

## web.xml 및 context-egovuserdetailshelper.xml의 설정

소스 코드에 포함되어 있는 web.xml은 모든 컴포넌트에 공통으로 적용되는 사항만 포함되어 있는 상태이며 추가 설정 사항이 필요한 컴포넌트는 배포파일에 부가 사항을 담은 파일이 web.xml과 함께 제공된다. 이러한 컴포넌트를 사용하는 경우에는 해당 파일의 내용이 web.xml에 포함되도록 수정해야 정상적으로 실행될 수 있다. 이와 같은 설정이 필요한 컴포넌트는 아래와 같다.

모든 설정은 web.xml과 context-egovuserdetailshelper.xml(src/main/resources/egovframework/spring/com/context-egovuserdetailshelper.xml)에 주석처리 된 상태로 배포되며 해당 컴포넌트(로그인, 권한관리 컴포넌트 등)를 사용할 경우에 주석을 해제하여 사용하면 된다.

단 해당 컴포넌트가 설치되지 않은 상태에서 주석을 해제하면 ClassNotFoundException이 발생할 수 있다. 또한 web.xml의 필터는 주석문에 붙여진 번호 순서대로 있어야 하며 순서가 바뀌는 경우 오작동 할 수 있다.

- 공통컴포넌트 환경설정
  - web.xml 및 context-egovuserdetailshelper.xml의 설정
    - 1. 로그인정책관리 컴포넌트를 포함하는 경우
    - 2. 권한관리 관련 컴포넌트를 포함하는 경우(스프링 시큐리티)
    - 3. Spring Security를 이용하지 않고 세션을 이용하여 인증을 유지하는 경우
  - Data Source 설정

### 1. 로그인정책관리 컴포넌트를 포함하는 경우

web.xml의 아래 부분을 주석해제해서 사용하면 된다.

```
<filter>
  <filter-name>LoginPolicyFilter</filter-name>
  <filter-class>egovframework.com.uat.uap.filter.EgovLoginPolicyFilter</filter-class>
</filter>
<filter-mapping>
  <filter-name>LoginPolicyFilter</filter-name>
  <url-pattern>/uat/uia/actionLogin.do</url-pattern>
</filter-mapping>
```

### 2. 권한관리 관련 컴포넌트를 포함하는 경우(스프링 시큐리티)

스프링 시큐리티관련 설정이 반드시 포함되어야 한다. context-security.xml/파일은 src/main/resources/egovframework/spring/com 폴더 아래 위치해야 한다.

1) context-egovuserdetailshelper.xml 수정 : 아래와 같이 egovUserDetailsSecurityService 빈 설정의 주석을 해제한 후 egovUserDetailsHelper 빈의 egovUserDetailsService 프러퍼티에 egovUserDetailsSecurityService 빈을 등록하면 된다.

```
<!-- 인증된 유저의 LoginVO, 권한, 인증 여부를 확인 할 수있는 서비스 클래스-->
<bean id="egovUserDetailsHelper" class="egovframework.com.cmm.util.EgovUserDetailsHelper">
  <property name="egovUserDetailsService">
    <!-- 원하는 bean id를 아래에 ref 태그에 적용한다 -->
    <ref bean="egovUserDetailsSecurityService" />
  </property>
</bean>

<!-- 0. 세션/시큐리티를 제외한 테스트위한 인증(더미)을 사용하는 빈 -->
<!--<bean id="egovTestUserDetailsService"
class="egovframework.com.cmm.service.impl.EgovTestUserDetailsServiceImpl"/>-->

<!-- 1. 세션을 이용한 인증을 사용하는 빈 -->
<!--<bean id="egovUserDetailsService"
class="egovframework.com.cmm.service.impl.EgovUserDetailsServiceImpl"/>-->

<!-- 2. 시큐리티를 이용한 인증을 사용할 빈 -->
<bean id="egovUserDetailsSecurityService"
class="egovframework.com.sec.ram.service.impl.EgovUserDetailsSecurityServiceImpl"/>
```

2) web.xml 수정 : 아래 설정 부분의 주석을 해제 하면 된다. EgovSpringSecurityLoginFilter 필터의 경우 로그인 인증이 실패할 경우 사용자에게 되돌려지 화면을 파라미터로 입력받고 있다. loginURL 파라미터값에 로그인 실패시에 리턴될 url경로를 입력하면 된다.

```
<!-- 1. Spring Security 필터 설정-->
<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>

<listener>
  <listener-class>org.springframework.security.ui.session.HttpSessionEventPublisher</listener-class>
</listener>

<!-- 2. Spring Security 로그아웃 설정을 위한 필터 설정 -->
<filter>
  <filter-name>EgovSpringSecurityLogoutFilter</filter-name>
  <filter-class>egovframework.com.sec.security.filter.EgovSpringSecurityLogoutFilter</filter-class>
</filter>
<filter-mapping>
  <filter-name>EgovSpringSecurityLogoutFilter</filter-name>
```

```

<url-pattern>/uat/uia/actionLogout.do</url-pattern>
</filter-mapping>

<!-- 2. Spring Security 로그인 설정을 위한 필터 설정 -->

<filter>
<filter-name>EgovSpringSecurityLoginFilter</filter-name>
<filter-class>egovframework.com.sec.security.filter.EgovSpringSecurityLoginFilter</filter-class>
<init-param>
<description>로그인 실패시 반환 될 URL 설정</description>
<param-name>loginURL</param-name>
<param-value>/uat/uia/egovLoginUsr.do</param-value>
</init-param>
</filter>
<filter-mapping>
<filter-name>EgovSpringSecurityLoginFilter</filter-name>
<url-pattern>*.do</url-pattern>
</filter-mapping>

```

### 3. Spring Security를 이용하지 않고 세션을 이용하여 인증을 유지하는 경우

1) context-egovuserdetailshelper.xml 수정 : 아래와 같이 egovUserDetailsService빈 설정의 주석을 해제한 후 egovUserDetailsHelper 빈의 egovUserDetailsService 프러퍼티에 egovUserDetailsService빈을 등록하면 된다.

```

<!-- 인증된 유저의 LoginVO, 권한, 인증 여부를 확인 할 수있는 서비스 클래스-->
<bean id="egovUserDetailsHelper" class="egovframework.com.cmm.util.EgovUserDetailsHelper">
<property name="egovUserDetailsService">
<!-- 원하는 bean id를 아래에 ref 태그에 적용한다 -->
<ref bean="egovUserDetailsService" />
</property>
</bean>

<!-- 0. 세션/시큐리티를 제외한 테스트위한 인증(더미)을 사용하는 빈 -->
<!--<bean id="egovTestUserDetailsService"
class="egovframework.com.cmm.service.impl.EgovTestUserDetailsServiceImpl"/>-->

<!-- 1. 세션을 이용한 인증을 사용하는 빈 -->
<bean id="egovUserDetailsService"
class="egovframework.com.cmm.service.impl.EgovUserDetailsServiceImpl"/>

<!-- 2. 시큐리티를 이용한 인증을 사용할 빈 -->
<!--<bean id="egovUserDetailsSecurityService"
class="egovframework.com.sec.ram.service.impl.EgovUserDetailsSecurityServiceImpl"/>-->

```

2) web.xml 수정 : 수정사항 없이 디폴트 설정대로 사용하면 된다.

## Data Source 설정

context-datasource.xml의 DB관련 설정은 모두 변수처리되어 있고 실제 설정 내용은 globals.properties(src/main/resources/egovframework/egovProps/globals.properties)파일을 수정하면 된다. 초기 상태는 mysql DB로 설정되어 있으므로 DbType에 실제 사용하는 DB를 설정하고 해당하는 DB의 DriverClassName, Url을 설정한다. 마지막으로 UserName, Password 을 설정하면 된다.

```

# DB서버 타입(ORACLE, MYSQL) - datasource 및 sqlMap 파일 지정에 사용됨
Globals.DbType = mysql

# 위저드 사용시 데이터베이스 관련 설정을 불러옴
#mysql
Globals.DriverClassName=net.sf.log4jdbc.DriverSpy
Globals.Url=jdbc:mysql://127.0.0.1:1621/egovfrm

#oracle
Globals.DriverClassName=oracle.jdbc.driver.OracleDriver
Globals.Url=jdbc:oracle:thin:@127.0.0.1:1521:egovfrm

#Altibase
Globals.DriverClassName=Altibase.jdbc.driver.AltibaseDriver
Globals.Url=jdbc:Altibase://127.0.0.1:1721/egovfrm?encoding=UTF-8

#Tibero
Globals.DriverClassName=com.tmax.tibero.jdbc.TbDriver
Globals.Url=jdbc:tibero:thin:@127.0.0.1:1821:egovfrm

#DB 계정 ID/PW 설정
Globals.UserName = ID
Globals.Password = PW

```

사용하는 데이터베이스를 변경하였을 경우에는 pom.xml의 jdbc 드라이버에 대한 Dependency를 설정해야 한다. 3rd party library의 경우에는 직접 Local Repository에 library를 넣어 주거나 Nexus서버를 구성하여 다운로드 받을 수 있다. Nexus에 대한 자세한 설정은 개발환경의 [Nexus](#)페이지를 참조한다.

<EXAMPLE>

```

<!-- mysql -->
<dependency>
<groupId>mysql</groupId>
<artifactId>mysql-connector-java</artifactId>
<version>5.1.17</version>
</dependency>
<!-- oracle 10g driver -->
<dependency>
<groupId>ojdbc</groupId>
<artifactId>ojdbc</artifactId>
<version>14</version>

```

```
</dependency>
<!-- altibase driver -->
<dependency>
  <groupId>altibase</groupId>
  <artifactId>altibase-jdbc-driver</artifactId>
  <version>5.1.3.18</version>
</dependency>
<!-- tiberero driver -->
<dependency>
  <groupId>com.tmax.tiberero</groupId>
  <artifactId>tiberero-jdbc</artifactId>
  <version>3.0</version>
</dependency>
```