

# EXPAND DOCUMENTS

## DESCRIPCIÓN TÉCNICA

A continuación se detallan las distintas tecnologías utilizadas para la construcción del sistema documental EXPAND DOCUMENTS

### CAPA DE PRESENTACIÓN

Para la implementación de la capa de presentación de EXPAND DOCUMENTS se utiliza como marco de trabajo el framework Java Server Faces, tal y como se detalla a continuación. JSF es el nuevo framework estándar, desarrollado vía JPC (Java Community Process), que permite construir fácilmente interfaces de Usuario para aplicaciones Web desarrolladas en Java.

#### A. FACELETS 1.1

Facelets es una tecnología de presentación orientada a la construcción de componentes JSF. Aunque no forma parte de la especificación original de JSF, en la actualidad está considerado ya como estándar y ligado a la tecnología JSF.

#### B. RICHFACES 3.3.0

RichFaces es una implementación JSF que extiende a este a través del uso de AJAX.

#### C. SEAM-UI

Seam incluye un conjunto de controles y tags JSF que resultan muy útiles en conjunción con el framework de Seam. Permiten aprovechar ciertas características de Seam desde la página JSF.

#### D. SOPORTE I18N

Internacionalización es el proceso necesario de diseño que permite que una aplicación se pueda adaptar a varios idiomas sin necesidad de realizar cambios.

#### E. PÁGINAS XHTML (JSF)

XHTML es el lenguaje de marcado pensado para sustituir a HTML como estándar para las páginas Web. XHTML es la versión XML de HTML, por lo que tiene, básicamente, las mismas funcionalidades, pero cumple las especificaciones, más estrictas, de XML.

### CAPA DE LÓGICA DE NEGOCIOS

En su implementación de la capa de lógica de negocios, EXPAND DOCUMENTS hace uso del framework JBoss Seam, permitiendo así unir los beneficios del uso de Java Server

Faces para la capa de presentación y EJB 3.0 para la capa de la lógica de negocios.

#### A. SEAM / EJB 3.0

JBoss SEAM es un framework para Java EE 5.0 que permite un modelo de programación consistente y fácil de entender y que integra todos los componentes en una aplicación Web moderna.

#### B. SERVICIOS DE NEGOCIO DE LA APLICACIÓN

Los Servicios de Negocio de la Aplicación permiten la definición de Servicios Web que serán publicados para su acceso desde interfaces externas.

Los objetos funcionales correspondientes a Servicios Web están creados mediante JBoss WS haciendo uso de los objetos de negocio ya existentes.

También existe la posibilidad de integración para acceder y encadenar las funciones de la plataforma como servicios REST/SOAP para flexibilizar la incorporación de Servicios mediante esta tecnología.

### CAPA DE PERSISTENCIA

#### A. HIBERNATE 3.3.1 GA

Hibernate es una herramienta de mapeo entre objetos y tablas relacionales que permite una integración transparente y automática de la persistencia entre objetos Java y tablas de base de datos.

#### B. DAO

La capa de objetos de acceso a datos (DAO) sirve para separar en clases diferentes el código de acceso a datos y la lógica de la aplicación.

#### C. MODELO DE DOMINIO (ENTIDADES)

El Modelo de Dominio es el conjunto de clases Java que representan las entidades persistentes que se manejarán por parte del motor de persistencia.

#### D. GESTOR DE PERSISTENCIA (JPA)

El contenedor de EJB3 incluye un gestor/manejador de persistencia basado en ORM. En el caso de JBOSS-AS el gestor de persistencia es Hibernate.

#### E. GESTOR DE CONVERSACIONES (SEAM)

Tanto la aplicación Web como algunos servicios SOAP deben realizar operaciones de negocio que impliquen la navegación por más de una página o la coordinación de llamadas a servicios Web. Para que estas operaciones puedan ser confirmadas o canceladas de forma segura, Seam introduce el concepto de contexto de conversación. Todas las operaciones que se realicen sobre una misma conversación serán confirmadas y/o canceladas de forma atómica.

### GESTOR DE TRANSACCIONES (JTA)

El contenedor de EJB3 proporciona automáticamente un gestor estandarizado de control de transacciones. El gestor de transacciones permite coordinar una transacción que incluya diferentes recursos en diferentes orígenes de datos (JDBC, JMS, etc.). Además, permite especificar declarativamente los límites de la transacción a nivel de componente EJB.

### INTEGRACIÓN CON OTROS COMPONENTES

Entre los componentes que se integran dentro de EXPAND DOCUMENTS se encuentran los siguientes:

- Comunicación con componentes COM de Microsoft desde Java. Para más información:

<http://sourceforge.net/projects/jacob-project/>

Mediante este componente permite ofrecer la funcionalidad de "Combinar correspondencia" de Microsoft Word desde Java.

- Integración con plataforma de desarrollo (SDK) de OpenOffice. Para más información:

<http://www.openoffice.org/>

La funcionalidad de "Combinar correspondencia" de OpenOffice es ofrecida por el producto, así como el servicio de conversión de formatos, entre los que se encuentra la conversión a PDF desde documentos Microsoft Word.

- Integración con plataformas de firma electrónica: Las funcionalidades de Firma Electrónica se apoyan en dos plataformas de firma electrónica que son: @Firma (<http://www.csae.map.es/csi/pg5a12.htm>) y TrustedX (<http://www.safelayer.com/es/soluciones/productos-y-servicios/trustedx>).

### CARACTERÍSTICAS DE GESTIÓN DOCUMENTAL

- El repositorio de documentos gestionado por EXPAND DOCUMENTS puede estar definido en un SGDB (Oracle, SQL Server) o bien en una estructura de directorios. En el caso de la presente oferta se propone como solución que dicho repositorio quede almacenado sobre Oracle.

- La indexación de los documentos y sus metadatos se realiza a través de Lucene.

Para más información:

<http://lucene.apache.org/java/docs/index.html>