

## **Máster Interuniversitario en Seguridad de las TIC (MISTIC)**

### **PEC 3 TFM UOC**

#### **Estudio de la Integración del estándar OKI OSID con Learning Tools Interoperability (LTI versiones 1.1.1 y 2.0)**

**Convocatoria: enero**

**Profesor: Antoni Bertran Bellido**

**Curso 2013-2014**

**Alumno: José María de Córdoba Zea [josedcz@uoc.edu](mailto:josedcz@uoc.edu)**



## ÍNDICE

### Índice

Implementación.....	5
Instalación de Tomcat en un Mac OS 10.9.1 Mavericks.....	5
Configuración del entorno de desarrollo .....	5
Creación de la Base de Datos.....	6
Archivo de Configuración del enganche con la Base de Datos:.....	9
Configuración de puertos y seguridad.....	10
Manual de la aplicación.....	11

## ÍNDICE DE FIGURAS

### Índice de Figuras

Figura 1: Esquema de la Base de datos de LTI Tool Provider.....	9
Figura 2: Configuración Dynamic Domain Name System (DDNS) en el router.....	10
Figura 3: Vista de configuración de herramientas consumidoras.....	12
Figura 4: Vista de ejecución de la herramienta en Moodle.....	12
Figura 5: Configuración de Tool Provider con Clave y Secreto en Moodle.....	13

## **IMPLEMENTACIÓN**

### **Implementación**

En este capítulo se describe la forma de llevar a cabo la implementación de una aplicación LTI que permita ser integrada en una aplicación que soporte el protocolo, por ejemplo un Moodle.

#### ***Instalación de Tomcat en un Mac OS 10.9.1 Mavericks***

Seguir los pasos de: <http://wolfpaulus.com/journal/mac/tomcat7/> :

- Descargar Tomcat
  - get <http://tomcat.apache.org/download-70.cgi>
- Moverlo a la carpeta /usr/local
  - sudo mv apache-tomcat-7.0.50/ /usr/local/
- Crear un alias para ubicar fácilmente Tomcat
  - sudo ln -s /usr/local/apache-tomcat-7.0.50/ /Library/Tomcat
- Dar permisos al usuario actual
  - sudo chown -R jose /Library/Tomcat
- Dar permisos de ejecución usuario actual
  - sudo chmod +x /Library/Tomcat/bin/\*.sh
- Lanzar Tomcat
  - /Library/Tomcat/bin/startup.sh

### ***Configuración del entorno de desarrollo***

La aplicación también se puede lanzar desde el propio de desarrollo el tomcat, que quizá sea más cómodo para el desarrollo. Sin embargo voy a explicar cómo sería:

Ahora se procede a poner disponible la aplicación LtiToolProvider-1.1 a partir de la cual se hará la implementación, añadiéndole una adaptación con la parte de gestión de consumidores disponible en la aplicación rating. De esta última sólo se utiliza la parte de admin, el css que necesita la pantalla de configuración, las imágenes de la carpeta imágenes (no todas) y las clases Config y Db, que se pueden incluir por ejemplo en el propio paquete org.oscelot.lti.tp.

Se crea en IDE de desarrollo por ejemplo NetBeans o Eclipse con soporte para JEE

## IMPLEMENTACIÓN

Hay que satisfacer las siguientes dependencias de las bibliotecas (se incluye el directorio /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/ en el path o se incluyen directamente en las librerías del proyecto):

- `servlet-api` (si se usa tomcat 7.0 no es necesaria)
- [commons-httpclient](#) (commons-httpclient-3.0.1.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/ )
- [jdom](#) (cp -R jdom/\*.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/) tiene que ser la versión 1.1.3
- [oauth-core.jar](#) (ya estaba cp oauth-core-20100601.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/)
- [commons-codec](#) (cp commons-codec-1.7.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/)
- [gson](#) (google-gson-2.2.4/\*.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/)
- [LtiToolProvider](#) (cp LtiToolProvider-1.1.01.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/)
- [mysql-connector-java](#) (cp -R mysql-connector-java-5.1.28/\*.jar /Library/Tomcat/webapps/Rating-1.1.0/WEB-INF/lib/)

Por otro lado habría que crear un servidor apache tomcat 7.0 con el Asistente y luego arrancarlo.

### ***Creación de la Base de Datos***

Se ha instalado una instancia de mysql (bastaría por ejemplo con tener un XAMPP). Se crea la BD, el usuario y las tablas (en mysql):

```
create database uoc;

CREATE USER 'uoc'@'localhost' IDENTIFIED BY 'SnFbL6p4crUCt9ue';

GRANT USAGE ON * . * TO 'uoc'@'localhost' IDENTIFIED BY 'SnFbL6p4crUCt9ue'
WITH MAX_QUERIES_PER_HOUR 0 MAX_CONNECTIONS_PER_HOUR 0
MAX_UPDATES_PER_HOUR 0 MAX_USER_CONNECTIONS 0 ;

GRANT ALL PRIVILEGES ON `uoc` . * TO 'uoc'@'localhost';

CREATE TABLE lti_consumer (
  consumer_key varchar(255) NOT NULL,
  name varchar(45) NOT NULL,
  secret varchar(32) NOT NULL,
```

## **IMPLEMENTACIÓN**

```
Iti_version varchar(12) DEFAULT NULL,  
consumer_name varchar(255) DEFAULT NULL,  
consumer_version varchar(255) DEFAULT NULL,  
consumer_guid varchar(255) DEFAULT NULL,  
css_path varchar(255) DEFAULT NULL,  
protected tinyint(1) NOT NULL,  
enabled tinyint(1) NOT NULL,  
enable_from datetime DEFAULT NULL,  
enable_until datetime DEFAULT NULL,  
last_access date DEFAULT NULL,  
created datetime NOT NULL,  
updated datetime NOT NULL,  
PRIMARY KEY (consumer_key)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
CREATE TABLE Iti_context (  
    consumer_key varchar(255) NOT NULL,  
    context_id varchar(255) NOT NULL,  
    Iti_context_id varchar(255) DEFAULT NULL,  
    Iti_resource_id varchar(255) DEFAULT NULL,  
    title varchar(255) NOT NULL,  
    settings text,  
    primary_consumer_key varchar(255) DEFAULT NULL,  
    primary_context_id varchar(255) DEFAULT NULL,  
    share_approved tinyint(1) DEFAULT NULL,  
    created datetime NOT NULL,  
    updated datetime NOT NULL,  
    PRIMARY KEY (consumer_key, context_id)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
CREATE TABLE Iti_user (  
    consumer_key varchar(255) NOT NULL,  
    context_id varchar(255) NOT NULL,  
    user_id varchar(255) NOT NULL,  
    Iti_result_sourcedid varchar(255) NOT NULL,  
    created datetime NOT NULL,  
    updated datetime NOT NULL,  
    PRIMARY KEY (consumer_key, context_id, user_id)
```

## **IMPLEMENTACIÓN**

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE lti_nonce (  
  consumer_key varchar(255) NOT NULL,  
  value varchar(32) NOT NULL,  
  expires datetime NOT NULL,  
  PRIMARY KEY (consumer_key, value)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
CREATE TABLE lti_share_key (  
  share_key_id varchar(32) NOT NULL,  
  primary_consumer_key varchar(255) NOT NULL,  
  primary_context_id varchar(255) NOT NULL,  
  auto_approve tinyint(1) NOT NULL,  
  expires datetime NOT NULL,  
  PRIMARY KEY (share_key_id)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
ALTER TABLE lti_context  
  ADD CONSTRAINT lti_context_consumer_FK1 FOREIGN KEY (consumer_key)  
  REFERENCES lti_consumer (consumer_key);
```

```
ALTER TABLE lti_context  
  ADD CONSTRAINT lti_context_context_FK1 FOREIGN KEY  
  (primary_consumer_key, primary_context_id)  
  REFERENCES lti_context (consumer_key, context_id);
```

```
ALTER TABLE lti_user  
  ADD CONSTRAINT lti_user_context_FK1 FOREIGN KEY (consumer_key,  
  context_id)  
  REFERENCES lti_context (consumer_key, context_id);
```

```
ALTER TABLE lti_nonce  
  ADD CONSTRAINT lti_nonce_consumer_FK1 FOREIGN KEY (consumer_key)  
  REFERENCES lti_consumer (consumer_key);
```

```
ALTER TABLE lti_share_key  
  ADD CONSTRAINT lti_share_key_context_FK1 FOREIGN KEY  
  (primary_consumer_key, primary_context_id)
```



## IMPLEMENTACIÓN

REFERENCES lti\_context (consumer\_key, context\_id);

El esquema de la base de datos sería:

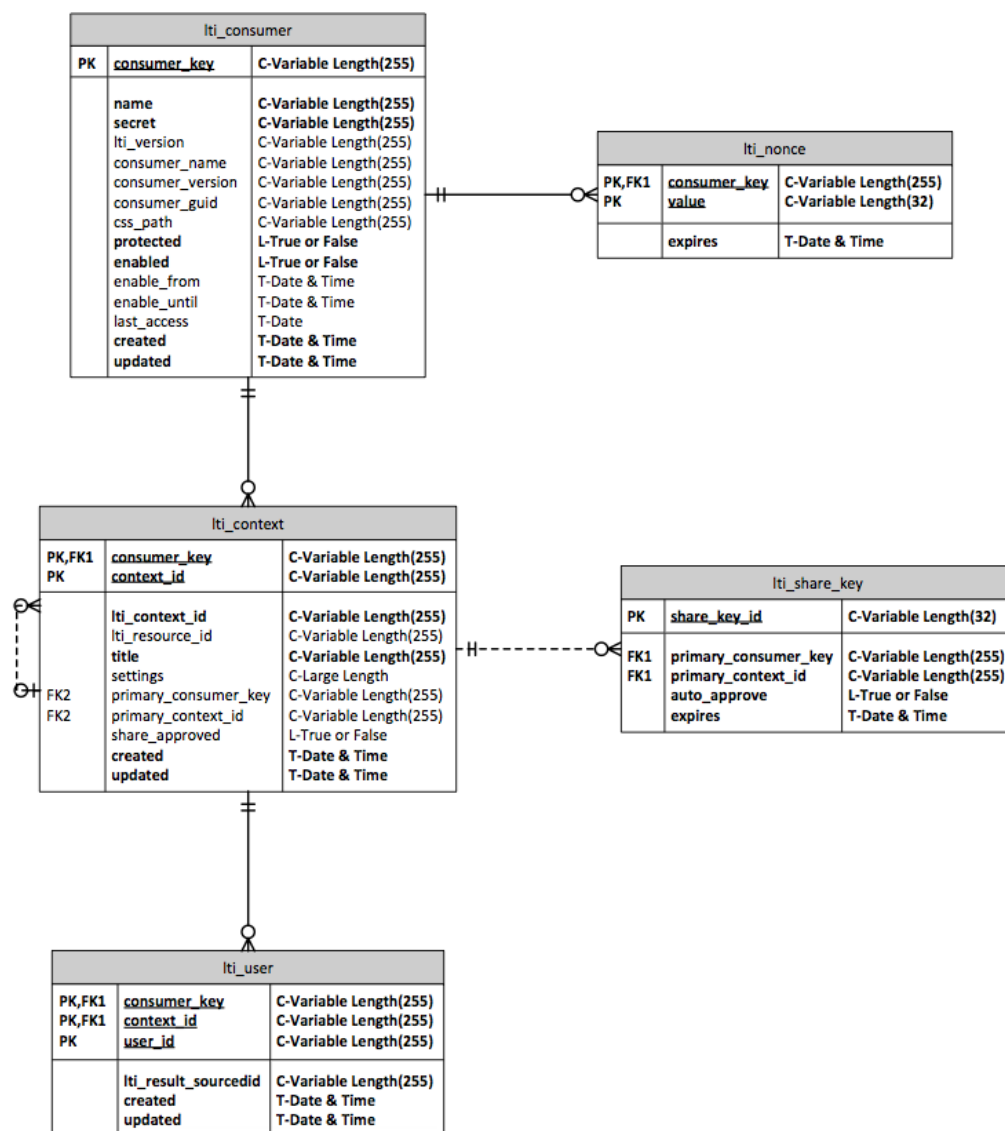


Figura 1: Esquema de la Base de datos de LTI Tool Provider

## Archivo de Configuración del enganche con la Base de Datos:

Habría que configurar el archivo Config.java así:

```

    /// Application setting

    public static final String APP_NAME = "TMF_UOC";
    
```

## IMPLEMENTACIÓN

```
/// Database connection settings

public static final String DB_NAME = "jdbc:mysql://localhost/uoc"; //
jdbc:mysql://localhost/MyDb"

public static final String DB_USERNAME = "uoc";

public static final String DB_PASSWORD = "SnFbL6p4crUCt9ue";

public static final String DB_TABLENAME_PREFIX = "";
```

### **Configuración de puertos y seguridad**

Una vez habilitada la herramienta hay que ponerla disponible al exterior y si nos encontramos detrás de una infraestructura con una red local y contamos con una ip dinámica, se puede proceder así:

- Abrir puerto 80 en el servidor
  - En Mac OS bastaría con deshabilitar el firewall
- Redireccionar el puerto 8080, el que usa tomcat al 80 en el servidor
  - En el terminal con la instrucción: `ipfw add 100 fwd 127.0.0.1,8080 tcp from any to any 80 in`
- Abrir puerto 80 en el router para la ip del servidor
  - Añadir la regla a tal efecto
- Habilitar mapeo de una dirección dinámica ip con una url estática con un servicio para esta función, por ejemplo dyndns y que se ejecute el cliente para realizar la actualización cuando se cambie la ip.

**Dynamic Domain Name System (DDNS)**

**DDNS**

DDNS Service: DynDNS.org

User Name: [Redacted]

Password: [Masked]

Host Name: tfmapp.dyndns.org

Type: Dynamic

Wildcard: ☐

*Figura 2: Configuración Dynamic Domain Name System (DDNS) en el router*

## **IMPLEMENTACIÓN**

### ***Manual de la aplicación***

Para la aplicación sólo son necesarios dos ficheros jsp. El primero de ellos de configuración en la carpeta admin permite dar de alta consumidores en base a los siguientes parámetros:

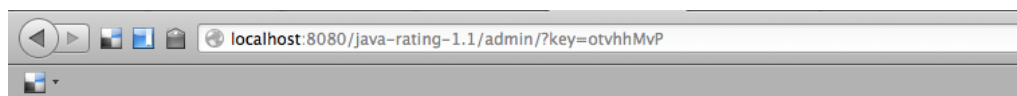
<b>Nombre</b>	<b>Descripción</b>	<b>Obligatorio ?</b>	<b>Valor por defecto</b>
Name	El nombre que se muestra en la instancia de consumidor	Sí	Ninguno
Key	Una cadena única para identificar al VLE que conecta con la aplicación	Sí	Ninguno
Secret	Secreto usado por oauth para lanzar la aplicación	Sí	Una cadena generada de forma aleatoria
Enabled?	Determina si las peticiones de ejecución de la herramienta deben ser aceptadas	No	No
Enable from	Fecha/hora DESDE la que se aceptan las peticiones de ejecución(ej. 1-Jan-2013 00:00)	No	Ninguno
Enable until	Fecha/hora HASTA la que se aceptan las peticiones de ejecución(ej. 1-Jan-2013 00:00)	No	Ninguno
Protected?	Indica si la petición de ejecución debe hacerse solo sin el Tool Consumer tiene el mismo GUID que el que se le pasa	No	No

*Tabla 1: Parámetros de configuración de la aplicación*

El funcionamiento de esta configuración es el definido en la aplicación rating y no se ha modificado por ser correcto. Además se ha dado de alta una aplicación consumidora.

## Estudio de la Integración del estándar OKI OSID con Learning Tools Interoperability (LTI versiones 1.1.1 y 2.0)

### IMPLEMENTACIÓN



Access to this page should be restricted to tool provider administrators only.

Name	Key	Version	Available?	Protected?	Last access	Options
Moodle UOC TFM	otvhhMvP		✓	✓	None	 

#### Update consumer

Name:\*

Key:\*

Secret:\*

Enabled?

Enable from:

Enable until:

Protected?

\* = required field

Update consumer

Cancel

Moodle UOC TFM

otvhhMvP

wyBKdBA4FRd01TFzHNJg4uAi7tGegVOt

☒

☒

Figura 3: Vista de configuración de herramientas consumidoras

El fichero del proveedor en sí mismo index.php en el raíz de la carpeta lo que hace es coger los parámetros que se pasan en la llamada desde Moodle comprueba que hay una clave para ese consumidor (si no hay lo indica por pantalla y no se mostraría nada).

Una vez comprobado que es un cliente se muestran todos los datos que se han recibido desde moodle incluidos nombre de usuario, nombre y apellidos, clave del curso, nombre del curso y rol, etc.

Ahora queda comprobar que es un cliente válido para ello se compara la firma que se puede generar con los datos almacenados en la base de datos (consumidores dados de alta en la aplicación de administración) con los datos recibidos desde Moodle y se generan 2 mensajes y se validan, indicando si es correcto no.

A continuación se ven ambas vistas de configuración en Moodle y de Ejecución:

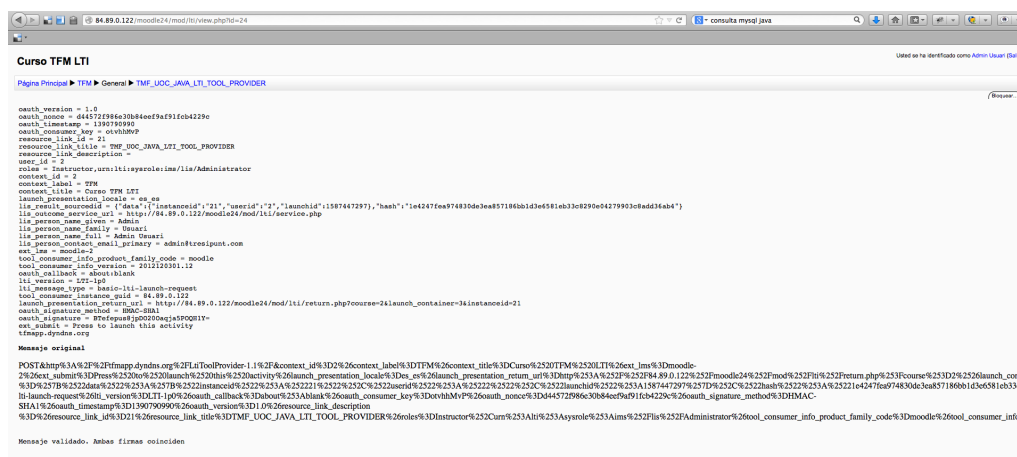


Figura 4: Vista de ejecución de la herramienta en Moodle

## Estudio de la Integración del estándar OKI OSID con Learning Tools Interoperability (LTI versiones 1.1.1 y 2.0)

### IMPLEMENTACIÓN

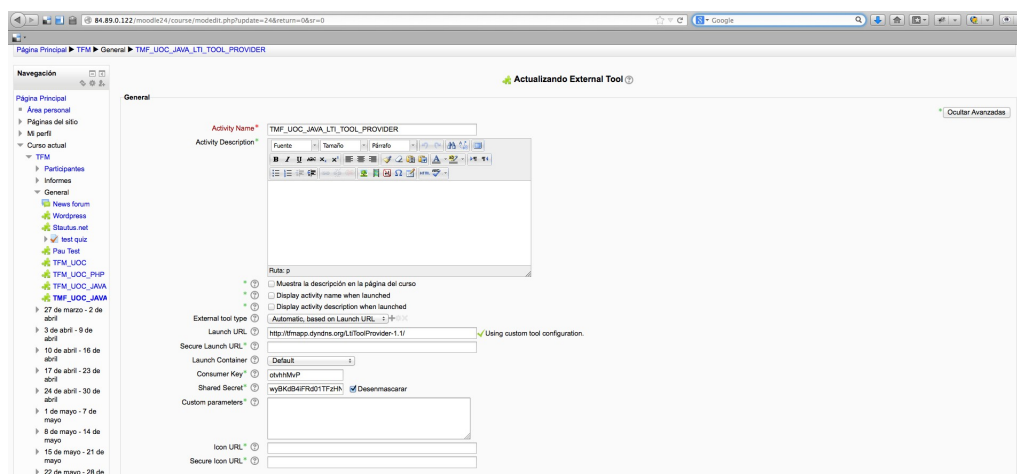


Figura 5: Configuración de Tool Provider con Clave y Secreto en Moodle

El resultado de la implementación se distribuye como un zip para trabajar con él en netbeans o como un .war para su despliegue.