

## Especificaciones técnicas para el desarrollo de aplicaciones J2EE

Referencia:	AST-ESPEC-EspecificacionesTecnicasJ2EE.doc
Autor:	Natividad Porta
Fecha de creación:	02/11/2007
Última actualización:	12/12/2007
Versión:	v1.0
Clasificación:	Restringido

Colección: Especificaciones técnicas para el desarrollo de aplicaciones para el Gobierno de Aragón.

## Control del documento

### Registro de cambios

Versión	Fecha	Autor	Descripción
v1.0	12/12/2007	Natividad Porta	Primera versión del documento.

### Revisores

Nombre	Área

### Lista de distribución

Nombre	Área
Todo el personal	Aragonesa de Servicios Telemáticos
Personal informático	Gobierno de Aragón
Empresas desarrolladoras	Empresas externas

## Contenido

<b>1. INTRODUCCIÓN .....</b>	<b>4</b>
<b>2. ESPECIFICACIONES TÉCNICAS DE LAS APLICACIONES.....</b>	<b>5</b>
2.1. ESPECIFICACIONES DE BASE DE DATOS.....	5
2.1.1. Usuarios de base de datos .....	5
2.1.2. Configuración del acceso a base de datos.....	5
2.1.3. Restricciones de seguridad en el acceso a base de datos .....	5
2.2. ESPECIFICACIONES DE ESTILOS, ACCESIBILIDAD Y USABILIDAD .....	5
2.3. CONVENCION DE CÓDIGO .....	6
2.4. NOMENCLATURA .....	6
2.4.1. Nomenclatura de la aplicación.....	6
2.4.2. Nomenclatura de paquetes y clases Java .....	6
2.5. SISTEMA DE LOG .....	6
• Ejemplo de posible configuración de log4j .....	7
2.6. PATRÓN DE DISEÑO .....	7
2.7. VERSIONADO DE APLICACIONES .....	7
• Primer nivel X.....	7
• Segundo nivel Y .....	8
• Tercer nivel Z .....	8
2.8. ENTREGA DE APLICACIONES .....	8
<b>3. ENTORNOS DE DESPLIEGUE.....</b>	<b>9</b>
3.1. ENTORNO DE DESARROLLO .....	9
3.2. ENTORNO DE PREPRODUCCIÓN .....	9
3.3. ENTORNO DE PRODUCCIÓN.....	10
<b>4. DOCUMENTACIÓN DE LA APLICACIÓN.....</b>	<b>11</b>
• Descripción general de la aplicación.....	11
• Requisitos del aplicativo.....	11
• Descripción del modelo de datos .....	11
• Arquitectura de aplicación.....	11
• Fuentes Java .....	11
• Administración de la aplicación .....	12
• Errores de aplicación .....	12
• Plan de pruebas .....	12
• Manual de usuario .....	12

## 1. Introducción

Este documento describe las características que deben cumplir las aplicaciones desarrolladas en entorno J2EE para el Gobierno de Aragón.

El documento trata los siguientes temas:

- Especificaciones de las aplicaciones: descripción de las características técnicas de las aplicaciones desarrolladas.
- Entornos de despliegue disponibles.
- Documentación de la aplicación: descripción de la documentación que se debe entregar junto con la aplicación.

## 2. Especificaciones técnicas de las aplicaciones

### 2.1. Especificaciones de base de datos

Para las especificaciones propias de los objetos de base de datos, diseño, nomenclatura, dimensionamiento, etc., se seguirá lo detallado en el documento “Especificaciones de base de datos – Oracle 10g” (AST-ESPEC-BaseDatosOracle10g.pdf).

#### 2.1.1. Usuarios de base de datos

El usuario con que se conecte la aplicación **no será nunca el propietario** del esquema; en caso de conexiones personalizadas (aplicación con autenticación), cada usuario de aplicación tendrá su correspondiente usuario de base de datos.

Para la utilización anónima o pública de la aplicación, se utilizará un único usuario de base de datos, empleando además el pool de conexiones proporcionado por el servidor de aplicaciones, parametrizado adecuadamente, para una gestión más eficiente de las conexiones.

#### 2.1.2. Configuración del acceso a base de datos

La configuración de acceso a la base de datos se definirá en el fichero data-sources.xml, incluido en la propia aplicación, dentro del directorio META-INF del ear generado.

Se puede ver un ejemplo de este fichero en la nota técnica “Configuración data-sources” (documento AST-NTEC-DefinicionDataSourcesOAS.pdf).

#### 2.1.3. Restricciones de seguridad en el acceso a base de datos

Todos los accesos a tablas de la base de datos se harán **a través de procedimientos almacenados**, -procedimientos y funciones- convenientemente agrupados en paquetes.

Se definirán los roles convenientes -uno o varios-, a los que se darán los permisos de ejecución sobre los paquetes estrictamente necesarios, nunca otorgando permisos directamente sobre las tablas o vistas. A cada usuario –personalizado o de aplicación- se le asignará el rol apropiado.

### 2.2. Especificaciones de estilos, accesibilidad y usabilidad

Las directrices de estilo, accesibilidad y usabilidad que deben cumplir **todas las aplicaciones**, independientemente de su público objetivo, vienen especificadas en la “Guía de Estilos del Portal de Internet del Gobierno de Aragón” (AST-ESPEC-GuiaEstilosPortalInternet.pdf).

## 2.3. Convención de código

En el código Java se seguirá, en la medida de lo posible, el estándar de codificación Java de Sun Microsystems que se puede encontrar en la dirección <http://java.sun.com/docs/codeconv/>

## 2.4. Nomenclatura

### 2.4.1. Nomenclatura de la aplicación

Para la aplicación se utilizará un nombre suficientemente descriptivo, con el que se registrará en el inventario de aplicaciones de Aragonesa de Servicios Telemáticos. Además, se le asignará un código identificativo alfanumérico, preferiblemente de tres o cuatro caracteres, generalmente formado con las iniciales del nombre.

Este código deberá ser único en todo el ámbito del Gobierno de Aragón y por ello será acordado al inicio de proyecto con Aragonesa de Servicios Telemáticos.

### 2.4.2. Nomenclatura de paquetes y clases Java

Los nombres de los paquetes Java comenzarán, por convención, con `es.aragon.codigo_aplicacion` –donde `codigo_aplicacion` se sustituye por el código mencionado en el apartado anterior- y posteriormente con los paquetes que sean necesarios.

Así, por ejemplo, son nombres válidos de paquetes:

- `es.aragon.codigo_aplicacion.dal`: paquete de clases de acceso a datos almacenados en distintas fuentes de datos, como bases de datos, ficheros, directorios LDAP, etc.
- `es.aragon.codigo_aplicacion.bl`: paquete de clases que implementan la lógica de negocio de la aplicación.
- `es.aragon.codigo_aplicacion.servlet`: paquete que contiene las clases correspondientes a los servlets del aplicativo.
- `es.aragon.codigo_aplicacion.util`: paquete de utilidades genéricas de la aplicación.

## 2.5. Sistema de log

Las aplicaciones utilizarán un sistema de log que escriba en disco los mensajes, generando un fichero por día –*rolling* diario-. De esta forma, se tendrá un histórico del log almacenado en ficheros ordenados según el día, para que en caso de ser solicitados, los ficheros se puedan enviar fácilmente al proveedor.

Se utilizará el sistema Log4j por ser un estándar e implementar *rolling* diario de los ficheros de log, debiendo configurar el nivel de log deseado –DEBUG, INFO, WARN, ERROR, FATAL– acorde con el entorno –desarrollo, preproducción y producción–.

La librería de Log4j ya se encuentra en los servidores de aplicaciones, de forma que no debe ser incluida en cada aplicación.

### ■ Ejemplo de posible configuración de log4j

En el siguiente ejemplo es necesario sustituir la cadena `cod_app` por el código de la aplicación y la cadena `nivel_log` por el nivel de log deseado.

```
# cod_app = nombre del appender que se configura
# nivel_log = DEBUG | INFO | WARN | ERROR | FATAL
log4j.logger.cod_app=nivel_log, cod_app
# almacena el fichero de log diariamente
log4j.appender.cod_app=org.apache.log4j.DailyRollingFileAppender
# path y nombre del fichero de log
log4j.appender.cod_app.File=/logs/app/cod_app/cod_app.log
# Se guarda el fichero de log de cada día
log4j.appender.cod_app.DatePattern='.'yyyy-MM-dd
# Definir el formato de salida del log por fichero
log4j.appender.cod_app.layout=org.apache.log4j.PatternLayout
# formato de escritura en el fichero de log
# %d fecha
# %p prioridad
# %m mensaje
# %n salto de línea
log4j.appender.cod_app.layout.ConversionPattern= %d %p %m %n
```

En particular, no se utilizará nunca la propiedad `rootLogger`.

## 2.6. Patrón de diseño

Las aplicaciones seguirán un diseño que independice las capas de presentación, lógica de negocio y acceso a datos. Se recomienda en particular la implementación del patrón de diseño Modelo-Vista-Controlador.

## 2.7. Versionado de aplicaciones




Cada entrega de aplicación debe ir adecuadamente versionada, incluyendo en la entrega la información de los cambios que motivan la nueva versión. La versión se especificará, además, en el archivo MANIFEST.MF del fichero EAR generado.

Se puede obtener más información de cómo versionar aplicaciones en la nota técnica "Versionado de librerías y aplicaciones J2EE" (AST-NTEC-VersionadoAplicacionesJ2EE.pdf).

El versionado se define en base a tres niveles X.Y.Z que se explican a continuación:



### ■ Primer nivel X

El primer nivel X define versiones mayores, lo que corresponde a una línea base en terminología de gestión de requisitos.

-  La versión 0.a.b corresponde a las versiones de un programa que no incorporan toda la funcionalidad definida para la línea base.
-  La versión 1.0.0 corresponde a la primera versión estable y completa de un programa que incorpora todas las funcionalidades definidas en el análisis.
-  La versión 2.0.0 corresponde a la versión 2 de un proyecto que se ha reestructurado completamente.


### ■ Segundo nivel Y

El segundo nivel Y define versiones menores, como modificación de algún requisito que implica cambios menores en un programa.

-  La versión 1.0.0 corresponde a la primera versión estable y completa de un programa que incorpora todas las funcionalidades definidas en el análisis.
-  La versión 1.1.0 corresponde con un cambio de un requisito pedido por el cliente y que cambia el planteamiento inicial. Sucesivos cambios irían incrementando este valor Y.

### ■ Tercer nivel Z

El tercer nivel Z define modificaciones muy pequeñas como corrección de *bugs* en desarrollos.

-  Una corrección simple en un programa puede hacer que pasemos de la versión 1.2.12 a la versión 1.2.13.





## 2.8. Entrega de aplicaciones

Para la entrega de aplicaciones para su instalación en servidores del Gobierno de Aragón, se seguirán las especificaciones marcadas en el documento "Especificaciones de entrega de aplicaciones J2EE" (AST-ESPEC-EntregaAplicacionesJ2EE.pdf).



### 3. Entornos de despliegue

Existen tres entornos de despliegue, que corresponden a desarrollo, preproducción y producción. Las características técnicas de los tres son iguales, salvo las diferencias que más abajo se detallan:

-  Sistemas Sun Solaris
-  Servidor de aplicaciones Oracle Application Server 10g Release 2
  - Estándar J2EE 1.3 –implementado por el servidor de aplicaciones-
-  Servidor de base de datos Oracle Database 10g Release 2 RAC
-  Driver thin JDBC de conexión a base de datos: Oracle JDBC Driver 10.1.0

Se puede encontrar información sobre el estándar J2EE 1.3 implementado por el servidor de aplicaciones en la URL: <http://java.sun.com/j2ee/1.3/index.jsp>

A modo de resumen, el estándar incluye las siguientes especificaciones:

Estándar	Versión
<b>J2SE</b>	1.4 (Java Runtime 1.4.2)
<b>EJB</b>	2.0
<b>Servlet</b>	2.3
<b>JSP</b>	1.2

En los entornos de preproducción y producción, la base de datos se encuentra en *RAC –Real Application Cluster-*, y el servidor de aplicaciones en *cluster*; ésta es la diferencia con el entorno de desarrollo, donde no hay alta disponibilidad.

#### 3.1. Entorno de desarrollo

Este entorno es el utilizado para las primeras pruebas de las aplicaciones por parte de los usuarios validadores. Se puede utilizar para realizar pruebas funcionales y de integración con otros sistemas.

#### 3.2. Entorno de preproducción

En este entorno se pueden hacer pruebas de carga, así como de alta disponibilidad, tanto de servidor de aplicaciones como de base de datos, previo acuerdo con los técnicos de Aragonesa de Servicios Telemáticos.

### 3.3. Entorno de producción

Es el entorno real de ejecución de las aplicaciones, donde se encuentran accesibles para los usuarios finales.

## 4. Documentación de la aplicación

Este apartado detalla la documentación que se debe incluir en las aplicaciones J2EE desarrolladas para el Gobierno de Aragón.

### ■ Descripción general de la aplicación

Introducción a la aplicación, cuál es su objetivo, a qué público atiende, si responde a una normativa legal determinada, si su uso es continuo o en determinados periodos, volumen de usuarios esperado, horario de funcionamiento, requerimientos especiales de seguridad –como acceso mediante HTTPS–, etc.

### ■ Requisitos del aplicativo

Se detallarán las necesidades hardware y software del aplicativo, incluyendo entre otros los siguientes datos:

- Hardware: capacidad de disco necesaria (base de datos u otras fuentes de datos, otros ficheros de aplicación, ficheros temporales, ficheros de log), previsión de crecimiento, y otros requisitos particulares.
- Software: librerías específicas (no estándar del servidor) utilizadas por la aplicación, incluyendo la versión.

### ■ Descripción del modelo de datos

Se describirán las fuentes de datos utilizadas por la aplicación, como el modelo relacional de una base de datos. Para otro tipo de fuentes de datos –ficheros, directorio LDAP–, se utilizará el formato que se considere más adecuado para su descripción.

### ■ Arquitectura de aplicación

Se describirá la arquitectura en componentes de la aplicación, y su relación entre ellos.

### ■ Fuentes Java

Se documentará adecuadamente los fuentes Java en el formato de la herramienta Javadoc, generando la documentación a partir de los fuentes comentados en el proceso de compilación de la aplicación. La documentación incluirá la descripción de todas las clases y métodos.

Se puede encontrar más información sobre la herramienta Javadoc y su utilización en la página <http://java.sun.com/j2se/javadoc/writingdoccomments/> "How to Write Doc Comments for the Javadoc Tool".

### ■ *Administración de la aplicación*

Si la aplicación requiere tareas de administración, se describirán en este documento.

### ■ *Errores de aplicación*

Se realizará una descripción de los posibles errores que se produzcan en la aplicación, incluyendo para cada uno código de error, texto descriptivo del error y la actuación necesaria, en caso de producirse, para su corrección.

La pantalla de error que se muestre al usuario deberá ser suficientemente descriptiva del error para permitir su identificación, mostrando su código y texto descriptivo.

### ■ *Plan de pruebas*

Se especificará el plan de pruebas de la aplicación: pruebas funcionales, de carga y de alta disponibilidad.

### ■ *Manual de usuario*

Se entregará el manual de usuario detallado de la aplicación.