

JUNTA DE EXTREMADURA

*Dirección General de
Tecnologías de la Información y la Comunicación*

GUÍA TÉCNICA

**Desarrollo de Proyectos en
Plataforma Liferay**

GUÍA TÉCNICA

**Desarrollo de Proyectos
en Plataforma Liferay
en la Junta de
Extremadura**

Control de versiones

Núm	Fecha	Descripción	Autores
1.0	01/09/2012	• Estándar para el desarrollo de portales con el gestor de contenidos LIFERAY	SDP
2.0	15/07/14	• Actualización completa a las nuevas plataformas tecnológicas	SDP
3	08/03/19	• Actualización versión Liferay y uso compartido	SDP

Índice del Documento

Capítulo	Página
1- Alcance del documento	4
2.- Plataforma	4
3.- Requisitos que deben cumplir los portales	6
4.- Bases de datos	10

1.- Alcance del documento

Este documento establece las características que el Servicio de Desarrollo de Proyectos considera que deben cumplir los portales web desarrollados en Java utilizando el gestor de portales Liferay.

Se describe el entorno tecnológico existente, el cual hay que tener en cuenta a la hora de implementar y testear la compatibilidad de los proyectos desarrollados.

En este documento se establecen los estándares, frameworks y librerías que deben utilizarse en las aplicaciones a desarrollar, para asegurar unos requisitos mínimos de calidad y estandarización y una completa compatibilidad con el entorno, así como facilitar las tareas de despliegue y mantenimiento de las aplicaciones.

En principio, todas las aplicaciones web desarrolladas en Java utilizando el gestor de portales Liferay deberán respetar las características establecidas en este documento. En el caso de necesitarse algún componente (framework, librería, entorno, etc.), tecnología o herramienta no descrita en el documento, el Servicio con competencias en materia de Administración Electrónica deberá conocer este hecho y aprobar su uso.

Este documento es complementario a la **guía técnica para el desarrollo de aplicaciones JAVA (<http://www.juntaex.es/con01/documentos-tecnicos>)**. Como norma general, se deben seguir todas las directrices marcadas en dicho estándar que no han sido recogidas y/o modificadas en este documento.

2.- Plataforma

El entorno tecnológico LIFERAY utilizado como referencia por la Junta de Extremadura para el desarrollo de sistemas de información es el siguiente:

- **Java Platform, Enterprise Edition 7 (Oracle jdk1.7.0_45)**
 - Es una plataforma de programación para desarrollar y ejecutar software de aplicaciones en lenguaje de programación Java con arquitectura de niveles distribuida, basándose ampliamente en componentes de software modulares ejecutándose sobre un servidor de aplicaciones.
 - <http://www.oracle.com/us/technologies/java/enterprise-edition>
- **Apache Tomcat 7.0.42**
 - Tomcat es un servidor web con soporte de servlets y JSPs. Al estar basado en Java, Tomcat puede ser utilizado en cualquier sistema operativo que lo soporte. Tomcat es mantenido y desarrollado por miembros de la Apache Software Foundation y voluntarios independientes. Tomcat se utiliza como base del despliegue del servidor corporativo de Liferay.
 - Tomcat 7.x Servlet 3.0 JSP 2.2 y EL 2.2 .
 - <http://tomcat.apache.org/>
- **Liferay Portal 6.2: Liferay Portal Community Edition 6.2 CE GA4**
 - Liferay Portal es una plataforma o framework Open Source para construir portales web. Liferay proporciona inmediatamente amplias capacidades como Gestión Documental y de Contenido, Publicación Web y Espacios de Trabajo Compartidos, Colaboración Empresarial, Redes Sociales de Conocimiento, Portales Empresariales, Sitios Web 2.0, Integración de Aplicaciones Empresariales, etc. Liferay está 100% desarrollada en Java y utiliza estándares, frameworks y librerías ampliamente probados como hibernate, struts, spring, etc. Por tanto, es compatible con la mayoría SOs, contenedores Servlet, servidores de aplicaciones y Bases de Datos.
 - <http://www.liferay.com/>
- **SQL Server 2008**
 - SQL Server 2008 es un gestor de bases de datos transaccional basado

en el modelo relacional.

- <http://www.microsoft.com/sqlserver>

3.- Requisitos que deben cumplir los portales

3.1.- Compatibilidad de navegadores y sistemas operativos

A continuación se exponen los criterios a seguir en relación a hacer compatibles las aplicaciones y sus entornos de ejecución :

- Los portales web deben ser compatibles con los navegadores más utilizados hoy en día. Se exige como mínimo compatibilidad con los navegadores:
 - Internet Explorer
 - Mozilla Firefox
 - Chrome

En todo momento hay que evitar código que pueda provocar problemas de incompatibilidad entre navegadores diferentes e incluso entre diferentes versiones del mismo navegador.

- El portal debe funcionar correctamente en clientes *Windows* y clientes *Linux*.

3.2.- Independizar el portal de un contexto

Deberán cumplirse los siguientes criterios:

- Los portales deberán implementarse y empaquetarse de manera que sea

posible desplegar el mismo portal más de una vez sobre el mismo servidor (con diferente nombre y contexto).

Para conseguir esta compatibilidad, los aspectos que hay que tener en cuenta a la hora de desarrollar la aplicación son los siguientes:

- Será necesario poder modificar el contexto de la aplicación. Una de las posibilidades es cambiar el nombre del fichero *.war* de la aplicación. En algunos servidores también es posible definir el contexto de la aplicación en ficheros de configuración (*context.xml*, *jboss-web.xml*, ...).
- Todas las referencias a URLs y/o recursos de la aplicación deben ser relativas o dinámicas a partir del contexto de la aplicación.
- El acceso a BD (datasource utilizado) debe ser parametrizable. Así conseguiremos definir a la BD que accede cada aplicación.
- Se deberá parametrizar todo lo posible aquellos parámetros relacionados con la configuración de la aplicación, y que puedan desencadenar problemas en el momento de desplegar más de una aplicación sobre el mismo servidor.

3.3.- No ejecutar código que afecte al comportamiento de la máquina virtual

- No puede existir código en los portales que pueda modificar o cambiar el comportamiento de la máquina virtual de JBoss. Existen algunas instrucciones, tales como `System.setProperty(String key, String value)` ó `System.gc()` que tienen efectos sobre la maquina virtual que ejecuta JBoss, y por lo tanto puede tener efectos secundarios sobre el resto de aplicaciones que están corriendo en el servidor.

3.4.- Requisitos generales

- Se eliminarán del war de liferay todos los elementos (jar, librerías,

portlets, etc.) que no se empleen en el portal.

- Los elementos desplegados automáticamente en el directorio deploy propio de una instancia de liferay vendrán integrados en el war de la aplicación, esto incluye todos los plugins, portlets, temas, etc., que pudieran.
- Todas las propiedades necesarias para el portal, se deberán definir en el fichero LIFERAY_WAR/WEB-INF/clases/portal-ext.properties. Nunca se deben almacenar propiedades en la base de datos (tabla portletpreferences).
- Los enlaces presentes en la aplicación (en contenidos web, noticias, artículos, etc.) deben ser siempre relativos, para que se añada a la url automáticamente el servidor, puerto y contexto en el que se despliega la aplicación, por ej: <http://servidor:8080/portalejemplo>
- Importante destacar que van a coexistir varios portales dentro de una misma instancia de Liferay. Por ello:
 - La modificación de los portlets base o el core de Liferay mediante hooks o plug-ins de extensión no está permitida ya que puede afectar a futuras actualizaciones y cambios de versión de Liferay, además de producir comportamientos no deseados en los otros portales.
 - Por motivos de compatibilidad en los cambios de versión, no está permitido el desarrollo de componentes que hagan uso del API de Liferay.
 - Si es necesaria la creación de un nuevo componente se empaquetará en un fichero war. Estos componentes que deben seguir la siguiente nomenclatura:
 1. • Temas de apariencia: *juntaex--NOMBRE_TEMA-theme.war*
 2. • Layouts de página: *juntaex--NOMBRE_LAYOUTP-layout.war*
 3. • Portlets: *juntaex--NOMBRE_PORTLET-portlet.war*
 - Ya que Liferay 6.2 está muy integrado con bootstrap 2.3.2, se recomienda utilizar esos estilos en las páginas jsp y aprovechar sus características de responsividad.
 - Los portlets no incluirán librerías css ni javascript que ya se estén

incluyendo en el tema.

- En caso de necesitar acceso a BD se recomienda utilizar el "Service Builder" de Liferay para la creación de los servicios de acceso a BD.
- Los componentes que requieran de sus propias tablas en base de datos deberán respetar la nomenclatura de los elementos de base siguiente: *_Nombre_de_la_tabla*
- Para el despliegue de los componentes con acceso a datos deben adjuntarse los scripts de creación/modificación de las tablas, los scripts de introducción/actualización de datos y los scripts para realizar una marcha atrás.
- Los portlets deben venir categorizados en una 'Categoría' propia para cada grupo de portlets, según el tema de la web, para que cuando se seleccione desde Liferay el/los elementos a poner en una página no se tenga que buscar donde está, tan solo irse a la categoría asociada al tema de la web. Por ejemplo para los portlet desarrollados para un portal "Turismo Extremadura" se crearía la **categoría "Turismo Extremadura"**.

3.5.- Tratamiento de procesos pesados en el servidor

- En algunas ocasiones los portales web requieren la ejecución de procesos que pueden llegar a consumir muchos recursos de máquina (memoria y procesador), y cuya ejecución se puede alargar durante un tiempo no despreciable, como pueden ser:
 - procesos de calculo complejo
 - consultas masivas de datos
 - creación de listados o informes masivos de datos
 - etc..
- La ejecución de estos procesos de forma interactiva suele provocar problemas como:
 - la perdida de *timeout* de la sesión web
 - problemas con los sistemas de filtrado, seguridad, y/o balanceo de carga que puedan existir en la interconexión entre el cliente y el

- servidor.
- penalización del rendimiento del servidor Java, y en consecuencia penalización del rendimiento de las aplicaciones desplegadas.
 - Si la aplicación requiere de la ejecución de algún proceso con estas características, es necesario que ofrezca al usuario una forma alternativa de ejecución que no ocasione los problemas anteriormente mencionados, y lance el proceso en "*segundo plano*", e incluso en su caso pudiese derivarlo a otro servidor que llevaría a cabo el proceso.

A su vez deberá implementarse un "*proceso de notificación*", de forma que cuando el proceso haya finalizado se le comunique al usuario de alguna forma. En ese momento el usuario de la aplicación podrá acceder o consultar los resultados de la ejecución del proceso.

3.6.- Gestión de documentos en el gestor de contenidos

- El repositorio de documentos e índices estarán definidos en el servidor de ficheros implementado con OCFS (Oracle Cluster File System). Cada instancia y sitio tendrá su propio espacio.

4.- Bases de datos

4.1.- Drivers JDBC

Microsoft SQL Server

- **Microsoft sql server jdbc driver 4.0 (sqljdbc4.jar)**
 - Driver de Microsoft que permite a las aplicaciones Java, acceder a las bases de datos Microsoft SQL Server 2000, Microsoft SQL Server 2005 y Microsoft SQL Server 2008 a través de JDBC (Java Database Connectivity).
 - <http://www.microsoft.com/downloads/>

PostgreSQL

- **PostgreSQL JDBC Driver (postgresql-9.4.1209.jre6.jar)**
 - Driver JDBC 4 de PostgreSQL.
 - <http://jdbc.postgresql.org/>

MySQL

- **MySQL Connector/J 5.1 (mysql-connector-java-5.1.11-bin.jar)**
 - El MySQL Connector/J es un driver JDBC 4 y contiene todas las características de JDBC para manejar Mysql.
 - <http://dev.mysql.com/downloads/connector/j/>

ORACLE

- **ORACLE JDBC driver (ojdbc6.jar)**
 - El driver JDBC 4 para ORACLE, contiene todas las características de JDBC para manejar ORACLE.
 - <http://www.oracle.com/technetwork/database/features/jdbc>