

# openFWPA Internacional

# openEUG

# Manual de Instalación y Configuración

(01. ManualDeInstalaciónYConfiguración\_openEUG\_20111230\_v1.0)



Proyecto	OpenFWPA Internacional openEUG	Página 2 de 9
Estado	Definitivo	
Documento	Manual de Instalación y Configuración	

## <u>ÍNDICE</u>

1.	CONTROL DEL DOCUMENTO	.3
	1.1. Información general	.3
	1.2. Histórico de revisiones	.3
	1.3. Estado del documento	.3
2.	INTRODUCCIÓN.	.4
3.	INSTALACIÓN	.4
	3.1. GENERACIÓN DE LAS BASES DE DATOS	.4
	3.2. COMPILACION	.5
4.	DESPLIEGUE Y ARRANQUE.	.6
5.	CONFIGURACIÓN	.7
6.	Anexo. Secuencias en MySQL	.9



Proyecto	OpenFWPA Internacional openEUG	Página 3 de 9
Estado	Definitivo	
Documento	Manual de Instalación y Configuración	

## **1. CONTROL DEL DOCUMENTO**

#### 1.1. Información general

Título	Manual de Instalación y Configuración	
Creado por:	Rubén Granda Velasco	
A revisar por:	Consultores Senior: Juan José Parada Vales e Ignacio Álvarez Valdeón	
A aprobar por:	Jefe de Proyecto: Joaquín Fernández Juárez	

#### **1.2. Histórico de revisiones**

Versión	Fecha	Autor	Observaciones
1.0	30/12/2011	Ruben Granda Velasco	

#### 1.3. Estado del documento

Versión	Estado	Fecha
1.0	Definitivo	30/12/2011



Proyecto	OpenFWPA Internacional openEUG	Página 4 de 9
Estado	Definitivo	
Documento	Manual de Instalación y Configuración	

## 2. INTRODUCCIÓN.

Este documento tiene como objeto servir de guía para la instalación, configuración y arranque de la aplicación de ejemplo del Framework de Tramitación Electrónica de Expedientes: openEUG

### **3. INSTALACIÓN.**

Inicialmente será necesario descargar los fuentes del proyecto, alojados en la siguiente direccion <u>https://joinup.ec.europa.eu/svn/openfwpa/openEUG</u>

#### **3.1. GENERACIÓN DE LAS BASES DE DATOS.**

El openEUG utiliza dos bases de datos distintas para su funcionamiento.

La primera recoge la esquematización de los procesos administrativos. Basándose en el "Procedimiento Administrativo Común" (PAC) se ha recogido toda la información en cuanto al procedimiento, fases, actos administrativos, flujos de tramitación, etc. Si bien esta base de datos es capaz de almacenar la información de cualquiera otros procedimientos. Esta base de datos se le conoce como el "catalogo".

Toda la información relativa a los expedientes, y las acciones que se realizan con ellos se almacenan de forma independiente en otra base de datos denominada "expediente".

Para la generación de estas bases de datos:

- Disponer de una base de datos MySQL en el equipo (probado en versión 5.5.13).
- Crear esquema "catalogo" y un usuario

```
>mysql -u root -p;
mysql > CREATE DATABASE catalogo;
mysql > GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, CREATE
TEMPORARY TABLES, DROP, INDEX, ALTER ON catalogo.* TO
'catalogo'@'%' IDENTIFIED BY 'prueba';
```



• Crear esquema "expediente" y un usuario.

```
>mysql -u root -p;
mysql > CREATE DATABASE expediente;
mysql > GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, CREATE
TEMPORARY TABLES, DROP, INDEX, ALTER ON expediente.* TO
'expediente'@'%' IDENTIFIED BY 'prueba';
```

• Se dispone de las querys para construir y rellenar con los datos necesarios para arrancar la aplicación. Se utiliza Maven para ello.

Nota: Puede ser necesario modificar las propiedades del openEUG/trunk/fwpa-et-sql/pom.xml

```
> cd openEUG/trunk/fwpa-et-sql;
> mvn -Pmysql,run-mysql clean install
```

• Simular las secuencias. Ver Anexo. Secuencias en MySQL.

Si todo ha ido bien, se podrá comprobar que se han creado las tablas necesarias en "expediente" y las tablas en el "catalogo" con los datos, al menos, del procedimiento "Procedimiento Administrativo Común" en su versión 3.1

#### **3.2. COMPILACION**

Será necesario compilar previamente todos los módulos.

En este primero se utiliza un perfil para que se utilicen las querys específicas para MySQL

```
> cd openEUG/trunk/fwpa-et-core;
```

```
> mvn -Pmysql clean install
```

```
> cd openEUG/trunk/fwpa-et-dav;
```

```
> mvn clean install
```



Proyecto	OpenFWPA Internacional openEUG	Página 6 de 9
Estado	Definitivo	
Documento	Manual de Instalación y Configuración	

> cd openEUG/trunk/fwpa-et-proxy;

```
> mvn clean install
```

## 4. DESPLIEGUE Y ARRANQUE.

Para el despliegue de la aplicación se realiza mediante maven y plugins de éste, destacando el *tomcat-maven-plugin* y el *maven-soapui-plugin*. Lo que nos permite arrancar la aplicación sin necesidad de tener instalado software adicional, como contenedor de aplicaciones, etc.

La secuencia de comandos seria

```
> cd openEUG/trunk/sampleapp-et/sampleapp-et-war;
> mvn -Pmysql,soapui,tomcat clean install
```

Una vez haya terminado, al final de los mensajes en la consola deberá aparecer

```
04-ene-2012 9:54:52 org.apache.catalina.startup.Catalina start
INFO: Server startup in 13168 ms
```

Lo que nos indicará en la aplicación está desplegada. Para comprobarlo, en el navegador:

```
http://localhost:8888/pacv32
```

Login: *admin* Password: *admin* 

También se puede comprobar la lista de los mocks<sup>\*</sup> de los servicios que están disponibles al utilizar el perfil *-Psoapui* 

http://localhost:8088

(\*) Se llaman **objetos simulados** (mock, *mock object*, objetos de pega) a los objetos que imitan el comportamiento de objetos reales de una forma controlada.



Proyecto	OpenFWPA Internacional openEUG	Página 7 de 9
Estado	Definitivo	
Documento	Manual de Instalación y Configuración	

## 5. CONFIGURACIÓN.

Durante el proceso despliegue se habrán copiado los ficheros de configuración (.properties) para la aplicación de ejemplo.

Se ubicaran, bajo la ruta *c:/config/pacv32* tanto la configuración (*/config*) como los logs (*/logs*) (en Linux: */opt/oraias/WEBAPPS/pacv32*)

La lista de los ficheros que debieran aparecer en c:/config/pacv32/config

Nombre	Descripción
asturcon.properties	Datos de conexión con asturcon XXI
asturconConstants.properties	Constantes
BDTercerosBusIntegProxy.properties	Datos de conexión con M.C. de Terceros
ceBus.properties	Datos de conexión con M.C. "Cambio de Estado-
creadorDocumentos.properties	Datos de conexión con el Gestor Documental
docAportada.properties	Configuraciones de la docAportada
ehcache.xml	Configuraciones de EHCACHE
estadoCivil.properties	Constantes
GenericosBusIntegProxy.properties	Datos de conexión con M.C. de Genéricos
isBus.properties	Datos de conexión con M.C. de "Inicio Solicitud"
jass-config.xml	Configuración de la autenticación
log4j.xml	Configuración del log4j
mice.properties	Datos de conexión con el M.I. de Cambio de Estado
mipe.properties	Datos de conexión con el M.I. de Publicación de Expediente
numExpediente.properties	Datos de conexión con el servicio de numeración de expedientes
pacConstants.properties	Constantes
pacv32.properties	Configuración app



Proyecto	OpenFWPA Internacional openEUG	Página 8 de 9
Estado	Definitivo	
Documento	Manual de Instalación y Configuración	

Nombre	Descripción
princast-security-rules.xml	Configuración de la seguridad
SicresProxy.properties	Datos de conexión con Sicres
sms.properties	Datos de conexión con M.C. de SMS
tipoCanal.properties	Constantes
webdav.properties	Configuración webdav
wsinterop.properties	Datos de conexión con M.C. Interoperabilidad

Estos ficheros ya disponen de lo necesario para la probar la aplicación, por lo que no sería necesario modificarlos inicialmente.

**Nota**: Si durante el proceso de despliegue, por cualquier motivo, no se hubiera realizado la copia se pueden encontrar en la ruta *openEUG\trunk\sampleapp-et\sampleapplapp-et\sampleapp-et\sampleapp-et\sampleapp-et\sampleappla* 



### 6. Anexo. Secuencias en MySQL

Las querys del openEUG son originalmente usadas en bases de datos oracle. Posteriormente han sido transformadas para poder ser utilizadas con MySQL.

Las secuencias características de Oracle no existen en MySQL, por lo que se necesita un mecanismo para simularlas. La primera parte es la creación de una tabla para contenerlas (que ya está generada). Solo faltaría añadir una función, obtener el siguiente valor de la secuencia y un procedimiento para cambiarla, si fuera necesario.

```
drop function if exists getNextSeq;
delimiter //
create function getNextSeq(sSeqName varchar(50)) returns
                                                                 int.
unsigned
begin
   declare nLast_val int;
    set nLast_val = (select seq_val
                          from _sequence
                          where seq name = sSeqName);
    if nLast val is null then
       set \overline{nLast} val = 1;
        insert into sequence (seq name, seq val)
        values (sSeqName, nLast Val);
    else
        set nLast val = nLast val + 1;
        update sequence set seq val = nLast val
        where seq name = sSeqName;
    end if;
    return nLast val;
end;//
drop procedure if exists sp setSeqVal;
create procedure setSeqVal(sSeqName varchar(50), nVal int unsigned)
begin
   if (select count(*) from sequence where seq name = sSeqName) =
0 then
        insert into sequence (seq name, seq val)
        values (sSeqName,nVal);
    else
        update _sequence set seq_val = nVal
        where seq_name = sSeqName;
    end if;
end//
```