema/util
http://www.springframework.org/schema/util/spring-util.xsd
       http://www.springframework.org/schema/aop
http://www.springframework.org/schema/aop/spring-aop.xsd
       http://www.springframework.org/schema/task
http://www.springframework.org/schema/task/spring-task.xsd
http://ehcache-spring-annotations.googlecode.com/svn/schema/ehcache-spr
ing
http://ehcache-spring-annotations.googlecode.com/svn/schema/ehcache-spr
ing/ehcache-spring-1.1.xsd
       http://www.springframework.org/schema/data/mongo
http://www.springframework.org/schema/data/mongo/spring-mongo.xsd
   "
>
    <context:component-scan base-package="com.geowebframework" />

    <!-- Convention: automatically including all resources declared
under the folder META-INF/ inside files with '.beandef.xml' extension
-->
    <import resource="classpath*:META-INF/*.beandef.xml" />

    <!-- METADATASOURCE -->
    <bean id="metadataSource"
class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">
        <property name="driverClassName"
value="${jdbcmetadata.driverClassName}" />
        <property name="url" value="${jdbcmetadata.url}" />
        <property name="username" value="${jdbcmetadata.username}" />
        <property name="password" value="${jdbcmetadata.password}" />
        <property name="maxActive" value="${jdbcmetadata.maxActive}" />
```

```xml
        <property name="minIdle" value="${jdbcmetadata.minIdle}" />
        <property name="maxIdle" value="${jdbcmetadata.maxIdle}" />
        <property name="validationQuery"
value="${jdbcmetadata.validationQuery}"/>
    </bean>

    <!-- DATASOURCE -->
    <bean id="dataSource"
class="org.apache.commons.dbcp.BasicDataSource" destroy-method="close">
        <property name="driverClassName"
value="${jdbc.driverClassName}" />
        <property name="url" value="${jdbc.url}" />
        <property name="username" value="${jdbc.username}" />
        <property name="password" value="${jdbc.password}" />
        <property name="maxActive" value="${jdbc.maxActive}" />
        <property name="minIdle" value="${jdbc.minIdle}" />
        <property name="maxIdle" value="${jdbc.maxIdle}" />
        <property name="validationQuery"
value="${jdbc.validationQuery}"/>
    </bean>

    <!-- MARSHALL -->
    <mvc:annotation-driven>

        <mvc:path-matching suffix-pattern="true"/> <!-- issue #589 -->

        <mvc:message-converters register-defaults="true">
            <bean id="marshallingHttpMessageConverter"
class="org.springframework.http.converter.xml.MarshallingHttpMessageCon
verter">
                <property name="marshaller" ref="oxmMarshaller" />
                <property name="unmarshaller" ref="oxmMarshaller" />
            </bean>
        </mvc:message-converters>

    </mvc:annotation-driven>

    <bean id="oxmMarshaller"
class="org.springframework.oxm.xstream.XStreamMarshaller">
        <property name="autodetectAnnotations" value="true" />
    </bean>

    <!-- FILE UPLOAD -->
    <bean id="multipartResolver"
class="org.springframework.web.multipart.commons.CommonsMultipartResolv
er">

        <!-- one of the properties available; the maximum file size in
bytes -->
        <property name="maxUploadSize" value="2000000000" />
    </bean>
```

```
</beans>
```

Modifiche

E' stato rimosso il blocco *log4jInitialization*, non piu necessario con **log4j2**

```
    <!-- LOG4J -->
    <bean id="log4jInitialization"
class="org.springframework.beans.factory.config.MethodInvokingFactoryBean">
        <property name="targetClass"
value="org.springframework.util.Log4jConfigurer" />
        <property name="targetMethod" value="initLogging" />
        <property name="arguments">
            <list>
                <value>classpath:log4j.properties</value>
                <value>10000</value>
            </list>
        </property>
    </bean>
```

web.xml

Localizzazione: *WEB-INF/web.xml*

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
  xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd"
  version="3.1"
>

    <display-name>GeoWeb Manager</display-name>


    <welcome-file-list>
        <!--DEFAULT -->
        <welcome-file>index.html</welcome-file>

        <!-- SOLO EXITONE -->
        <!--<welcome-file>indexPortal.html</welcome-file>-->
```

```xml
        <!--SOLO CDE 3.5 -->
        <!--<welcome-file>gwindex.html</welcome-file>-->
    </welcome-file-list>


    <servlet>
        <servlet-name>dispatcher</servlet-name>
        <servlet-
class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
        <load-on-startup>1</load-on-startup>
        <!--<async-supported>true</async-supported> must be enable for
use websocket with mapguide -->
        <async-supported>true</async-supported>
        <!--
    It should to solve the problem:
    Request processing failed; nested exception is
org.springframework.web.multipart.MultipartException:
        Failed to parse multipart servlet request; nested exception is
org.apache.commons.fileupload.FileUploadException:
            UT000020: Connection terminated as request was larger than
10485760

    Sizes must to be expressed in byte (B)
    -->
        <multipart-config>
            <location>/tmp</location>
            <max-file-size>104857600</max-file-size><!--100MB-->
            <max-request-size>209715200</max-request-size><!--200MB-->
            <file-size-threshold>0</file-size-threshold>
        </multipart-config>
    </servlet>


    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>*.html</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>/rest/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>/services/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>/files/*</url-pattern>
    </servlet-mapping>
    <servlet-mapping>
```

```xml
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>/</url-pattern>
    </servlet-mapping>


    <!--HTML REPORT SERVELT STARTS-->
    <servlet>
        <servlet-name>ImageServlet</servlet-name>
        <servlet-class>net.sf.jasperreports.j2ee.servlets.ImageServlet</servlet-class>
        <async-supported>true</async-supported>
    </servlet>
    <servlet-mapping>
        <servlet-name>ImageServlet</servlet-name>
        <url-pattern>/jrImage</url-pattern>
    </servlet-mapping>
    <!--HTML REPORT SERVELT ENDS-->


    <jsp-config>
        <taglib>
            <taglib-uri>http://geowebframework.com/jsp/jstl/functions</taglib-uri>
            <taglib-location>/WEB-INF/gwutil-functions.tld</taglib-location>
        </taglib>

        <jsp-property-group>
            <url-pattern>*.jsp</url-pattern>
            <page-encoding>UTF-8</page-encoding>
        </jsp-property-group>
    </jsp-config>


    <listener>
        <listener-class>
            org.springframework.web.context.ContextLoaderListener
        </listener-class>
    </listener>


    <!--
 It's needed to have HttpServletRequest request available with
@Autowired;
 inside a generic XUserDetailsService that implements
UserDetailsService.
    example: SiteminderUserDetailsService.java
 -->
    <listener>
```

```xml
        <listener-class>
org.springframework.web.context.request.RequestContextListener
        </listener-class>
    </listener>


    <!-- Spring Security STARTS -->
    <context-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>
            <!-- TODO CONFIGURE, choose one between: -->
            <!-- /WEB-INF/spring-security.xml -->
            <!-- /WEB-INF/spring-security-keycloak.xml -->
            /WEB-INF/spring-security.xml
        </param-value>
    </context-param>

    <filter>
        <filter-name>springSecurityFilterChain</filter-name>
        <filter-class>
            org.springframework.web.filter.DelegatingFilterProxy
        </filter-class>
        <!--<async-supported>true</async-supported> must be enable for
use websocket whith mapguide, must stay before <init-param>  and after
<filter-class>-->
        <async-supported>true</async-supported>
        <init-param>
            <param-name>contextAttribute</param-name>
            <param-value>org.springframework.web.servlet.FrameworkServlet.CONTEXT.dispatcher</param-value>
        </init-param>
    </filter>

    <filter-mapping>
        <filter-name>springSecurityFilterChain</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <!-- Spring Security ENDS -->


    <!-- FORCE UTF-8 STARTS -->
    <!-- important that is at the beginning -->
    <filter>
        <filter-name>encodingFilter</filter-name>
        <filter-class>org.springframework.web.filter.CharacterEncodingFilter</filter-class>
        <async-supported>true</async-supported>
        <init-param>
            <param-name>encoding</param-name>
```

```xml
            <param-value>UTF-8</param-value>
        </init-param>
        <init-param>
            <param-name>forceEncoding</param-name>
            <param-value>true</param-value>
        </init-param>
    </filter>

    <filter-mapping>
        <filter-name>encodingFilter</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <!-- FORCE UTF-8 ENDS -->


    <!-- CORS STARTS-->
    <filter>
        <filter-name>customOptionsMethodFilter</filter-name>
        <filter-class>com.geowebframework.webclient.security.cors.CustomOptionsMethodFilter</filter-class>
    </filter>
    <filter-mapping>
        <filter-name>customOptionsMethodFilter</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>

    <filter>
        <filter-name>customCorsFilter</filter-name>
        <filter-class>com.geowebframework.webclient.security.cors.CustomCorsFilter</filter-class>
    </filter>
    <filter-mapping>
        <filter-name>customCorsFilter</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <filter>
        <filter-name>XSSFilter</filter-name>
        <filter-class>com.geowebframework.webclient.security.xss.CrossSiteScriptingFilter</filter-class>
    </filter>
    <filter-mapping>
        <filter-name>XSSFilter</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <!-- CORS ENDS-->
```

```xml
    <!-- MULTIPART FILTER STARTS-->
    <filter>
        <filter-name>MultipartFilter</filter-name>
        <filter-
class>org.springframework.web.multipart.support.MultipartFilter</filter
-class>
    </filter>
    <filter-mapping>
        <filter-name>MultipartFilter</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>
    <!-- MULTIPART FILTER ENDS-->


    <!-- [RP20_666_27] INFORMATION DISCLOSURE OF THE INFRASTRUCTURE -->
    <context-param>
        <param-name>com.sun.faces.sendPoweredByHeader</param-name>
        <param-value>false</param-value>
    </context-param>


    <session-config>
        <session-timeout>15</session-timeout>

        <!-- [RP20_666_14] INSECURE SESSION COOKIE -->
        <!-- The following 4 lines (cookie-config tag) must be
uncommented in production -->

        <!--
     <cookie-config>
        <http-only>true</http-only>
        <secure>true</secure>
     </cookie-config>
     -->
    </session-config>

</web-app>
```

Modifiche

Nel blocco sottostante si può scegliere quale spring-security utilizzare fra *spring-security.xml* e *spring-security-keycloak.xml*. Il default è spring-security.xml

```xml
...
    <context-param>
        <param-name>contextConfigLocation</param-name>
        <param-value
```

```
            <!-- TODO CONFIGURE, choose one between: -->
            <!-- /WEB-INF/spring-security.xml -->
            <!-- /WEB-INF/spring-security-keycloak.xml -->
            /WEB-INF/spring-security.xml
        </param-value>
    </context-param>
...
```

configuration.properties

Localizzazione:

- *[static-contents-folder]/conf/configuration.properties*
- *src/main/resources/configuration.properties*

Va utilizzato il template disponibile [qui](#) , integrandolo con i dati e le personalizzazioni del vecchio *configuration.properties*

log4j2.properties

Nelle versioni 4.6.X è stato introdotto **log4j2**, che sostituisce **log4j**.

Localizzazione:

- *[static-contents-folder]/conf/log4j2.properties*
- *src/main/resources/log4j2.properties*

Va utilizzato il template disponibile nella [guida](#), integrandolo con i dati del vecchio log4j.properties.

In particolare dovrebbe essere solo necessario riprendere i nomi dei file di log, e configurali nei blocchi:

```
appender.rolling.fileName = ${sys:catalina.home}/logs/geoweb_client.log
appender.rolling.filePattern = ${sys:catalina.home}/logs/geoweb_client-
%d{yyyy-MM-dd-HH-mm-ss}-%i.log
```

```
appender.rollingEmail.fileName = ${sys:catalina.home}/logs/gwMail.log
appender.rollingEmail.filePattern = ${sys:catalina.home}/logs/gwMail-
%d{yyyy-MM-dd-HH-mm-ss}-%i.log
```

```
appender.rollingAccess.fileName = ${sys:catalina.home}/logs/gwAccess.log
appender.rollingAccess.filePattern = ${sys:catalina.home}/logs/gwAccess-
%d{yyyy-MM-dd-HH-mm-ss}-%i.log
```

Eventualmente integrate ulteriori personalizzazione di progetto/commessa.

Migrazione metadati

Nella 4.6.0-SNAPSHOT la mappa 2D è stata resa configurabile con un xml nel webadmin (come le mappe BIM) (issue #563). A tale scopo è stato aggiunto il campo **xml_map_mode (stringa, maxLenght 4000)** alla tabella dei metadati **gwm_maps**. Quindi, tramite script del *gw_resource_deployer*, o manualmente va eseguito lo script:

```sql
ALTER TABLE [SCHEMA_NAME].GWM_MAPS ADD XML_MAP_MODE VARCHAR2(4000 BYTE);
```

I metadati che si riferiscono a versioni particolarmente vecchie possono essere importati dal **webadmin**, e venire automaticamente aggiustati, tramite gli appositi *converters* che si preoccupano di effettuare il passaggio da una versione all'altra.

Al momento non viene ancora gestito il popolamento automatico dell'xml. Utilizzare quindi il template seguente:

map_mode.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<mode name='default' label='Base Mode' icon='defaultMapMode.png'>
    <mainToolbar>
        <buttonGroup name='selection' />
        <buttonGroup name='navigation' />
        <buttonGroup name='info' />
        <buttonGroup name='refresh' />
        <buttonGroup name='report' />
        <buttonGroup name='googleStreetView' />
        <buttonGroup name='commandsTab' />
    </mainToolbar>
    <tabList>
        <tab type='layers' name='layers' label='' />
        <tab type='baseMaps' name='baseMaps' label='' />
        <tab type='edit' name='edit' label=''>
            <!--<tabAttribute name='cutCopyPasteToolbar' />-->
            <tabAttribute name='singleInsertToolbar' />
            <tabAttribute name='singleMultipleEditToolbar' />
        </tab>
        <!--<tab type='furniture' name='furniture' label='' />-->
    </tabList>
    <properties>
        <entry key="showScaleLineControl" value="true" />
        <entry key="showScaleControl" value="true" />
        <entry key="showMousePositionControl" value="true" />
        <entry key="startWithMapSwitcherOpened" value="true" />
    </properties>
</mode>
```

Il template va anche puntualmente configurato per decidere quali gruppi di bottoni, tab e controlli visualizzare e quali no. In precedenza queste configurazioni erano globali ed impostatabili dall'xml di

progetto. Tutti quei tag vanno considerati **deprecati**:

```
<mapSettings>
        <showScaleLineControl>true</showScaleLineControl>
        <showScaleControl>true</showScaleControl>
        <showMousePositionControl>true</showMousePositionControl>
        <showBaseMapChange>true</showBaseMapChange>
        <showLayersManagement>true</showLayersManagement>
        <showThematisms>true</showThematisms>
        <showMapEditing>true</showMapEditing>
        <startWithMapSwitcherOpened>true</startWithMapSwitcherOpened>
        ...
</mapSettings>
```

Note Migrazione

Download Webadmin war

Il download dell'applicazione gw-webadmin-X.Y.Z.war rilasciate è disponibile in artifactory a questo link:

http://artifactory.geowebframework.com/artifactory/gw-release/com/geowebframework/gw-webadmin/X.Y.Z/gw-webadmin-X.Y.Z.war

dove X.Y.Z è la versione di geoweb rischiesta.

Esempio webadmin versione 4.6.10
http://artifactory.geowebframework.com/artifactory/gw-release/com/geowebframework/gw-webadmin/4.6.10/gw-webadmin-4.6.10.war

Nel caso straordinario in cui si richiede una versione hotfix la struttura del link è la seguente:

http://artifactory.geowebframework.com/artifactory/gw-hotfix/com/geowebframework/gw-webadmin/X.Y.Z-HOTFIX/gw-webadmin-X.Y.Z-HOTFIX.war

http://artifactory.geowebframework.com/artifactory/gw-hotfix/com/geowebframework/gw-webadmin/4.6.11-HOTFIX/gw-webadmin-4.6.11-HOTFIX.war

Workflow

In presenza del modulo del workflow, *Activiti* provvederà ad aggiornare le tabelle **act_***. In particolare viene aggiornata l'informazione sulla version di activiti che passa da

```
5.17.0
```

a

```
5.23.0
```

.

Questo renderà inutilizzabili eventuali applicativi webapp che puntano allo stesso database, e che sono versioni precedenti alla 4.6.X

Preferiti del browser

La pagina di login è passata da **login.html** a **loginPage.html**. Nel caso in cui nei *preferiti*, il link sia salvato con parti oltrelo / successivo al contesto, come per esempio

```
[server]/[nome_progetto]/login.html
```

Modificare il link della voce dei preferiti, cosi:

```
[server]/[nome_progetto]/
```

File groovy

Con il passaggio alla versione 4.6.0 del framework, c'è stato un'aggiornamento di librerie. Vanno quindi allineati tutti i **file .groovy** che facevano riferimento alla vecchia versione.

In rosso i vecchi import, in verde i nuovi path per l' import.

log4j

Era affetta da varie falle di sicurezza. Adesso viene utilizzato log4j2.

private static final org.apache.log4j.Logger log = org.apache.log4j.Logger.getLogger("com.geowebframework.groovy");

private static final org.apache.logging.log4j.Logger log = org.apache.logging.log4j.LogManager.getLogger("com.geowebframework.groovy");

---

import org.apache.log4j.Logger;

…

private static final Logger log = Logger.getLogger("com.geowebframework.groovy");

---

import org.apache.logging.log4j.LogManager;

import org.apache.logging.log4j.Logger;

…

private static Logger log = LogManager.getLogger("com.geowebframework.groovy");

Inoltre ora log4j 2 viene gestito come parametro sia String che CharSequence, quindi nei .groovy potrebbe generarsi l'errore

```
2023-01-23 16:15:06,330 [ajp-nio-8019-exec-70] ERROR
com.geowebframework.dataservice.service.GwClassListService - Ambiguous
method overloading for method org.apache.logging.log4j.core.Logger#warn.
Cannot resolve which method to invoke for [null, class
java.lang.NullPointerException] due to overlapping prototypes between:
    [interface java.lang.CharSequence, class java.lang.Throwable]
    [interface org.apache.logging.log4j.message.Message, class
java.lang.Throwable]
    [interface org.apache.logging.log4j.util.MessageSupplier, class
java.lang.Throwable]
    [interface org.apache.logging.log4j.util.Supplier, class
java.lang.Throwable]
groovy.lang.GroovyRuntimeException: Ambiguous method overloading for method
org.apache.logging.log4j.core.Logger#warn.
Cannot resolve which method to invoke for [null, class
java.lang.NullPointerException] due to overlapping prototypes between:
    [interface java.lang.CharSequence, class java.lang.Throwable]
    [interface org.apache.logging.log4j.message.Message, class
java.lang.Throwable]
    [interface org.apache.logging.log4j.util.MessageSupplier, class
java.lang.Throwable]
    [interface org.apache.logging.log4j.util.Supplier, class
java.lang.Throwable]
```

in casi del tipo:

```
log.warn(e.getMessage(), e);
```

Da sostituire con:

```
log.warn((String) e.getMessage(), e);
```

jackson

import org.codehaus.jackson.map.annotate.JsonSerialize.Inclusion;

import com.fasterxml.jackson.annotation.JsonInclude.Include;

---

import org.codehaus.jackson.map.JsonMappingException;

import com.fasterxml.jackson.databind.JsonMappingException;

---

import org.codehaus.jackson.type.TypeReference;

import com.fasterxml.jackson.core.type.TypeReference;

---

import org.codehaus.jackson.JsonGenerationException;

import com.fasterxml.jackson.core.JsonGenerationException;

---

import org.codehaus.jackson.map.ObjectMapper;

import com.fasterxml.jackson.databind.ObjectMapper;

JTS

in generale:

import com.vividsolutions.jts.*

import org.locationtech.jts.*

---

import com.vividsolutions.jts.geom.Geometry;

import org.locationtech.jts.geom.Geometry;

---

import org.locationtech.jts.geom.Coordinate;

import org.locationtech.jts.geom.Geometry;

import org.locationtech.jts.geom.GeometryCollection;

import org.locationtech.jts.geom.GeometryFactory;

import org.locationtech.jts.geom.LineString;

import org.locationtech.jts.geom.LinearRing;

import org.locationtech.jts.geom.MultiLineString;

import org.locationtech.jts.geom.MultiPoint;

---

import org.locationtech.jts.geom.MultiPolygon;

import org.locationtech.jts.geom.Point;

import org.locationtech.jts.geom.Polygon;

import org.locationtech.jts.geom.Coordinate;

import org.locationtech.jts.geom.Envelope;

Report

La libreria di JasperReports **6.20.1**. E' consigliabile utilizzare per la gestione dei file *.jasper* l'ultimo Jaspersoft Studio.

Il passaggio a **java 11**, ha portato con se il definito superamento di una serie di classi che erano da tempo state marchiate come deprecate, e che adesso sono inutilizzabili.

In particolare, in alcune implementazioni di report, posoono essere stati utilizzati costrutti non piu funzionanti, in particolare che fanno riferimento a metodi della **class java.util.Date**.

Vanno quindi allineati tutti i **file .jrxml** che facevano riferimento a metodi non più supportati.

In rosso esempi di vecchio codice, in verde il nuovo equivalente.

java.util.Date format()

risorse utili:

- java-8-date-time-intro
- java-8-how-to-format-localdatetime

<textFieldExpression><![CDATA["PRENOTAZIONI DI OGGI " + new java.util.Date().format('EEEE dd-MM-yyyy')]]></textFieldExpression>

<textFieldExpression><![CDATA["PRENOTAZIONI DI OGGI " + java.time.LocalDateTime.now().format(java.time.format.DateTimeFormatter.ofPattern("EEEE dd-MM-yyyy"))]]></textFieldExpression>

From:
https://wiki.geowebframework.com/ - **GeowebFramework**

Permanent link:
https://wiki.geowebframework.com/doku.php?id=gwusermanual:migration_to_4_6_x&rev=1704279259

Last update: **2024/01/03 11:54**