

	<p>Proyecto: PT-w@ndA. Construcción de procedimientos sobre PT-w@ndA</p> <p>DG.DF.E01-Diseño Funcional</p>	<p>Consejería de Justicia y Administración Pública</p>			
<h2>Resumen</h2> <p>El presente documento presenta la metodología a seguir para la construcción de procedimientos sobre la plataforma de tramitación w@ndA.</p>					
<h2>Registro de modificaciones</h2>					
Versión	Descripción	Autor	Fecha creación	Aprobado por	Fecha aprobación
0.1	Versión inicial del documento con una descripción básica del proceso para la construcción sistemática de procedimientos.	everis	26-09-08		
0.2	Añadida la ubicación lógica del modulo y la estructura del zip generado	everis	03-10-08		

Contenido

1. Requisitos previos a la construcción de procedimientos	4
2. Preparación del entorno de desarrollo	5
3. Estructura lógica de la información	6
3.1 Estructura lógica – Configuración	7
3.2 Estructura lógica – java	8
3.3 Estructura lógica – Recursos jsp	8
3.4 Estructura lógica – css e imágenes.....	9

Índice de figuras

Índice de tablas

1. Requisitos previos a la construcción de procedimientos

Descripción breve de los objetivos del sistema.

Es necesario disponer de una instalación válida de Trew@ con la definición del procedimiento que va a implementar cargada en ella.

Para la plataforma de tramitación, se requiere de una instancia de cualquier motor de base de datos relacional compatible con Hibernate 3 (por ejemplo, MySQL, PostgreSQL, Oracle, etc.).

Debido a que frecuentemente será necesario reiniciar la información relativa a módulos desplegados en la plataforma de tramitación, configuraciones, etc., se recomienda que cada desarrollador tenga una instancia local del SGDB utilizado para almacenar el esquema de datos de la plataforma.

2. Preparación del entorno de desarrollo

Presentación de la información del sistema y de su contexto a alto nivel. Definición de sus límites y de sus relaciones con otros sistemas y actores externos a la organización.

Dependiendo del enfoque escogido para el Diseño Funcional (Orientado a Objetos o Estructurado) se utilizarán unos u otros diagramas.

Ejemplo:

- Enfoque Estructurado: Diagrama de Contexto (DFD de alto nivel)
- Enfoque Orientado a Objetos: Diagrama de Casos de Uso (Diagrama de Casos de Uso de alto nivel)

Para el desarrollo de procedimientos se partirá de la última versión estable del kit de desarrollo de la plataforma de tramitación w@ndA. Dicho kit viene empaquetado en un fichero comprimido (.zip) que contiene:

- Las librerías utilizadas por la plataforma de tramitación. Entre ellas, destaca el fichero `escritorio-core.jar`, que contiene la lógica de la aplicación.
- Páginas web del escritorio de tramitación. Normalmente se tratará de ficheros JSP (*Java Server Pages*).
- Recursos web de la aplicación: imágenes, hojas de estilo, archivos con funciones JavaScript.
- Archivos de configuración, tanto de la propia aplicación como de los diferentes *frameworks* utilizados en su construcción (Struts, Spring, Hibernate, log4java, etc.).

El kit de desarrollo estará organizado como un proyecto Maven (<http://maven.apache.org>), por lo que se recomienda un aprendizaje de dicha herramienta para su correcta utilización.

3. Estructura lógica de la información

Para la construcción de un nuevo procedimiento en la plataforma de tramitación w@ndA se distinguen 3 tareas:

Tarea	Descripción de la Entidad	Ficheros
Módulo - Alta Expediente	Módulo para realizar el alta de expediente.	Librerías utilizadas,páginas web y recursos de configuración necesarios(ficheros de struts, spring y archivo de despliegue de módulo)
Módulo - Tareas	Módulo para realizar las tareas del expediente.	Librerías utilizadas,páginas web y recursos de configuración necesarios(ficheros de struts, spring y archivo de despliegue de módulo)
Módulo – Utilidades	Módulo con las utilidades del procedimiento.	Librerías utilizadas,páginas web y recursos de configuración necesarios(ficheros de struts, spring y archivo de despliegue de módulo)

Para el desarrollo de un módulo hay que tener en cuenta la ubicación de los fichero dentro del sdk. Se distinguirán 3 grandes apartados a la hora de estructurar los ficheros de un módulo:

- **configuración:** ficheros de configuración tanto de struts2 como de spring en el caso de que se usara.
- **java:** clases java para el desarrollo del módulo junto con sus ficheros properties, xml, etc. que fueran necesarios.
- **recursos jsp:** ficheros jsp del módulo
- **css e imágenes:** estilos e imágenes usadas en el módulo

3.1 Estructura lógica – Configuración

En la siguiente tabla se detallan las tareas a realizar para desarrollar correctamente los archivos de configuración de struts y spring en un módulo.

struts2	Su ubicación debe ser <code><classpath>:/modulos/<NOMBRE_MODULO></code>
	La nomenclatura debe ser <code>struts-*.xml</code>
	Habrà un fichero de configuración por cada tipo de acción, esto es: <ul style="list-style-type: none"> - configuración del alta - configuración de las tareas - configuración de las utilidades
	El fichero de configuración para el alta debe tener el namespace: <ABREVIATURA_PROCEDIMIENTO>/altaExpediente
	El fichero de configuración para las tareas debe tener el namespace: <i>agenda/tareas</i>
	El fichero de configuración para las utilidades debe tener el namespace: <ABREVIATURA_PROCEDIMIENTO>/utilidades
spring	Su ubicación debe ser <code><classpath>:/modulos/<NOMBRE_MODULO></code>
	La nomenclatura debe ser <code>applicationContext.xml</code>
	Sólo habrá un fichero de configuración

3.2 Estructura lógica – java

El paquete base de cualquier fichero java/properties/config será

com.everis.<ABREVIATURA_PROCEDIMIENTO>, a partir de esta ruta tendremos los siguientes paquetes:

PAQUETE	DESCRIPCIÓN
altaExpediente	ficheros relacionados con el proceso de alta de un expediente
Tareas	ficheros relacionados con las tareas de un expediente
utilidades	ficheros relacionados con las utilidades
condiciones	ficheros relacionados con las condiciones
Acciones	ficheros relacionados con las acciones
Variables	ficheros relacionados con las variables
xmlExpediente	ficheros relacionados con la estructura XML de otros datos de expedientes
Dao	ficheros relacionados con el acceso a BBDD
Dto	tipos propios de datos
Config	ficheros de configuración (xml,betwin,etc)
properties	ficheros properties
Actions	ficheros de acciones de struts

3.3 Estructura lógica – Recursos jsp

La ruta base de cualquier recurso jsp será:

/modulos/<ABREVIATURA_PROCEDIMIENTO>. A partir de aquí tendremos los siguientes bloques:

BLOQUE	DESCRIPCIÓN
altaExpediente	jsp relacionadas con el alta de un expediente
tareas	jsp relacionadas con las tareas de un expediente
utilidades	jsp relacionadas con las utilidades
decorators	jsp para la decoración

3.4 Estructura lógica – css e imágenes

La ubicación de las hojas de estilo será:

/instalaciones/<NOMBRE_INSTALACION>/css.

La ubicación de las imágenes será:

/instalaciones/<NOMBRE_INSTALACION>/imágenes.

Tanto para las imágenes como las hojas de estilo será importante usar un prefijo de nombre unívoco.

Una vez terminado el desarrollo del módulo hay que empaquetarlo en un zip. La estructura debe ser la siguiente:

ruta	descripción
/	Raíz del zip. En esta ubicación se debe colocar el archivo de despliegue del módulo cuyo nombre debe ser despliegue.xml
/webapp	En esta se colocan los páginas jsp del módulo, Según la misma estructura definida en el desarrollo (a partir de la ruta /src/main/webapp/modulos/<nombreModulo>).
/webapp/CSS	En esta carpeta se deben colocar las hojas de estilos css usadas.
/webapp/IMÁGENES	En esta carpeta se deben colocar las imágenes usadas en el módulo.
/conf	En esta carpeta se deben colocar los archivos de configuración de struts, spring y los archivos properties usados. Según la misma estructura definida en el desarrollo (a partir de la ruta /src/main/config/<nombreModulo>)
/lib	En esta carpeta se deben colocar la librería jar generada con las clases del módulos y otras librerías usadas para su desarrollo.

NOTA: Si el módulo es de tipo NONE no es necesario que existan las carpetas IMÁGENES Y CSS, en otro caso es necesario que existan aunque se encuentren vacías.

A continuación se muestra un ejemplo de archivo de despliegue de módulo

<pre> <def-modulo> <nombre>nombreModulo</nombre> <version>versión (ejemplo: 1.0)</version> <titulo>título del módulo</titulo> <descripcion>descripción del módulo</descripcion> <url>url de acceso al action. Necesario para el módulo de alta</url> <menu> <nombre>ESCRITORIO o ADMINISTRACION</nombre> </menu> <autor> <nombre> </nombre> <email> </email> <grupo>general</grupo> </autor> <type>STRUTS-2</type> //indica el framework usado para el desarrollo del módulo. A fecha de redacción de este documento se aseguro el correcto funcionamiento con struts-2. <tipoInstalacion>Tipo de instalación (NONE,EXTERNO,PORTLET,UTILIDADES)</tipoInstalacion> </def-modulo> </pre>