GOBIERNO DE ESPAÑA
MINISTERIO DE HACIENDA Y ADMINISTRACIONES PÚBLICAS
SUBSECRETARÍA
DIRECCIÓN GENERAL DE RELACIONES INSTITUCIONALES Y ORGANIZACIÓN
DIVISIÓN DE SISTEMAS DE LA INFORMACIÓN Y COMUNICACIONES

DSIC
DIVISION DE SISTEMAS DE INFORMACION Y COMUNICACIONES

InSide – Manual de Instalación

Gestión de Expedientes y Documentos Electrónicos

Documento de Integración
Sistemas Desarrollo
Versión del documento
004
Fecha de revisión
04/11/2013
Realizado por
Sistemas Desarrollo

ÍNDICE

1 Control de modificaciones.....3

2 Objetivo de este documento.....4

3 Requisitos previos.....5

4 Instalación.....6

4.1 Base de datos.....6

Instalación de InSide (Oracle).....6

Instalación de InSide (MySQL).....6

Inserción de aplicación de prueba.....6

4.2 Ficheros de configuración.....7

server.cmis.properties.....7

log4j.properties.....7

database.properties (Oracle).....8

database.properties (MySQL).....8

firma.properties.....9

infofirma.properties.....9

visualizacion.properties.....10

Directorio schemas.....10

4.3 Instalación de Inside a partir del código fuente.....11

Descarga de dependencias:.....11

Ejecutar Inside en entorno local.....12

Construcción del WAR a partir del código fuente.....13

5 Migración desde una instalación anterior (Inside 1.1_b6 o anteriores).....14

6 Alta de nuevas aplicaciones.....15

7 Web Services externos a Inside.....17

7.1 Servicio de firma en servidor.....17

firmaFichero.....17

7.2 Servicio de obtención de información de firmas.....18

obtenerInformacionFirma.....18

7.3 Servicio de visualización.....19

visualizar.....20

1 Control de modificaciones

Revisión Actual: 004
Fecha: 04/11/2013
Autor: DSIC - Sistemas Desarrollo
Descripción:

- Se incluyen notas relativas a los requisitos para la instalación de la aplicación en el apartado 3.
- Se incluye el apartado 5 explicando los pasos a seguir para realizar la migración de una versión antigua de Inside (versión igual o anterior a la v1.1_b6) a la nueva.
- Se incluye información sobre cómo gestionar aplicaciones consumidoras de InSide. Apartado 6.
- Se modifica el apartado 4.2, especificando que el valor de la propiedad `server.cmis.url` en el fichero `server.cmis.properties` debe apuntar al XML donde se encuentre la descripción del servicio CMIS.

Revisión Anterior: 003
Fecha: 02/08/2013
Autor: DSIC - Sistemas Desarrollo
Descripción:

- Se modifica el apartado 4.1 en la descripción del proceso de instalación, incluyendo las instrucciones necesarias para la instalación de InSide en Oracle y MySQL.
- Se modifica el apartado 4.2 en la descripción del proceso de instalación, incluyendo las instrucciones necesarias para la configuración del fichero de propiedades `database.properties` para una base de datos Oracle y una base de datos MySQL.

Revisión Anterior: 002
Fecha: 11/04/2013
Autor: DSIC - Sistemas Desarrollo
Descripción: Se cambia la fuente Courier 10 Pitch por FreeMono

Revisió Anterior: 001
Fecha: 18/02/2013
Autor: DSIC - Sistemas Desarrollo
Descripción: Versión inicial del documento

2 Objetivo de este documento

Este documento pretende ser una guía de instalación del sistema Inside.

Para conseguir el WAR (fichero de despliegue de aplicaciones Web) existen dos alternativas:

- Descargarlo directamente.
- Realizar la compilación del código fuente y posterior empaquetamiento.

Si se ha descargado el WAR directamente puede ignorarse el punto 4.3

3 Requisitos previos

Para poder realizar una instalación correcta y completa de Inside, es necesario que el sistema cumpla los siguientes requisitos:

- Base de datos: MySQL. Versión 5.5.9 u Oracle 9i, 10g, 11g. **La aplicación no es compatible con la versión 5.6 de MySQL.**
- Sistema de gestión de contenidos que implemente CMIS.
- JDK 1.6.0_26 o superior. Con JDK 1.7 es necesario sustituir la librería
- Servidor de aplicaciones Apache Tomcat (versión 6 o superior), o similar.
- Apache Maven 2 (si se desea trabajar con el código fuente).

Notas:

- Por cuestiones de licenciamiento el driver necesario para la ejecución de Inside sobre Oracle no se puede distribuir. Puede obtenerse el driver JDBC de Oracle en la siguiente dirección: http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html

Una vez se ha obtenido el driver, será necesario añadirlo a las librerías de la aplicación, que se encuentran bajo el directorio *WEB-INF/lib.*

- Si se utiliza JDK 1.7 es necesario sustituir la librería `aspectjweaver`, en concreto, habrá que sustituir la versión 1.6.2 por la versión 1.6.12, que es compatible con Java 7.

Esta librería puede obtenerse en la siguiente dirección:
http://repo1.maven.org/maven2/org/aspectj/aspectjweaver/1.6.12/aspectjweaver-1.6.12.jar

4 Instalación

En este apartado se detallan los pasos para realizar la instalación de InSide.

4.1 Base de datos

En este apartado se describen los pasos para realizar la instalación de InSide en los dos SGBD para los que la aplicación es compatible.

Instalación de InSide (Oracle)

En primer lugar es necesario crear las tablas y objetos necesarios para el funcionamiento de Inside. Estos scripts podemos encontrarlos en los directorios:

```
resources/scripts_bbdd/oracle/
resources/scripts_bbdd/comun/
```

El orden a seguir para la ejecución de los scripts es el siguiente:

- `resources/scripts_bbdd/oracle/1_creacion_tablas_oracle.sql`
- `resources/scripts_bbdd/oracle/2_foreingKeys_oracle.sql`
- `resources/scripts_bbdd/oracle/3_funcion_oracle.sql`
- `resources/scripts_bbdd/oracle/4_trigger_oracle.sql`
- `resources/scripts_bbdd/comun/insert_aplicacion.sql`
- `resources/scripts_bbdd/oracle/5_inserts_GeneradorClaves.sql`

Instalación de InSide (MySQL)

En primer lugar es necesario crear las tablas y objetos necesarios para el funcionamiento de Inside. Estos scripts podemos encontrarlos en los directorios:

```
resources/scripts_bbdd/mysql/
resources/scripts_bbdd/comun/
```

El orden a seguir para la ejecución de los scripts es el siguiente:

- `resources/scripts_bbdd/mysql/1_creacion_objetos_mysql.sql`
- `resources/scripts_bbdd/comun/insert_aplicacion.sql`
- `resources/scripts_bbdd/mysql/2_inserts_GeneradorClaves_mysql.sql`

Inserción de aplicación de prueba

Tras ejecutar los scripts proporcionados se ha creado una aplicación de prueba con las credenciales:

- **idaplicacion**: apli

- **password**: pass

4.2 Ficheros de configuración

Los ficheros de configuración se proporcionan en la ruta `resources/config` deben guardarse en el mismo directorio. Además, este directorio debe indicarse en el arranque de la aplicación, asignando la ruta a la variable de entorno `config.path`:

```
java ….. -Dconfig.path=<ruta_ficheros_configuracion>
```

server.cmis.properties

En este fichero se configuran las propiedades del acceso al gestor de contenidos (que debe implementar CMIS).

```
server.cmis.url=http://host:port/path
server.cmis.pathRootFolder=TEST
server.cmis.username=admin@inside
server.cmis.password=inside
server.cmis.connectTimeout=60000
server.cmis.sessionTimeout=60000
```

Los campos a rellenar son los siguientes:

- **server.cmis.url**: Url del XML de descripción del servicio CMIS.
- **server.cmis.pathRootFolder**: Nombre de la carpeta padre. En esa carpeta se crearán las carpetas y documentos que almacene Inside.
- **server.cmis.username**: Usuario para conectarse al gestor de contenidos.
- **server.cmis.password**: Password para conectarse al gestor de contenidos.
- **server.cmis.connectTimeout**: Máximo tiempo de espera para establecer la conexión.
- **server.cmis.sessionTimeout**: Máximo tiempo de duración de sesión.

Los metadatos obligatorios de expedientes y documentos serán almacenados también en el gestor documental. Se adjunta en la distribución el fichero `insideModel.xml`, en el directorio `resources/alfresco`, que contiene el modelo para Alfresco. Si se utiliza otro gestor documental deberá crearse el modelo equivalente.

log4j.properties

En este fichero se configuraran las propiedades de log de la aplicación. El que se proporciona en la distribución contiene tres *appender:* consola, fichero y correo. No obstante podrá configurarse de otro modo si así se desea. Para una instalación inicial de pruebas se recomienda eliminar los *appenders* `file` y `mail` y dejar únicamente `stdout`, por lo que el fichero quedaría de la siguiente forma:

```
# Root logger option
log4j.rootLogger= WARN, stdout
log4j.logger.es.mpt= DEBUG, stdout
log4j.additivity.es.mpt=false
```

```
# Direct log messages to stdout
log4j.appender.stdout=org.apache.log4j.ConsoleAppender
log4j.appender.stdout.Target=System.out
log4j.appender.stdout.layout=org.apache.log4j.PatternLayout
log4j.appender.stdout.layout.ConversionPattern=%d{ABSOLUTE} %5p %c{1}:%L -
%m%n
```

database.properties (Oracle)

En este fichero se configuran las propiedades de la base de datos.

```
database.driverClassName=oracle.jdbc.driver.OracleDriver
database.url=jdbc:oracle:thin:@host:port:SID
database.username=XXX
database.password=XXX
database.showSql=0
database.hibernateDialect=org.hibernate.dialect.Oracle10gDialect
```

El driver proporcionado funciona para la versión 10g de Oracle. Si se desea instalar InSide sobre Oracle 11 será necesario descargar el driver e incorporarlo a las librerías de la aplicación.

Los campos a rellenar son los siguientes:

- **database.driverClassName**: Driver de conexión.
- **database.url**: URL de conexión a la BBDD.
- **database.username**: Usuario de conexión a la BBDD.
- **database.password**: Password de conexión a la BBDD.
- **database.showSql**: Si se quieren mostrar las consultas que se hacen a la BBDD deberá ponerse a “1”.
- **database.hibernateDialect**: Se deberá indicar esta propiedad atendiendo a la versión de Oracle de la BBDD.

database.properties (MySQL)

En este fichero se configuran las propiedades de la base de datos.

```
database.driverClassName=com.mysql.jdbc.Driver
database.url=jdbc\:mysql\://host\:port/database
database.username=XXX
database.password=XXX
database.showSql=0
database.hibernateDialect=org.hibernate.dialect.MySQLDialect
```

El driver proporcionado se ha probado para la versión 5.5.9 de MySQL. Si se desea instalar InSide sobre una versión de MySQL no compatible con este driver deberá incorporarse el driver adecuado a las dependencias del proyecto.

Los campos a rellenar son los siguientes:

- **database.driverClassName**: Driver de conexión. Si se modifica se deberá aportar el jar correspondiente a las dependencias del proyecto.
- **database.url**: URL de conexión a la BBDD.
- **database.username**: Usuario de conexión a la BBDD.
- **database.password**: Password de conexión a la BBDD.
- **database.showSql**: Si se quieren mostrar las consultas que se hacen a la BBDD deberá ponerse a "1".
- **database.hibernateDialect**: Se deberá indicar esta propiedad atendiendo a la versión de Oracle de la BBDD.

firma.properties

Para que Inside tenga la funcionalidad de firma en servidor, se deberá implementar un Web Service que cumpla una especificación determinada. Ver punto 7.1

En el fichero firma.properties se configuran las propiedades de acceso a dicho Web Service:

```
firma.activo=S
firma.url=http://host:port/eeutil/ws/EeUtilFirmarService?wsdl
firma.idaplicacion=xxx
firma.password=xxx
firma.cadena.ok=OK
```

Los campos a rellenar son:

- **firma.activo**: Informa de si el Web Service está activo o no. Valores posibles: S ó N. Si no está activo Inside no hará llamadas a este Web Service.
- **firma.url**: URL del Web Service de firma remota.
- **firma.idaplicacion**: Identificador de la aplicación Inside en el servicio de firma remota.
- **firma.password**: Password de la aplicación Inside en el servicio de firma remota.
- **firma.cadena.ok**: Cadena que devolverá el servicio de firma remota cuando una firma se ha realizado correctamente.

infofirma.properties

Si se desea que Inside sea capaz de obtener información (número de firmas, tipo de firmas) para componer documentos ENI a partir de un documento firmado previamente es necesario implementar un Web Service que cumpla una especificación determinada. Ver punto 7.2

En el fichero `infofirma.properties` se configuran las propiedades de acceso a dicho Web Service:

```
infofirma.activo=S
infofirma.url=http://host:port/eeutil/ws/EeUtilService?wsdl
infofirma.idaplicacion=xxx
infofirma.password=xxx
```

Los campos a rellenar son:

- **infofirma.activo**: Informa de si el Web Service está activo o no. Valores posibles: S ó N. Si no está activo Inside no hará llamadas a este Web Service.
- **infofirma.url**: URL del Web Service de obtención de información de firmas.
- **infofirma.idaplicacion**: Identificador de la aplicación Inside en el servicio de obtenciónd e firmas.
- **infofirma.password**: Password de la aplicación Inside en el servicio de obtención de firmas.

visualizacion.properties

La visualización del índice del expediente, la visualización del documento y el foliado de expedientes tendrá que implementarse en un Web Service que cumpla con una especificación determinada. Ver punto 7.3

En el fichero visualizacion.properties se configuran las propiedades de acceso a dicho Web Service:

```
visualizacion.activo=S
visualizacion.url=http://host:port/eeutil/ws/EeUtilService?wsdl
visualizacion.idaplicacion=xxx
visualizacion.password=xxx
visualizacion.nombre.organismo=MINISTERIO;DE HACIENDA;Y ADMINISTRACIONES PÚBLICAS
visualizacion.modelo.expediente=modelo1
visualizacion.modelo.documento=modelo2
```

Los campos a rellenar son los siguientes:

- **visualizacion.activo**: Informa de si el Web Service está activo o no. Valores posibles: S ó N. Si no está activo Inside no hará llamadas a este Web Service.
- **visualizacion.url**: URL del Web Service de visualización.
- **visualizacion.idaplicacion**: Identificador de la aplicación en el Web Service de visualización.
- **visualizacion.password**: Password de la aplicación Inside en el Web Service de visualización.
- **visualización.nombre.organismo**: Nombre del organismo emisor de expedientes o documentos ENI por defecto. Se puede ignorar por parte del Web Service, pero si se desea se podrá utilizar para estamparlo en la cabecera o el pie de página de la visualización, o para cualquier fin.
- **visualizacion.modelo.expediente**: Será la cadena que Inside envíe al Web Service de visualización para informar de que se desea obtener la visualización del índice de un expediente.
- **visualizacion.modelo.documento**: Será la cadena que Inside envíe al Web Service de visualización para informar de que se desa obtener la visualización de un documento.

Directorio schemas

Bajo este directorio deben almacenarse todos los esquemas XSD del documento y expediente ENI para la correcta validación de éstos.

- `ContenidoDocumentoEni.xsd`
- `DocumentoEni.xsd`
- `ExpedienteEni.xsd`
- `firmasEni.xsd`
- `IndiceContenidoExpedienteEni.xsd`
- `IndiceExpedienteEni.xsd`
- `MetadatosDocumentoEni.xsd`
- `MetadatosExpedienteEni.xsd`
- `xmldsig-core-schema.xsd`

4.3 Instalación de Inside a partir del código fuente

El código fuente de Inside se distribuye en distintos módulos:

- `inside-mvn-base`: Proyecto "padre", no contiene código fuente, tan sólo las propiedades y dependencias comunes.
- `inside-services`: Encargado de la lógica de negocio de Inside.
- `integration-cmis`: Encapsula las clases encargadas de interaccionar con el gestor documental.
- `infofirma-service`: Encapsula las clases encargadas de interaccionar con el servicio de obtención de información de firmas.
- `inside-model`: Contiene las clases de entrada y salida de los web services y de los servicios de inside.
- `inside-cliente-firma`: Contiene las clases del cliente del Web Service de firma remota en servidor.
- `eeutil-client`: Contiene las clases del cliente del Web Service de generación de visualización de índice del expediente y visualización del documento, y del Web Service de obtención de información de firmas.
- `inside-ws`: Contiene la capa de Web Service de la aplicación.

Descarga de dependencias:

La descarga de las librerías e instalación de dependencias se puede realizar instalando los proyectos en el siguiente orden:

- `inside-mvn-base`
- `integration-cmis`
- `eeutil-client`
- `inside-cliente-firma`
- `infofirma-service`

- `inside-model`
- `inside-services`
- `inside-ws`

Nota: Para que compile el proyecto inside-ws, así como para que su ejecución se realice de forma correcta es necesario que la JRE tenga instaladas dos librerías en el directorio `jre/lib/endorsed`. Estas librerías se proporcionan en la carpeta `resources/endorsed_lib` y son:

- `jaxb-api-2.2.jar`
- `jaxws-api-2.2.jar`

Ejecutar Inside en entorno local

Una vez se han rellenado correctamente los ficheros de configuración, descargado las dependencias y compilado el código, la aplicación puede ser ejecutada en un entorno de servidor local, ejecutando la siguiente orden en el directorio `inside-ws` del proyecto:

```
mvn -Dconfig.path=<ruta_ficheros_configuracion> clean jetty:run
```

Se puede deducir que la aplicación ha arrancado correctamente si en la consola no aparecen errores y si se puede acceder desde un navegador a la dirección:

```
http://localhost:8090/inside/ws
```

Se mostrará en esta dirección una lista con los servicios disponibles:

- InsideWS → Web Service para gestión de documentos y expedientes electrónicos.
- GinsideWS → Web Service para generación y validación de documentos y expedientes ENI, generación de visualización de documento.

Available SOAP services:

GInSideWebService • convertirDocumentoAEni • convertirDocumentoAEniConMAdicionales • validarDocumentoEniFile • convertirExpedienteAEniConMAdicionales • convertirExpedienteAEni • visualizarDocumentoEni • validarExpedienteEniFile	Endpoint address: http://localhost:8090/inside/ws/GInsideService WSDL : {https://ssweb.seap.minhap.es/Inside/XSD/v1.0/WebService}GInsideWS Target namespace: https://ssweb.seap.minhap.es/Inside/XSD/v1.0/WebService
InSideWebService • getFicheroFoliado • getDocumentoContenido • vincularExpediente • getMetadatosDocumento • cambiaMetadatosExpediente • importarExpediente • altaDocumentoEni • altaCarpetaEnExpediente • convertirDocumentoAEni • altaDocumentoOriginal • getExpedienteMetadatos • getVersionesExpediente • crearVistaAbiertaExpediente • modificaDocumento • altaDocumentoCambioFormato • borrarCarpetaEnExpediente • altaDocumentoDigitalizado • modificaExpedienteEni • altaDocumentoCopiaParcial • getExpedienteEni • getDocumentoEni • getDocumento • convertirExpedienteAEni • getVersionesDocumento • getIndiceExpediente • altaExpediente • crearVistaCerradaExpediente • buscarDocumentos • getExpediente • buscarExpedientes • vincularDocumentosEnExpediente • cambiaEstadoExpediente • altaExpedienteEni	Endpoint address: http://localhost:8090/inside/ws/InsideService WSDL : {https://ssweb.seap.minhap.es/Inside/XSD/v1.0/WebService}InsideWS Target namespace: https://ssweb.seap.minhap.es/Inside/XSD/v1.0/WebService

Construcción del WAR a partir del código fuente

Para crear el WAR a partir del código fuente se ejecuta la siguiente orden en el directorio `inside-ws` del proyecto:

```
mvn clean package
```

En el directorio `target` del proyecto `inside-ws` se habrá creado el fichero de despliegue `inside.war`, que contendrá dentro todas las librerías necesarias para la ejecución de la aplicación (exceptuando las librerías que se tienen que colocar en el directorio `endorsed` de la JRE).

5 Migración desde una instalación anterior (Inside 1.1_b6 o anteriores)

Para que la aplicación pueda ser instalada utilizando tanto una BBDD Oracle o MySQL, algunas tablas del modelo de datos han tenido que ser modificadas.

Si se realizó una instalación previa de la aplicación (versión 1.1_b6 o anteriores), es necesario ejecutar unos scripts en la BBDD para poder instalar la última versión.

Los scripts a ejecutar se encuentran en la ruta:

- `resources/scripts_bbdd/migracion/1_creacionTablaGeneradorClaves.sql`
- `resources/scripts_bbdd/migracion/2_renombrarObjetosMySQL.sql`
- `resources/scripts_bbdd/migracion/3_inserts_GeneradorClave_mysql.sql`

Una vez se hayan ejecutado los scripts la BBDD estará lista para desplegar el WAR de la nueva versión de InSide.

6 Alta de nuevas aplicaciones

Las aplicaciones consumidoras de los Web Services de Inside y de Ginside deberán estar dadas de alta en la tabla `InsideWsAplicacion`, que tiene los siguientes campos:

- **id** : clave primaria
- **nombre**: identificador de la aplicación
- **passwordHash**: hash del password de la aplicación, calculado con el algoritmo sha-256.
- **activo**: Indica si la aplicación está activa, si su valor es `false` no le dejará realizar ninguna operación
- **altaExpediente**: Indica si la aplicación podrá dar de alta expedientes.
- **modificarExpediente**: Indica si la aplicación podrá modificar expedientes.
- **leerExpediente**: Indica si la aplicación podrá leer expedientes.
- **altaDocumento**: Indica si la aplicación podrá dar de alta documentos.
- **modificarDocumento**: Indica si la aplicación podrá modificar documentos.
- **leerDocumento**: Indica si la aplicación podrá leer documentos.
- **fechaCreacion**: Indica la fecha de creación de la aplicación.

La gestión de la tabla `InsideWSAplicación` deberá hacerse directamente en la base de datos. El alta de cada nueva aplicación deberá realizarse creando un nuevo registro:

```
INSERT INTO InsideWsAplicacion (
                                id,

                                nombre,

                                passwordHash,

                                activo,

                                altaExpediente,

                                modificarExpediente,

                                leerExpediente,

                                altaDocumento,

                                modificarDocumento,

                                leerDocumento)
            values

                                   (
                                2,

                                <id_aplicacion>,

                                <sha-256_password>,

                                1, -- activo
```

```
1, -- altaExp
1, -- modifExp
1, -- leerExp
1, -- altaDoc
1, -- modifDoc
1); -- leerDoc
```

7 Web Services externos a Inside

Algunas funcionalidades de Inside dependen de la implementación de Web Services externos. Estos Web Services tienen que cumplir una especificación determinada.

7.1 Servicio de firma en servidor

Para que los documentos y el índice de los expedientes puedan ser firmados en servidor (sellados) se ha de implementar un Web Service que cumpla con el WSDL que se encuentra en la ruta `Web Services externos/firma-remota`. También se proporciona en este directorio el esquema XSD con la definición de los tipos de entrada y de salida.

Este servicio tiene dos métodos:

- firmaContenido: firma una cadena de texto. En la versión actual de Inside no se invoca a este método.
- **firmaFichero**: firma un array de bytes.

A continuación se describe la entrada y la salida de cada uno de los métodos a los que invoca la aplicación Inside.

`firmaFichero`

Los parámetros de entrada son los siguientes:

- **aplicacionInfo**:
  - **idAplicacion**: Inside enviará el valor firma.idaplicacion que encuentra en el fichero de configuración firma.properties
  - **password**: Inside enviará el valor firma.password que encuentre en el fichero de configuración firma.properties

- **datosEntrada**:
  - **contenido**: Inside enviará los bytes del contenido que se desea firmar.
  - **formatoFirma**: Inside enviará la cadena de texto CAdES para la firma del índice del expediente. En el caso de la firma de documentos se enviarán las siguientes cadenas:
    - Adobe PDF: El servicio deberá devolver una firma PAdES del contenido enviado a firmar
    - XAdES Enveloped: El servicio deberá devolver una firma XadES Enveloped del XML enviado a firma.
    - XAdES Detached: El servicio deberá devolver una firma XadES Detached del contenido enviado a firma.
  - **modoFirma**: Se envía IMPLICIT para el caso de la firma del expediente. Para el caso del documento se envía a nulo. Se supone que el formato por defecto será implícito.
  - **algoritmoFirma**: Se envía a nulo, se supone el formato por defecto configurado en el servicio.

- **cofirmarSiFirmado**: Indica si se debe cofirmar el documento en caso de que lo que se envíe sea un documento firmado. Inside siempre envía "false", para que se contrafirme.

La salida del servicio se corresponde con el tipo `datosSalida`, que tiene los siguientes elementos:

- **estado**: Cadena que informa del estado de la operación. Inside tomará la operación como correcta si esta cadena coincide con el valor de la propiedad `firma.cadena.ok` del fichero de configuración `firma.properties`.

- **resultado**: Descripción del resultado de la operación. Inside lo ignora

- **datosResultado**
  - **formatoFirma**: Formato de firma. Inside lo ignora.
  - **modoFirma**: Inside lo ignora.
  - **algoritmoFirma**: Inside lo ignora.
  - **contenido**: Bytes del contenido que se ha firmado en base64. Inside lo ignora.
  - **firma**: Bytes de la firma en base 64.
  - **fechaFirma**: Fecha de la firma. Inside lo ignora.
  - **datosFirmante**: Inside lo ignora
    - **nombreFirmante**. Inside lo ignora.
    - **numeroIdentificacion**. Inside lo ignora
    - **mailFirmante**. Inside lo ignora.
    - **entidadCertificadora** Inside lo ignora.

7.2 Servicio de obtención de información de firmas

Para que Inside pueda componer documentos ENI a partir de documentos firmados previamente es necesario que se implemente el Web Service de obtención de información de firmas.

La especificación que tiene que cumplir este servicio se proporciona en la ruta `Web Services externos/infofirma`.

Este servicio tiene un sólo método a implementar, `obtenerInformacionFirma`, cuyos parámetros de entrada y salida se describen en el siguiente apartado.

`obtenerInformacionFirma`

Los parámetros de entrada son los siguientes:

- **aplicacionInfo**:
  - **idAplicacion**: Inside enviará el valor `infofirma.idaplicacion` que encuentra en el fichero de configuración `infofirma.properties`
  - **password**: Inside enviará el valor `infofirma.password` que encuentre en el fichero de configuración `infofirma.properties`

- **Firma**: Firma en base 64 de la que se quiere obtener información.

- **opcionesObtenerInformacionFirma**:
  - **obtenerFirmantes**: Inside envía true para obtener información de los firmantes.
  - **ObtenerDatosFirmados**: Inside envía true para obtener los bytes firmados.
  - **obtenerTipoFirma**: Inside envía true para obtener el tipo de firma.

La salida atiende al tipo `resultadoObtenerInformacionFirmas`, compuesto por los siguientes elementos:

- **esFirma**: Indica si los bytes enviados se corresponden con una firma. Si se devuelve false Inside levantará una excepción.
- **tipoDeFirma**:
  - **tipoDeFirma**: Indica el tipo de firma. Las cadenas esperadas por Inside son:
    - XAdES Detached
    - XAdES Enveloped
    - PAdES
    - CAdES
  - **modoFirma**: Inside lo ignora.
- **documentoFirmado**: En caso de que lo que se esté firmando no sea un hash, el servicio devolvería este elemento.
  - **contenido**: Bytes en base 64 que se han firmado.
  - **tipoMime**: Mime del documento que se ha firmado.
- **hashFirmado**: En caso de que lo que se haya firmado sea un hash, el servicio devolvería este elemento en lugar de documentoFirmado. Si Inside obtiene este elemento en la respuesta levantará una excepción, ya que sólo admite firmas implícitas.
- **algoritmoHashFirmado**: Algoritmo con el que se calculó el hash.
- **firmantes**: Información sobre los firmantes. Esta información será utilizada para generar la visualización del documento (será enviada en la petición de visualización).
  - **informacionFirmas**
    - **informacionFirmas**: Un elemento de este tipo por cada firma.
      - **nifcif**: Nif o Cif del firmante.
      - **nombre**: Nombre del firmante.
      - **apellido1**: Primer apellido del firmante
      - **apellido2**: Segundo apellido del firmante.
      - **fecha**: Fecha de firma
      - **extras**: Las cadenas que espera Inside son:
        - F: Inside interpretará que se trata de una firma.
        - CF: Inside interpretará que se trata de una cofirma.
        - XF: Inside interpretará que se trata de una contrafirma.

7.3 Servicio de visualización

Para que Inside pueda generar la visualización del índice de un expediente ENI y visualización de documentos ENI es necesario implementar un servicio que cumpla con una especificación determinada.

La especificación se encuentra en la ruta `Web Services externos/visualizacion`

Para la visualización del índice del expediente y el documento habrá que implementar el método `visualizar`

`visualizar`

Aunque Inside llama al mismo servicio para obtener visualización del índice de los expedientes y visualización de documentos, los parámetros de entrada se llenan de una forma u otra atendiendo a lo que se desea visualizar.

La salida atiende al siguiente tipo:

- **SalidaVisualizacion**:
  - **DocumentoContenido**
    - **bytesDocumento**: Bytes del documento de visualización, en base 64.
    - **mimeDocumento**: Mime del documento de visualización.

7.3.1.1 Parámetros de entrada para la visualización del índice de un expediente

- **aplicacionInfo**:
  - **idAplicacion**: Inside enviará el valor `visualizacion.idaplicacion` que encuentre en el fichero de configuración `visualizacion.properties`
  - **password**: Inside enviará el valor `visualizacion.password` que encuentre en el fichero de configuración `visualizacion.properties`
- **item**: En el caso de un expediente, se enviará un elemento de este tipo conteniendo los metadatos del expediente en las propiedades y tantos elementos hijos como carpetas, subexpedientes o documentos tenga el expediente.
  - **Identificador**: Inside envía lo siguiente dependiendo del tipo de ítem:
    - El identificador del expediente en caso de tratarse del expediente padre.
    - Identificador aleatorio en caso de tratarse de un expediente importado o vinculado, una carpeta indizada o un documento indizado.
  - **Nombre**: Inside envía lo siguiente dependiendo del tipo de ítem
    - El identificador del expediente en caso de tratarse del expediente padre.
    - La cadena “Expediente Electrónico” + `<identificador>` en caso de tratarse de un expediente importado o vinculado.
    - El identificador de la carpeta indizada en caso de tratarse de una carpeta indizada.
    - El nombre del documento si el documento cuenta con ese metadato adicional. En caso contrario se envía el identificador del documento indizado.

- **Padre**: Inside envía `null` este elemento para el caso del item padre. En caso de ser otro item, apuntaría al item padre (identificador de carpeta o expediente en un nivel superior).

- **Hijos**: Estructura jerárquica para representar los hijos del expediente, que serán carpetas, subexpedientes o documentos.
  - **items**
    - **item**

- **documentoContenido**: En el caso del expediente Inside no envía nada.
- **PropiedadesItem**
  - **propiedades**
    - **propiedad**
      - **clave**
      - **valor**
      - **importancia**

En el caso de ser el item padre, se enviarán los metadatos del expediente como las siguientes tuplas clave/valor/importancia (cada una de ellas un elemento propiedad).

- `Referencia / valor metadato identificador / 0`
- `Descripcion / valor metadato descripcion / 0`
- `Organos / valor metadato organo / 0`
- `Procedimiento / valor metadato clasificacion / 0`
- `Descripcion Procedimiento / valor metadato DescripcionProcedimiento / 0`
- `Fecha Apertura Expediente ENI / FechaAperturaExpediente / 0`
- `Interesados / interesado / 0`

Si los metadatos adicionales Descripcion y DescripcionProcedimiento no existen se enviarán cadenas vacías en su lugar.

En caso de ser un item que corresponda a un subexpediente importado o vinculado, no se enviarán propiedades.

En caso de ser un item que corresponda a un documento se enviarán las siguientes tuplas:

- `Orden / orden del documento / 0`
- `Hash / hash del documento / 0`

- **opcionesVisualizacion**: Pensadas para formar la cabecera del documento de visualización
  - **opcionesLogo**
    - **estamparLogo**: Inside envía `true`
    - **estamparNombreOrganismo** Inside envía `true`
    - **listaCadenasNombreOrganismo**: Inside dividirá el valor que encuentre en la propiedad `visualizacion.nombre.organismo` en elementos cadenas, separando la propiedad por los caracteres ";" que encuentre

        - **cadenas**
            - **cadenas**

- **modelo**: Inside enviará el valor de la propiedad `visualizacion.modelo.expediente`

7.3.1.2 Parámetros de entrada para la visualización de un documento

- **aplicacionInfo**:
    - **idAplicacion**: Inside enviará el valor `visualizacion.idaplicacion` que encuentre en el fichero de configuración `visualizacion.properties`
    - **password**: Inside enviará el valor `visualizacion.password` que encuentre en el fichero de configuración `visualizacion.properties`

- **item**: En el caso de un documento, el item padre será el que contenga la información del documento (metadatos y contenido), y los ítems hijos la información de las firmas.
    - **Identificador**: Inside envía lo siguiente dependiendo del tipo de ítem:
        - El identificador del documento en caso de tratarse del ítem padre.
        - Identificador aleatorio en caso de tratarse de los ítems que representan a las firmas.
    - **Nombre**: Inside envía lo siguiente dependiendo del tipo de ítem
        - El identificador del documento en caso de tratarse del ítem padre.
        - El tipo de firma en caso de tratarse de los ítems que representan a las firmas.
    - **Padre**: Inside envía `null` este elemento para el caso del item padre (ítem del documento). En caso de ser algún ítem que represente a las firmas, apuntaría al ítem padre (el identificador del documento).
    - **Hijos**: En este caso la estructura jerárquica contiene tan sólo dos nieveles, el nivel superior (o el del documento) y un segundo nivel para representar las firmas.
        - **items**
            - **item**
    - **documentoContenido**:
        - **bytesDocumento**: bytes del documento codificado en base 64.
        - **mimeDocumento**: mime del documento.
    - **PropiedadesItem**
        - **propiedades**
            - **propiedad**
                - **clave**
                - **valor**
                - **importancia**

En el caso de ser el item padre, se enviarán los metadatos del documento como las siguientes tuplas clave/valor/importancia (cada una de ellas un elemento propiedad)

- `Versión NTI / valor metadato versionNTI / 0`
- `Identificador / valor metadato identificador / 0`
- `Organos / valor metadato organo / 0`
- `Fecha Captura / valor metadato fecha captura / 0`
- `Origen / valor metadato origen / 0`
- `Estado Elaboracion / valor metadato E.Elaboracion / 0`
- `Tipo Documental / valor metadato T.Documental / 0`

En caso de ser el item de una firma CSV se enviarán las siguientes tuplas:

- `valorCSV / valor del CSV / 0`
- `regulacionCSV / valor de regulacion CSV / 0`

En caso de ser el item de una firma no CSV se enviarán las siguientes tuplas:

- `nombreFirmante / nombre del firmante / 0`
- `fechaFirma / fecha de la firma / 0`

- **opcionesVisualizacion**: Pensadas para formar la cabecera del documento de visualización
  - **opcionesLogo**
    - **estamparLogo**: Inside envía true
    - **estamparNombreOrganismo** Inside envía true
    - **listaCadenasNombreOrganismo**: Inside dividirá el valor que encuentre en la propiedad `visualizacion.nombre.organismo` en elementos `cadenas`, separando la propiedad por los caracteres ";" que encuentre
      - **cadenas**
        - **cadenas**

- **modelo**: Inside enviará el valor de la propiedad `visualizacion.modelo.documento`