



Unión Europea

Fondo Europeo de
Desarrollo Regional

“Una manera de hacer Europa”



COMPULSA

Manual de integración con repositorios para las Consejerías y entidades

Versión: v01r00

Fecha: 29/12/2015



Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso y por escrito de la Junta de Andalucía.

HOJA DE CONTROL

Título	Manual de integración con repositorios para las Consejerías y entidades		
Entregable	Manual de integración con repositorios para las Consejerías y entidades		
Nombre del Fichero	CMP430_MIC_IntegraciónRepositorios_v01r00.doc		
Autor	UTE		
Versión/Edición	v01r00	Fecha Versión	29/12/2015
Aprobado por	-	Fecha Aprobación	DD/MM/AAAA
		Nº Total Páginas	30

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Área	Fecha del Cambio
v01r00	Versión inicial	CHAP	-	29/12/2015

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos	Cargo	Área	Nº Copias
Manuel Perera Domínguez	Jefe de Servicio	Servicio de Coordinación de Administración Electrónica	1
Francisco González Guillén	Director de Proyecto	Servicio de Coordinación de Administración Electrónica	1
José Ignacio Cortés Santos	Director de Proyecto	Servicio de Coordinación de Administración Electrónica	1
Francisco Mesa Villalba	Director de Proyecto	Servicio de Coordinación de Administración Electrónica	1
Pedro José Casanova Luis	Jefe de Proyecto	UTE	1

ÍNDICE

1	INTRODUCCIÓN	5
2	INTEGRACIÓN DE LOS REPOSITORIOS	6
2.1	Incorporación de la librería “enidocWS”	7
2.1.1	Dependencias software	7
2.1.2	Fichero “web.xml”	8
2.1.3	Parámetros de configuración	9
2.1.3.1	Fichero de propiedades	9
2.1.3.2	Parámetros de contexto	9
2.1.4	Implementación de la clase invocada por el WS	11
2.1.4.1	A) Obtención del documento ENI	11
2.1.4.2	B) Incorporación del Documento ENI	13
2.2	Despliegue de la aplicación “enidocWS”	13
3	ANEXO I. COMPONENTE ENIDOCWS	14
3.1	WS obtener documento ENI	14
3.1.1	Flujo del Servicio de Integración	14
3.1.2	Estructura de la Petición	15
3.1.3	Estructura de la Respuesta	19
3.1.4	Gestión de respuestas y errores	22
3.2	WS almacenar documento ENI	23
3.2.1	Flujo del Servicio de Integración	23
3.2.2	Estructura de la Petición	23
3.2.3	Estructura de la Respuesta	24
3.2.4	Gestión de respuestas y errores	24
4	ANEXO II. GENERACIÓN DE CÓDIGO SEGURO DE VERIFICACIÓN	25
5	ANEXO III. COMPROBACIÓN DE LA INTEGRACIÓN CON ENIDOCWS	26
5.1	Postman: Cliente REST para Chrome	26
6	ANEXO IV. VALIDACIÓN DOCUMENTO ENI	29
7	REFERENCIAS	30

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Diagrama de integración con el Servicio Web.....	6
Ilustración 2. Diagrama de clases EnidocWS (verificarFirma)	14
Ilustración 3. Diagrama de clases Peticion	15
Ilustración 4. Diagrama de clases Respuesta EniDoc	19
Ilustración 5. Diagrama de clases EnidocWS (insertarDocumentoEni)	23
Ilustración 6. Diagrama de clases Respuesta DatosEni	24
Ilustración 7. Pantalla del complemento de Chrome	26
Ilustración 8. Formulario de autenticación	26
Ilustración 9. Especificar los parámetros de inserción.....	27
Ilustración 10. Respuesta correcta del servicio verificarFirma	27
Ilustración 11. Respuesta correcta del servicio insertarDocumentoEni	28

1 INTRODUCCIÓN

El objetivo del presente documento es servir de guía para los diferentes sistemas/repositorios que requieran integrarse con la herramienta informática genérica para la expedición de copias electrónicas auténticas, Compulsa, a través de las funcionalidades disponibles en la librería enidocWS.

Se describe “enidocWS”, como un componente de servicios web basados en REST, cuya operativa se centra en recuperar un Documento Electrónico con el formato definido por el Esquema Nacional de Interoperabilidad (en adelante, documento ENI) a partir de un identificador normalizado de documento o código seguro de verificación. Esto permite abstraer al integrador del desarrollo de un servicio web para tal fin y de la especificación del mismo.

Actualmente también se dispone de un servicio que permite realizar la custodia remota de un documento ENI devolviendo en caso de éxito el csv y el identificador normalizado.

2 INTEGRACIÓN DE LOS REPOSITORIOS

Para integrar el sistema/repositorio deseado con Compulsa a través del componente enidocWS, descrito en el Anexo I del presente documento, se plantean dos alternativas en función de las necesidades /limitaciones del propio sistema/repositorio, siendo las siguientes:

- 1) **Incorporando una librería** al sistema/repositorio. (véase 2.1).
- 2) **Desplegando una nueva aplicación** que actúa de fachada de interconexión entre la herramienta destino y la fuente de los datos (véase 2.2).

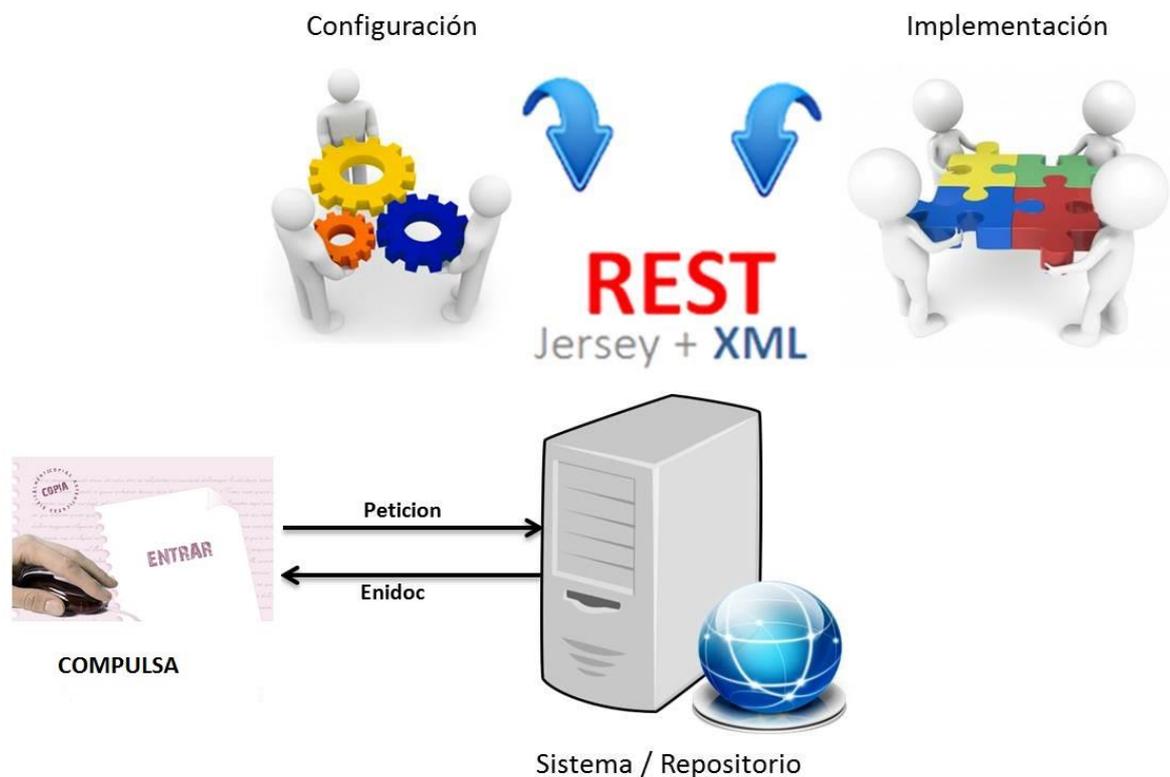


Ilustración 1. Diagrama de integración con el Servicio Web

NOTA: La obtención de los recursos necesarios para la integración se materializará mediante la zona de "descargas privadas" de la web de soporte de administración electrónica de la Junta de Andalucía. Para ello, se deberá realizar la correspondiente solicitud en el apartado "descargas de software" en la herramienta de "gestión de incidencias, consultas y solicitudes" disponible en la citada web.

2.1 Incorporación de la librería “enidocWS”

Si se desea utilizar este método de integración hay que tener en cuenta, como requisito técnico, que el sistema/repositorio debe ejecutarse sobre una **JDK 6 o superior**.

Para la integración del servicio web será necesario seguir los siguientes pasos:

2.1.1 Dependencias software

A continuación se exponen las dependencias software para la correcta integración con la librería, identificando las actuaciones a realizar según se disponga de una aplicación mavenizada o no.

A. Establecer dependencias sin maven: dirigido a aplicaciones no mavenizadas.

Son necesarias una serie de dependencias (ficheros jar) que deben estar en el classpath de ejecución de la librería principal para su correcto funcionamiento. La librería que implementa el servicio se denomina **enidocWS-x.x.jar** y las dependencias mencionadas son las siguientes:

- com.google.guava:guava:jar:14.0.1
- com.sun.xml.wsit:xmlsec:jar:1.1
- commons-lang:commons-lang:jar:2.3
- commons-logging:commons-logging:jar:1.1.1
- es.juntadeandalucia.nti:conversor-eni:jar:1.3.0
- javax.annotation:javax.annotation-api:jar:1.2
- javax.servlet:javax.servlet-api:jar:3.1.0
- javax.validation:validation-api:jar:1.1.0.Final
- javax.ws.rs:javax.ws.rs-api:jar:2.0
- javax.xml.stream:stax-api:jar:1.0-2
- log4j:log4j:jar:1.2.17
- org.codehaus.castor:castor-core:jar:1.3.2
- org.codehaus.castor:castor-xml:jar:1.3.2
- org.glassfish.hk2.external:asm-all-repackaged:jar:2.2.0-b21
- org.glassfish.hk2.external:cglib:jar:2.2.0-b21
- org.glassfish.hk2.external:javax.inject:jar:2.2.0-b21
- org.glassfish.hk2:hk2-api:jar:2.2.0-b21
- org.glassfish.hk2:hk2-locator:jar:2.2.0-b21
- org.glassfish.hk2:hk2-utils:jar:2.2.0-b21
- org.glassfish.hk2:osgi-resource-locator:jar:1.0.1
- org.glassfish.jersey.containers:jersey-container-servlet-core:jar:2.4.1
- org.glassfish.jersey.core:jersey-client:jar:2.4.1
- org.glassfish.jersey.core:jersey-common:jar:2.4.1
- org.glassfish.jersey.media:jersey-media-multipart:jar:2.4.1
- org.glassfish.jersey.core:jersey-server:jar:2.4.1
- org.hamcrest:hamcrest-core:jar:1.3
- org.jvnet.mimepull:mimepull:jar:1.9.3
- stax:stax:jar:1.2.0
- stax:stax-api:jar:1.0.1

NOTA: En el caso de utilizar un servidor Tomcat 6 o superior, no se deberá incluir la dependencia **“javax.servlet:javax.servlet-api:jar:3.1.0”**.

B. Establecer dependencias con maven: dirigido a aplicaciones mavenizadas.

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>D.G. de Política Digital</p>	<p>COMPULSA</p> <p>Manual de integración con repositorios para las Consejerías y entidades</p>
---	--	--

El componente “enidocWS” puede ser incluido en aplicaciones desarrolladas en maven añadiendo la siguiente dependencia al fichero **pom.xml** de la aplicación:

```
<dependency>
  <groupId>es.juntadeandalucia.nti</groupId>
  <artifactId>enidocWS</artifactId>
  <version>x.x</version>
</dependency>
```

Este componente puede encontrarse en el Repositorio Maven del Servicio de Coordinación de Administración Electrónica (<https://ws024.juntadeandalucia.es/ae/adminelec/repositoriomaven>).

Para que Maven obtenga el artefacto que contiene el componente y resuelva automáticamente sus dependencias, es necesario incluir del mismo modo, la referencia al repositorio de software donde se encuentra el componente:

```
<repository>
  <id>RepositorioMavenSCAE</id>
  <name>Repositorio de la Junta de Andalucía</name>
  <url>https://ws024.juntadeandalucia.es/maven/repository/internal</url>
</repository>
```

NOTA: El Repositorio Maven del Servicio de Coordinación de Administración Electrónica de la Consejería de Hacienda y Administración Pública está accesible a través del protocolo HTTPS utilizando el certificado **cn=*.juntadeandalucia.es, ou=FNMT Clase 2 CA, o=FNMT, c=ES** de la Junta de Andalucía. Para que Maven confíe en el dominio ws024.juntadeandalucia.es a través de HTTPS y logre descargar el componente y sus dependencias, es necesario que incluya el certificado indicado en el almacén de certificados por defecto (cacerts) de la Máquina Virtual Java sobre la que se ejecute el script **mvn**.

2.1.2 Fichero “web.xml”

Es requisito indispensable para la llevar a cabo la integración que se defina el siguiente servlet, como cualquier otro integrante de la aplicación web cliente, en el fichero “**web.xml**”. Se muestra a continuación un ejemplo del fichero con la definición del servlet mencionado:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
  <display-name>Web Service Enidoc</display-name>
  <servlet>
    <servlet-name>EnidocService</servlet-name>
    <servlet-class>org.glassfish.jersey.servlet.ServletContainer</servlet-class>
    <init-param>
      <param-name>jersey.config.server.provider.classnames</param-name>
      <param-value>org.glassfish.jersey.media.multipart.MultiPartFeature;
es.juntadeandalucia.nti.ws.eni.services.EnidocWS</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>
  <servlet-mapping>
    <servlet-name>EnidocService</servlet-name>
    <url-pattern>/rest/eni/*</url-pattern>
  </servlet-mapping>
</web-app>
```

2.1.3 Parámetros de configuración

Para la correcta configuración del componente, se presentan dos modalidades:

- Mediante un fichero de propiedades.
- Mediante parámetros de contexto definidos en el fichero “web.xml”.

2.1.3.1 Fichero de propiedades

Con el objetivo de facilitar la integración se establece un fichero de propiedades, denominado “**enidocWS.properties**” que contiene los argumentos de las peticiones a realizar.

A continuación se muestra un ejemplo del fichero que debe incluirse en el classpath donde se ejecute el servicio, donde se observan los parámetros a cumplimentar:

```
# Información de correspondencia entre la herramienta y el repositorio/sistema integrado
# usuario registrado y autorizado
usuariows = usuario
# password registrado y autorizado
passwordws = pass
# clase (ruta del paquete separado por puntos '.') donde se implementa el método de recuperación
clasesws = paquete.clase.MiClase
# nombre del método que recibe como parámetro el objeto Petición y devuelve Enidoc
metodows = nombreMetodo
# clase (ruta del paquete separado por puntos '.') donde se implementa el método de inserción
claseswsinsertar = paquete.clase.MiClaseInsertar
# nombre del método que recibe como parámetro el objeto InputStream y devuelve DatosEni
metodowsinsertar = nombreMetodoInsertar
# Valor true/false para obviar o no la limitación de 30 caracteres del CSV
obviarCaracteresCSV = valor
```

2.1.3.2 Parámetros de contexto

En caso de optar por no realizar la configuración directamente mediante el fichero de propiedades, se ofrece la posibilidad de realizar la configuración necesaria a través de parámetros de contexto. Para ello, será necesario especificar dos parámetros de contexto en el fichero “**web.xml**” del sistema que identifiquen, la clase y el método que instanciarán las propiedades necesarias para el funcionamiento del servicio web. Los parámetros deben ser:

1. **claseConfiguracionEnidocWS:** cuyo valor se corresponde con la ruta de la clase donde se instanciará el método de recuperación de las propiedades.
2. **metodoObtencionPropiedadesEnidocWS:** cuyo valor se corresponde con el nombre del propio método encargado de obtener las propiedades.

```
<context-param>
  <description>Indica la clase que contiene el método de obtención de las propiedades necesarias
para EnidocWS</description>
  <param-name>claseConfiguracionEnidocWS</param-name>
  <param-value>paquete.clase.MiClaseConfiguracion</param-value>
</context-param>

<context-param>
  <description>Indica el método que obtiene las propiedades necesarias para EnidocWS</description>
  <param-name>metodoObtencionPropiedadesEnidocWS</param-name>
  <param-value>nombreMetodoObtenerParametros</param-value>
</context-param>
```

La clase deberá llamarse como se indica en la etiqueta param-value definida anteriormente, y estar ubicada en el paquete indicado. Es estrictamente necesario que el método de obtención de las propiedades que se defina y se haya establecido en el parámetro de contexto sea **estático**. Dicho método será el encargado de dar valor a un atributo de tipo **java.util.Properties**, cumplimentando las propiedades indicadas en el apartado anterior.

Se muestra un ejemplo básico de construcción de la clase y el método necesarios:

```
package paquete.clase;

import java.util.Properties;

public class MiClaseConfiguracion{

    public static Properties properties = null;

    public static Properties nombreMetodoObtenerParametros() {
        properties = new Properties();
        // usuario registrado y autorizado
        properties.setProperty("usuarioWS", "usuario");
        // password registrado y autorizado
        properties.setProperty("passwordWS", "pass");
        // clase (ruta del paquete separado por puntos '.') donde se implementa el método de
        // recuperación
        properties.setProperty("claseWS", "paquete.clase.MiClase");
        // nombre del método que recibe como parámetro el objeto Peticion y devuelve
        // Enidoc
        properties.setProperty("metodoWS", "nombreMetodo");
        // clase (ruta del paquete separado por puntos '.') donde se implementa el método de
        // inserción
        properties.setProperty("claseWSInsertar", "paquete.clase.MiClaseInsertar");
        // nombre del método que recibe como parámetro el InputStream y devuelve
        // DatosEni
        properties.setProperty("metodoWSInsertar", "nombreMetodoInsertar");
        // Para obviar la limitación de 30 caracteres del CSV
        properties.setProperty("obviarCaracteresCSV", "valor");
        return properties;
    }
}
```

NOTA: Se permite la configuración múltiple de usuarios de acceso al Servicio Web. Esto es, múltiples usuarioWS-passwordWS, con el fin de que un sistema/repositorio que despliegue el componente enidocWS pueda dar servicio a varias herramientas.

En ambos modos de configuración, es suficiente con separar por comas ',' los valores en los campos usuarioWS y passwordWS. Deben indicarse los mismos números de valores en estos campos y en el orden adecuado, es imprescindible que no haya usuarios repetidos:

Por ejemplo:

```
usuarioWS = usuarioA,usuarioB
passwordWS = passA,passB
```

Se corresponderá con el sistemaA (usuarioA, passA) y con el sistemaB (usuarioB, passB).

NOTA2: En caso de que su herramienta sea capaz de devolver Documentos eni a raíz de códigos de verificación, para el correcto funcionamiento deberá indicar el valor *true* para el parámetro *obviarCaracteresCSV*, en otros casos, dicho parámetro puede estar a *false* o ni siquiera ser indicado.

2.1.4 Implementación de la clase invocada por el WS

Es necesario que esta clase contenga un método, que necesariamente debe ser **estático**, con la siguiente especificación, según el caso:

```
A) public static EniDoc nombreMetodo (Petición petición)
```

```
B) public static DatosEni nombreMetodoIn (InputStream in)
```

2.1.4.1 A) Obtención del documento ENI

Sobre este método recae la responsabilidad de obtener el documento ENI del sistema/repositorio cliente para devolver el objeto esperado.

A continuación se muestra un ejemplo de implementación de este método con el objetivo de guiar a los desarrolladores en la secuencia de pasos a seguir para obtener el documento ENI.

```
public static EniDoc nombreMetodo (Petición petición) {  
  
    // Creamos el objeto EniDoc que debemos rellenar  
    EniDoc eniDoc = new EniDoc();  
  
    // TODO Tratamiento opcional de los datos de la Petición para definir e implantar  
    // criterios y políticas para la gestión de las peticiones de acceso al documento  
    String idConsultado = petición.getIdConsultado();  
    String ip = petición.getIp();  
    String perfil = petición.getPerfil();  
    String motivo = petición.getMotivo();  
    CertificadoEP certEP = petición.getCertEP();  
    CertificadoPF certPF = petición.getCertPF();  
    CertificadoPJ certPJ = petición.getCertPJ();  
  
    // TODO Obtener documento de BBDD mediante el csv consultado  
    byte[] documentoBytes = obtenerDocumentoENIPorId(csv);  
  
    // Si el documento vinculado al CSV no existe en el repositorio  
    if (documentoBytes == null) {  
        eniDoc = new EniDoc();  
        eniDoc.setCodigoRespuesta(ErroresEni.STATUS_NOT_FOUND_RESOURCE);  
    }  
    // Si el documento vinculado al CSV existe en el repositorio  
    else {  
  
        // TODO Rellenar el objeto EniDoc  
        // 1.- Array de bytes que contiene el documento electrónico ENI.  
        // Obligatorio.  
        eniDoc.setDocumentoENI(documentoBytes);  
  
        // 2.- Array de bytes que contiene el documento original con el cajetín  
        // de firma incorporado. Opcional.  
        byte[] informeFirma = null;  
        eniDoc.setInformeFirma(informeFirma);  
  
        // 3.- Texto libre de respuesta para ser mostrado al usuario final. Si  
        // este campo se complementa se dispondrá al usuario única y  
        // exclusivamente dicho texto, ignorando el resto de la información.  
        // Opcional.  
        String textoLibre = null;  
        eniDoc.setTextoLibre(textoLibre);  
    }  
}
```

```
// 4.- Texto libre adicional de respuesta para ser mostrado al usuario
// final, junto con los demás resultados del proceso. Opcional.
String textoAdicional = null;
eniDoc.setTextoAdicional(textoAdicional);

// 5.- En caso de firma electrónica basada en certificado, componente o
// herramienta utilizada para su generación. Opcional.
String generadorFirma = null;
eniDoc.setGeneradorFirma(generadorFirma);

// 6.- Información de utilidad para las firmas electrónicas que no
// incorporen sello de tiempo. Opcional.
Date marcaTiempo = null;
eniDoc.setMarcaTiempo(marcaTiempo);

// 7.- Campo que recoge, en el caso de firma electrónica basada en
// certificado, si el sello de tiempo es requerido. Este dato resultará
// de utilidad en el caso de que la firma
// electrónica no disponga de sello de tiempo, o el mismo hubiera
// caducado, en el proceso de su validación. Opcional.
boolean selloTiempoRequerido = true;
eniDoc.setSelloTiempoRequerido(selloTiempoRequerido);

// 8.- Campo que recoge expresamente, en el caso de firma electrónica
// basada en certificado, si en su momento se validaron los certificados
// electrónicos utilizados para la firma. Tomará valores 'true' o
// 'false'. Este dato resultará de utilidad en el caso de que la firma
// electrónica no disponga de sello de tiempo, en el proceso de su validación.
// Opcional.
boolean validacionCertificadosFirma = true;
eniDoc.setValidacionCertificadosFirma(validacionCertificadosFirma);

// 9.- Campo con el identificador normalizado del documento que, en su
// caso, hubiera sustituido al documento al cual se intenta acceder.
// Opcional.
String identificadorDocumentoSustituto = null;
eniDoc.setIdentificadorDocumentoSustituto(identificadorDocumentoSustituto);

// 10.- Campo que indica si la firma electrónica está almacenada en un
// depósito seguro. Opcional
boolean depositoSeguro = false;
eniDoc.setDepositoSeguro(depositoSeguro);

// 11.- Campo que indica si se desea habilitar que la persona usuaria de
// la herramienta pueda descargar el correspondiente fichero de firma
// electrónica y el fichero en formato ENI. Opcional
boolean descargarFirmaydocumentoENI = true;
eniDoc.setDescargarFirmaydocumentoENI(descargarFirmaydocumentoENI);

// 12.- Se considera este campo, en el caso de que el tipo de firma no
// esté contemplado en la adaptación al NTI. Opcional
// - TF97 - CMS.
// - TF98 - XMLDSig.
// - TF99 - Indefinido.
TipoFirmaNoNormalizada tipoFirmaNoNormalizada = null;
eniDoc.setTipoFirmaNoNormalizada(tipoFirmaNoNormalizada);

// 13.- Campo adicional que se utilizará para posibles necesidades
// futuras, en el caso que se requiera alguna información adicional.
// Opcional.
String campoAdicional = null;
eniDoc.setCampoAdicional(campoAdicional);
}

// Devolver el objeto EniDoc relleno.
return eniDoc;
}
```

2.1.4.2 B) Incorporación del Documento ENI

Sobre este método recae la responsabilidad de obtener el csv y/o adicionalmente el identificador normalizado del documento a almacenar en el sistema/repositorio cliente.

A continuación se muestra un ejemplo de implementación de este método con el objetivo de guiar a los desarrolladores en la secuencia de pasos a seguir para obtener el objeto DatosEni.

```
public static DatosEni nombreMetodo (InputStream in) {
    DatosEni datosEni = null;
    try {
        // Creamos el objeto DatosEni que debemos rellenar
        datosEni = new DatosEni();
        // SISTEMA, se corresponde con los 5 caracteres que identifican el sistema
        // repositorio
        String csv = UtilidadesEni.generarCSV("SISTEMA",
            RandomStringUtils.randomNumeric(6)).getValor();
        // Indicar el identificador
        String identificador = "IDENTIFICADOR";
        // Se crea un DocumentoEni del inputStream recibido
        DocumentoEni conversorENI = new DocumentoEni(in);
        // Crear un DocumentoEni temporal para setear los valores
        DocumentoEni docEniAux = new DocumentoEni();
        docEniAux.setContenido(conversorENI.getContenido());
        docEniAux.setFirmas(conversorENI.getFirmas());
        docEniAux.setOrganos(conversorENI.getOrganos());
        docEniAux.setFechaCaptura(conversorENI.getFechaCaptura());
        docEniAux.setNombreFormato(conversorENI.getNombreFormato());
        docEniAux.setTipoDocumental(conversorENI.getTipoDocumental());
        docEniAux.setEstadoElaboracion(conversorENI.getEstadoElaboracion());
        docEniAux.establecerIdentificador(identificador);
        ByteArrayOutputStream out = new ByteArrayOutputStream();
        docEniAux.storeToXML(out);
        byte[] bytesEni = out.toByteArray();
        // Método encargado grabar el documento eni
        escribirArchivo(bytesEni, identificador);
        datosEni.setCsv(csv);
        String idEspecifico = UtilidadesEni.generaIdentificadorEni(
            ORGANO, identificador, new Date());
        datosEni.setIdentificadorDocumento(idEspecifico);
    } catch (ConversionException e) {
        datosEni = null;
    }
    return datosEni;
}
```

2.2 Despliegue de la aplicación “enidocWS”

En caso de optar por esta alternativa, se pone a disposición una aplicación intermedia entre la herramienta destino y el sistema/repositorio donde se deberá configurar el fichero de propiedades e implementar la clase contenedora del método/s a invocar desde el servicio web, de forma análoga a la explicada respectivamente en los puntos 2.1.3 y 2.1.4 del presente documento.

Para acceder al modelo de datos del sistema/repositorio, es posible que sea necesario hacer uso de jdbc. Para ello habría que definir el DataSource correspondiente en el fichero “context.xml”.

Una vez configurada la aplicación e implementado el código necesario, se debe generar el war de la aplicación y desplegarlo en un servidor **Tomcat** con **JDK 1.6**.

3 ANEXO I. COMPONENTE ENIDOCWS

Este componente, “enidocWS”, pone a disposición de los usuarios una librería de servicios web (archivo jar), para obtener el documento ENI, o almacenarlo, proporcionando la información requerida por la herramienta destino en cada caso. También se distribuye en formato de aplicación web (archivo war) para que sea desplegada en un contenedor web JEE, según las necesidades de cada sistema/repositorio.

Estos servicios web han sido desarrollados bajo REST (Representational State Transfer, técnica de arquitectura software para sistemas hipermedia distribuidos como la World Wide Web), haciendo uso del framework Jersey para implementación de Servicios Web RESTful en Java.

3.1 WS obtener documento ENI

3.1.1 Flujo del Servicio de Integración

Se muestra el diagrama UML que representa el flujo de integración y la relación entre las distintas clases que conforman el servicio web:



Ilustración 2. Diagrama de clases EnidocWS (verificarFirma)

La implementación que deberán realizar los sistemas/repositorios que se integren, consistirá, principalmente, en la implementación de un método que recibe como parámetro de entrada un objeto de tipo **Peticion** y devuelve como respuesta un objeto de tipo **Enidoc**, cuyo contenido será un documento ENI acompañado de un conjunto de metadatos adicionales.

3.1.2 Estructura de la Petición

A continuación se muestra el diagrama UML con la relación entre las entidades que conforman la Petición al servicio de integración:

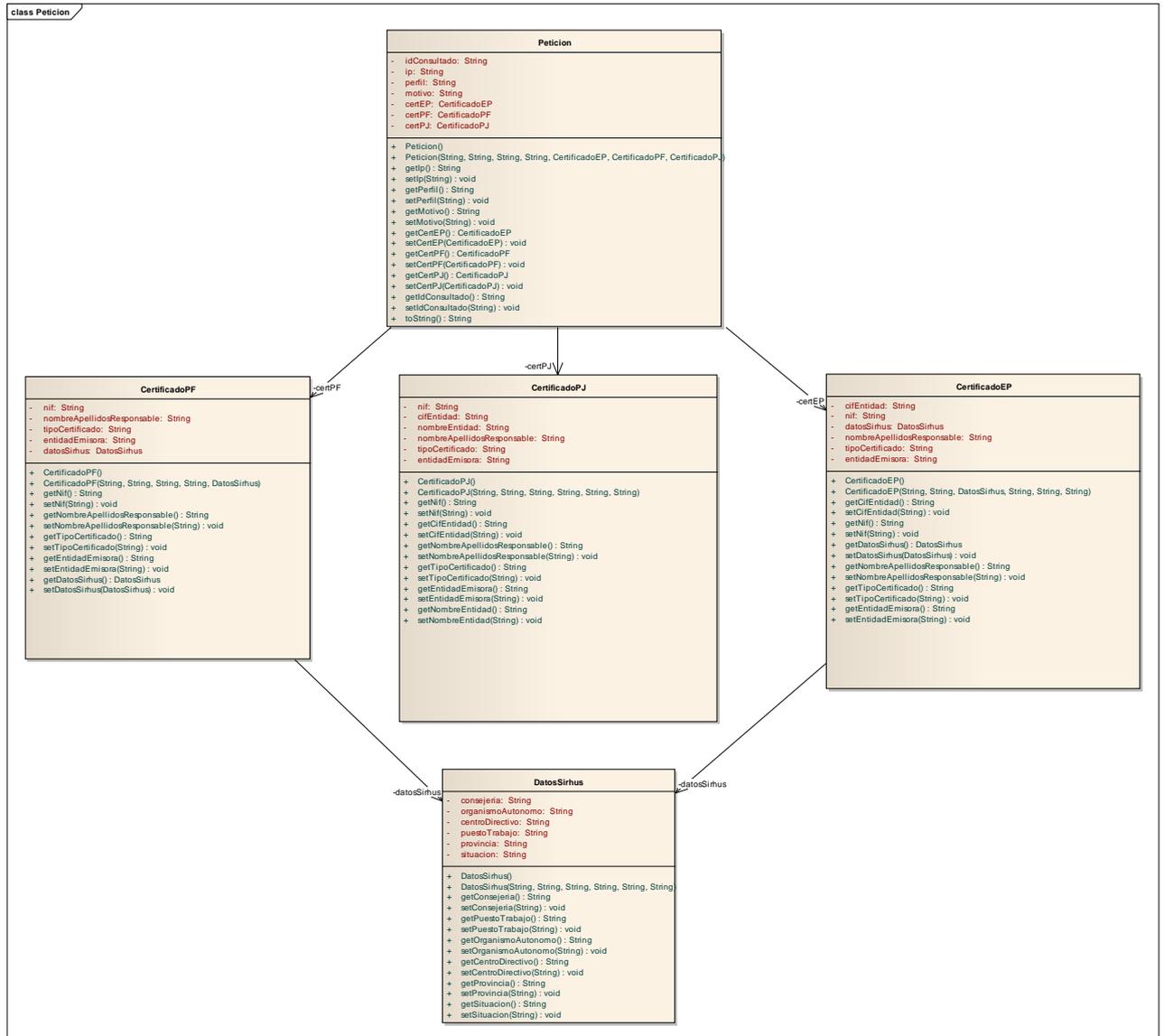


Ilustración 3. Diagrama de clases Petición

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>D.G. de Política Digital</p>	<p style="text-align: right;">COMPULSA</p> <p style="text-align: right;">Manual de integración con repositorios para las Consejerías y entidades</p>
---	--	--

El detalle de los campos que componen el objeto **Peticion** cuyos datos serán proporcionados por la herramienta destino que para invocar al componente “EnidocWS” correspondiente al servicio web del sistema/repositorio que custodia el **documento ENI** requerido, es el siguiente:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
peticion	Objeto que representa la petición realizada	Peticion	Obligatorio

El tipo de objeto **Peticion** se compone de los siguientes objetos Java:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
idConsultado	Identificador de la petición. Será el código seguro de verificación (CSV) facilitado por el usuario.	String	Obligatorio
ip	Identificador del host que realiza la petición.	String	Obligatorio
perfil	Cadena con uno de los siguientes valores <ul style="list-style-type: none"> • 0: Ciudadanía • 1: Personal empleado público de la Junta de Andalucía en el ejercicio de sus funciones • 2: Otro personal empleado público en el ejercicio de sus funciones 	String	Obligatorio
motivo	Motivo por el cual el usuario desea acceder al documento.	String	Opcional
certEP	Certificado de Empleado Público	CertificadoEP	Opcional
certPF	Certificado de Persona Física	CertificadoPF	Opcional
certPJ	Certificado de Persona Jurídica	CertificadoPJ	Opcional

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>D.G. de Política Digital</p>	<p style="text-align: right;">COMPULSA</p> <p style="text-align: right;">Manual de integración con repositorios para las Consejerías y entidades</p>
---	--	--

El campo “**idConsultado**” aceptado por la herramienta destino donde **es indispensable que los 5 primeros caracteres sean el identificador único del sistema/repositorio proporcionado por la propia herramienta**. Para el resto de caracteres, se recomienda usar el código seguro de verificación propuesto en el Anexo II del presente documento. No obstante, cada administración puede diseñar el proceso de generación según sus necesidades, asegurando en cualquier caso los principios de impredecibilidad, uniformidad, resistencia a colisiones e irreversibilidad.

Detalle del objeto **CertificadoEP**:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
cifEntidad	CIF de la entidad	String	Obligatorio
nif	NIF del empleado	String	Obligatorio
nombreApellidosResponsable	Nombre y apellidos	String	Obligatorio
tipoCertificado	Tipo de certificado	String	Obligatorio
entidadEmisora	Nombre de la entidad emisora del certificado	String	Obligatorio
datosSirhus	Datos devueltos por el sistema SIRhUS	DatosSirhus	Opcional

Detalle del objeto **CertificadoPF**:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
nif	NIF de la persona	String	Obligatorio
nombreApellidosResponsable	Nombre y apellidos	String	Obligatorio
tipoCertificado	Tipo de certificado	String	Obligatorio
entidadEmisora	Nombre de la entidad emisora del certificado	String	Obligatorio
datosSirhus	Datos devueltos por el sistema SIRhUS	DatosSirhus	Opcional

	Consejería de Hacienda y Administración Pública D.G. de Política Digital	COMPULSA Manual de integración con repositorios para las Consejerías y entidades
---	--	--

Detalle del objeto **CertificadoPJ**:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
nif	NIF de la persona	String	Obligatorio
cifEntidad	CIF de la entidad representada	String	Obligatorio
nombreEntidad	Nombre de la entidad representada	String	Obligatorio
nombreApellidosResponsable	Nombre y apellidos	String	Obligatorio
tipoCertificado	Tipo de certificado	String	Obligatorio
entidadEmisora	Nombre de la entidad emisora del certificado	String	Obligatorio

Detalle del objeto **DatosSirhus**:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
consejeria	Nombre de la Consejería	String	Obligatorio
organismoAutonomo	Entidad a la que pertenece	String	Obligatorio
centroDirectivo	Centro directivo	String	Obligatorio
puestoTrabajo	Puesto de trabajo	String	Obligatorio
provincia	Provincia	String	Obligatorio
situacion	Situación	String	Obligatorio

La obtención de datos del sistema SIRHUS a partir del NIF, extraído del certificado usado para el acceso a la herramienta, será "tolerante a fallos" en el sentido de que si no se obtiene respuesta en un tiempo prudencial, o bien se obtiene una respuesta errónea, se completará el objeto con los correspondientes campos vacíos.

Se incluye una cabecera con las credenciales de autorización (usuario y clave) en la petición del servicio de integración. Estas credenciales serán las proporcionadas por el sistema/repositorio tercero para su inclusión en la herramienta que consuma el servicio.

3.1.3 Estructura de la Respuesta

A continuación se muestra el diagrama que representa la clase de Respuesta a devolver por los sistemas/repositorios:



Ilustración 4. Diagrama de clases Respuesta EniDoc

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>D.G. de Política Digital</p>	<p>COMPULSA</p> <p>Manual de integración con repositorios para las Consejerías y entidades</p>
---	--	--

El detalle de los campos del objeto de repuesta **EniDoc** que deben devolver los sistemas/repositorios tras la petición de consulta por parte de la herramienta destino, es el siguiente:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
documentoENI	<p>Documento ENI, cuyo contenido se corresponde con lo definido en la NTI de Documento Electrónico¹.</p> <p>En aquellos casos en los que el repositorio/sistema que custodia el documento no esté en disposición de asignar identificadores normalizados de documentos según lo dispuesto en la Norma Técnica de Interoperabilidad de Documento Electrónico, como por ejemplo en el ámbito de las entidades no incorporadas al Directorio Común de Unidades Orgánicas, el metadato "Órgano" deberá adoptar el valor "XXXXXXXX".</p>	Fichero [cadena de bytes cuyo contenido se corresponde con lo definido en la Norma Técnica de Interoperabilidad del Documento Electrónico]	Obligatorio
informeFirma	Documento PDF de justificante de firma, en el que se incluye el pie de firma. En el caso de que se devuelva este fichero, se mostrará al usuario final la posibilidad de acceder al mismo.	Fichero [cadena de bytes]	Opcional
textoLibre	Texto libre de respuesta para ser mostrado al usuario final. Si este campo se complementa por el sistema/repositorio se mostrará al usuario única y exclusivamente dicho texto, ignorando el resto de la información.	String	Opcional

¹ Si se asigna un valor al metadato mínimo obligatorio **Identificador** del Documento Electrónico, deberá cumplir con los requisitos definidos por la Norma Técnica de Interoperabilidad del Documento Electrónico (<Idioma>_<Órgano>_<AAAA>_<ID_especifico>), donde:

1. <Idioma>: código del idioma o lengua cooficial del documento: - ES: español
2. <Órgano>: Código alfanumérico único para cada órgano/unidad/oficina extraído del Directorio Común gestionado por el Ministerio de Hacienda y Administraciones Públicas (Longitud: 9 caracteres).
3. <AAAA>: Año de la fecha de captura del documento. (Longitud: 4 caracteres).
4. <ID_especifico>: Código alfanumérico que identifica de forma única al documento dentro de los generados por la administración responsable. (Longitud: 30 caracteres).

Es indispensable que los 5 primeros caracteres sean el identificador único del sistema/repositorio proporcionado por la herramienta destino. Para el resto de caracteres, cada administración puede diseñar el proceso de generación según sus necesidades, asegurando en cualquier caso su unicidad y que **no coincidan con el valor del código seguro de verificación.**

 <p>JUNTA DE ANDALUCÍA CONSEJERÍA DE HACIENDA Y ADMINISTRACIÓN PÚBLICA</p>	<p>Consejería de Hacienda y Administración Pública</p> <p>D.G. de Política Digital</p>	<p>COMPULSA</p> <p>Manual de integración con repositorios para las Consejerías y entidades</p>
---	--	--

textoAdicional	Texto libre adicional de respuesta para ser mostrado al usuario final, junto con los demás resultados del proceso.	String	Opcional
generadorFirma	En caso de firma electrónica basada en certificado, componente o herramienta utilizada para su generación.	String	Opcional
marcaTiempo	Información de utilidad para las firmas electrónicas que no incorporen sello de tiempo. Marca de tiempo de generación / incorporación del documento al sistema/repositorio y conforme al artículo 15 del Real Decreto 4/2010, de 8 de enero, por el que se regula el Esquema Nacional de Interoperabilidad en el ámbito de la Administración Electrónica.	Date	Opcional
validacionCertificadosFirma	Validación de certificados en el momento de realizar la firma basada en certificado electrónico.	boolean	Opcional
selloTiempoRequerido	Indica si el sello de tiempo es o no requerido.	boolean	Opcional
identificadorDocumentoSustituto	Identificador normalizado que, en su caso, hubiera sustituido al documento al cual se intenta acceder.	String	Opcional
depositoSeguro	Indica si la firma electrónica está almacenada en un depósito seguro. ²	boolean	Opcional
descargarFirmaydocumentoENI	Indica si se desea habilitar que la persona usuaria de la herramienta pueda descargar el correspondiente fichero de firma electrónica y el documento ENI. ³	boolean	Opcional

² Según epígrafes 2.b y 5.b del apartado II.7 de la Norma Técnica de Interoperabilidad de Política de Firma Electrónica y de certificados de la Administración.

³ Ello, siempre que el sistema/repositorio se haya configurado previamente para habilitar esta posibilidad.

	<p>Consejería de Hacienda y Administración Pública</p> <p>D.G. de Política Digital</p>	<p style="text-align: right;">COMPULSA</p> <p style="text-align: center;">Manual de integración con repositorios para las Consejerías y entidades</p>
---	--	---

tipoFirmaNoNormalizada	Se considera este campo, en el caso de que el tipo de firma no esté contemplado en la adaptación al NTI ⁴	TipoFirmaNoNormalizada [Tipificado: - TF97 - TF98 - TF99]	Opcional
campoAdicional	Campo adicional que se utilizará para posibles necesidades futuras.	String	Opcional

3.1.4 Gestión de respuestas y errores

Los mensajes de respuesta y de error se producirán en base a las siguientes situaciones:

- La herramienta de verificación conoce el sistema/repositorio que custodia el documento, se comunica con él y este responde señalando que no conoce el código de verificación solicitado, cumplimentando el parámetro "documentoENI" a nulo o vacío. Se indicará al usuario que el documento no se ha encontrado.
- La herramienta de verificación no conoce el sistema/repositorio que custodia el documento. Se indicará al usuario que el documento no se ha encontrado.
- La herramienta de verificación trata de comunicarse con el sistema/repositorio que custodia el documento y no es posible o este devuelve un error. Se indicará al usuario que en ese momento debido a problemas técnicos no es posible recuperar el documento.
- La herramienta de verificación se comunica con el sistema/repositorio correctamente y este responde indicando el identificador normalizado del documento que sustituye al documento al cual se intenta acceder, cumplimentando el parámetro "identificadorDocumentoSustituto". Se indicará al usuario únicamente el identificador normalizado del documento sustituto, ignorando el resto de la información.
- La herramienta de verificación se comunica con el sistema/repositorio correctamente y este, en caso de que concluya que no procede aportar el documento a la persona usuaria de la herramienta, cumplimenta el parámetro "textoLibre" expresando un texto informativo que se le mostrará al usuario, ignorando el resto de la información.

⁴ Identificar que es necesario incluir nuevos valores de forma que se puedan informar formatos de firma no incluidos en la Norma Técnica de Interoperabilidad de Política de Firma Electrónica y de certificados de la Administración, sin perjuicio de que no se deba continuar con la utilización de los mismos. Se han incluido los tipos de firmas:

- **TF97**: para formatos de firma CMS
- **TF98**: para formatos de firma XMLDSig
- **TF99**: para otros formatos de firma (por ejemplo PKCS#7).

En el caso de que este campo venga informado, la herramienta de verificación informará a los usuarios el tipo de firma indicada por el valor de este campo. Es decir, el valor de este campo prevalece sobre el valor del campo documentoENI en el que se indica el tipo de firma, el cuál es obligatorio. Se establece la siguiente tabla de mapeos:

Tipo de Firma	Valor en el campo documentoENI	Valor campo otroTipoFirmaNoNormalizada
CMS	TF02	TF97
XMLDSig	TF03	TF98
Indefinido	TF02	TF99

3.2 WS almacenar documento ENI

3.2.1 Flujo del Servicio de Integración

Se muestra el diagrama UML que representa el flujo de integración y la relación entre las distintas clases que conforman el servicio web:

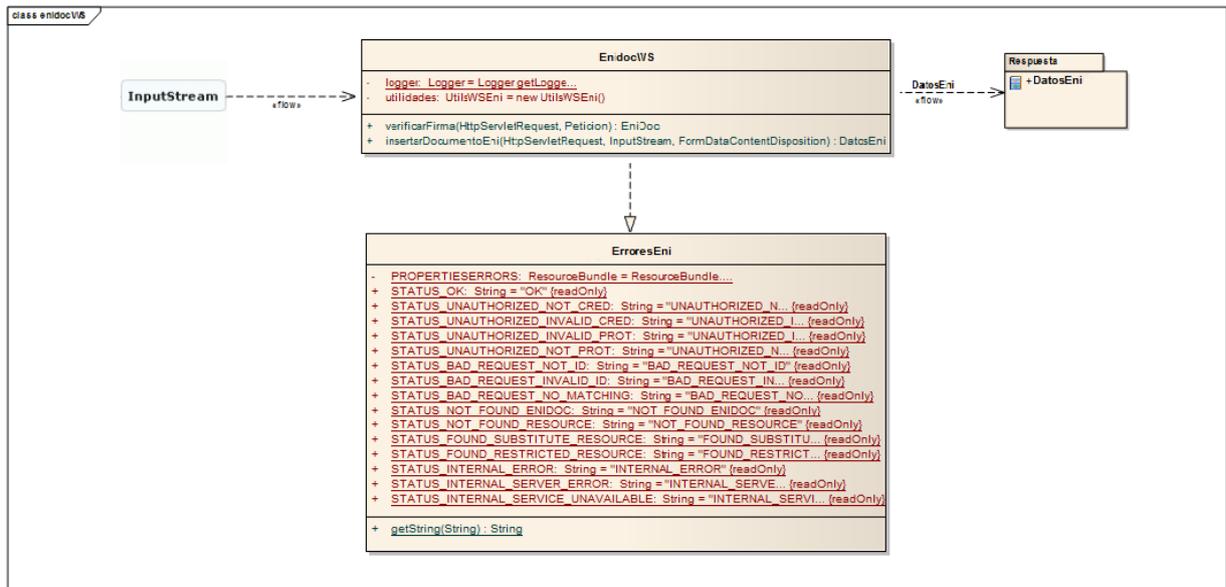


Ilustración 5. Diagrama de clases EndocWS (insertarDocumentoEni)

La implementación que deberán realizar los sistemas/repositorios que se integren, consistirá, principalmente, en la implementación de un método que recibe como parámetro de entrada un objeto de tipo **InputStream** y devuelve como respuesta un objeto de tipo **DatosEni**, cuyo contenido será un código de verificación acompañado de un identificador normalizado.

3.2.2 Estructura de la Petición

La petición estará compuesta por un **InputStream** cuyo contenido será un documento ENI con toda la información necesaria: contenido, firma y metadatos.

3.2.3 Estructura de la Respuesta

A continuación se muestra el diagrama que representa la clase de Respuesta a devolver por los sistemas/repositorios:

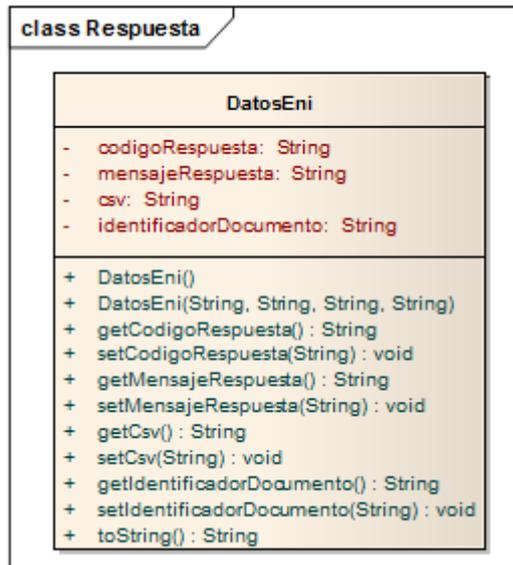


Ilustración 6. Diagrama de clases Respuesta DatosEni

El detalle de los campos del objeto de respuesta **DatosEni** que deben devolver los sistemas/repositorios tras la petición de consulta por parte de la herramienta destino, es el siguiente:

NOMBRE	DESCRIPCIÓN	TIPO DATO	CARÁCTER
csv	Código de verificación generado por la aplicación que realiza la custodia del documento, mediante el cual se obtendrá el documento ENI.	String	Obligatorio
identificadorDocumento	Identificador normalizado del documento almacenado, será necesario si se modifica el del documento original.	String	Opcional

3.2.4 Gestión de respuestas y errores

Los mensajes de respuesta y de error se producirán en base a las siguientes situaciones:

- La herramienta de verificación no conoce el sistema/repositorio que custodiará el documento. Se indicará al usuario que el documento no se ha almacenado.
- La herramienta de verificación trata de comunicarse con el sistema/repositorio que custodiará el documento y no es posible o este devuelve un error. Se indicará al usuario que en ese momento debido a problemas técnicos no es posible almacenar el documento.

	Consejería de Hacienda y Administración Pública D.G. de Política Digital	COMPULSA Manual de integración con repositorios para las Consejerías y entidades
--	--	--

4 ANEXO II. GENERACIÓN DE CÓDIGO SEGURO DE VERIFICACIÓN

Este anexo presenta, a modo de ejemplo, una propuesta para calcular los 25 caracteres últimos para formar un código seguro de verificación.

- a. En primer lugar, se generará una cadena de caracteres concatenando la dirección MAC del servidor, la fecha actual en milisegundos y un número aleatorio.
- b. Sobre esta cadena de caracteres resultante, se aplica un algoritmo SHA1 para generar un hash, el cual será truncado a 96 bits.
- c. Una vez obtenido este código, se codificará en base64 con el fin de obtener 19 caracteres alfanuméricos en los rangos de a-z, A-Z, 0-9, \$ y &.
- d. A estos 19 caracteres se le concatenarán 6 caracteres, siendo éstos de libre uso por la Consejería o entidad (para codificaciones propias, especificación de dominios funcionales, etc.)

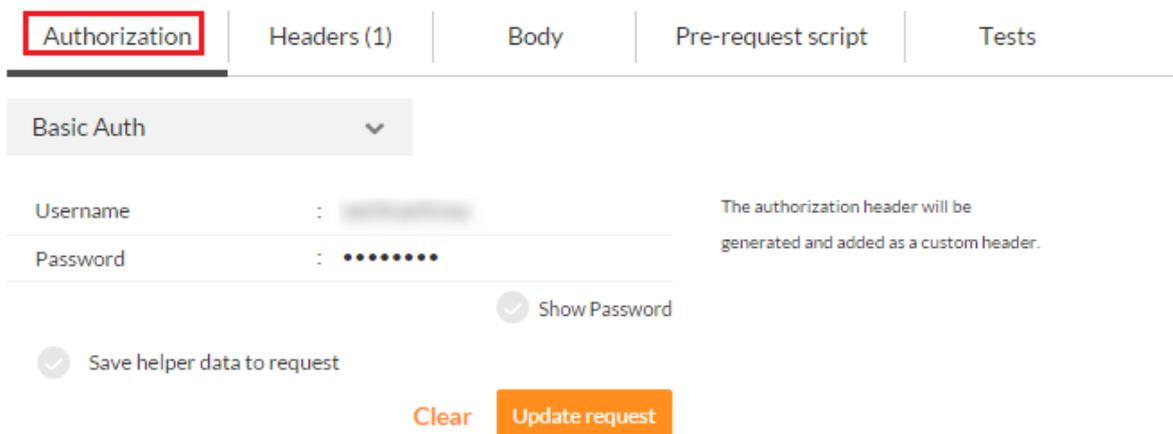
Cada Consejería, entidad, sistema/repositorio, puede utilizar otros medios para la generación de los códigos, atendiendo a sus prácticas, políticas y necesidades en materia de gestión documental.

5 ANEXO III. COMPROBACIÓN DE LA INTEGRACIÓN CON ENIDOCWS

5.1 Postman: Cliente REST para Chrome

Para verificar que se ha integrado correctamente la librería con la aplicación deseada y poder comprobar posibles errores o problemas se plantea la alternativa del uso de un cliente REST mediante la instalación de una aplicación complemento del navegador Chrome: **Postman**.

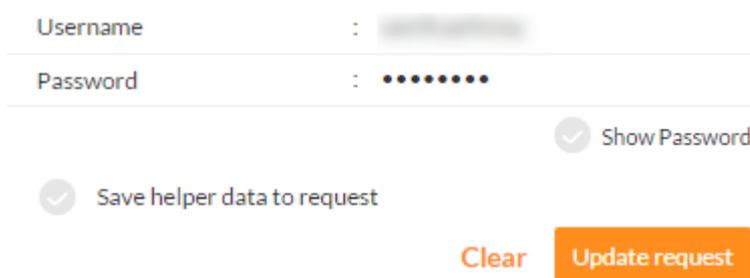
1. Instalar el complemento de Chrome:
<https://chrome.google.com/webstore/detail/postman/fhbjgbiflinjbdggehcddcbncdddomop>
2. Configurar los parámetros de autenticación en la opción "Authorization > Basic Authorization". Especificar el usuario y la clave de conexión al servicio ENIDOCWS facilitados.



The screenshot shows the 'Authorization' tab in Postman. The 'Basic Auth' dropdown is selected. The 'Username' field contains a blurred value, and the 'Password' field contains a blurred value. A 'Show Password' checkbox is checked. Below the fields are 'Clear' and 'Update request' buttons. A note on the right states: 'The authorization header will be generated and added as a custom header.'

Ilustración 7. Pantalla del complemento de Chrome

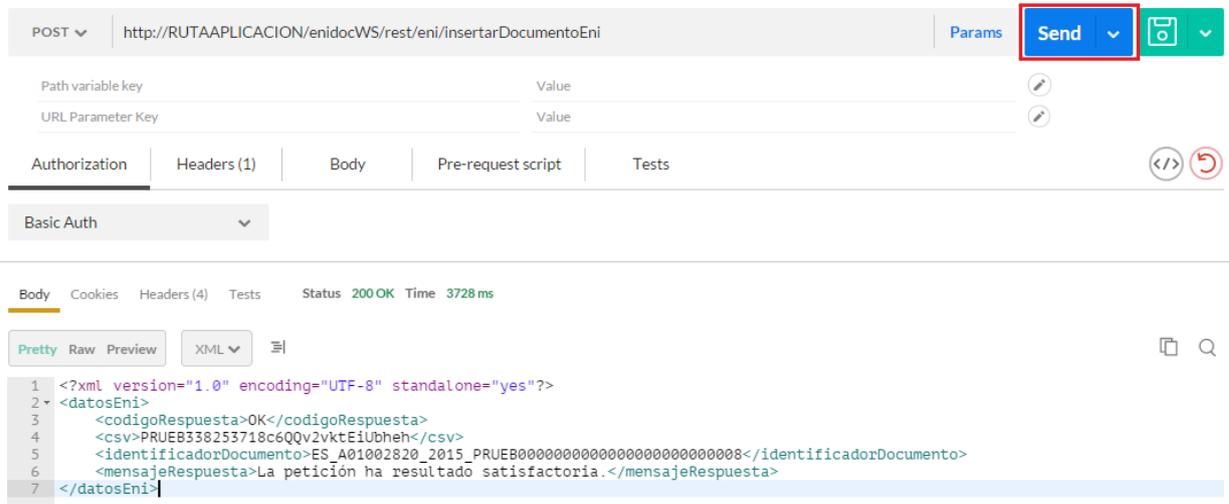
3. Especificar el usuario y la clave de conexión al servicio ENIDOCWS facilitados.



The screenshot shows the authentication form with 'Username' and 'Password' fields, both containing blurred text. A 'Show Password' checkbox is checked. Below the fields are 'Clear' and 'Update request' buttons.

Ilustración 8. Formulario de autenticación

b. *insertarDocumentoEni*.



The screenshot shows a REST client interface with the following details:

- Method:** POST
- URL:** http://RUTAAPLICACION/enidocWS/rest/eni/insertarDocumentoEni
- Authorization:** Basic Auth
- Status:** 200 OK
- Time:** 3728 ms
- Response Body (XML):**

```

1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <datosEni>
3   <codigoRespuesta>OK</codigoRespuesta>
4   <csv>PRUEB338253718c6QQv2vktEiUbheh</csv>
5   <identificadorDocumento>ES_A01002820_2015_PRUEB000000000000000000000008</identificadorDocumento>
6   <mensajeRespuesta>La petición ha resultado satisfactoria.</mensajeRespuesta>
7 </datosEni>

```

Ilustración 11. Respuesta correcta del servicio insertarDocumentoEni

- 2) Si por el contrario se produjera un error, se obtendrá información más detallada que permitirá identificarlo.

6 ANEXO IV. VALIDACIÓN DOCUMENTO ENI

Para validar un documento electrónico (ENI) se recomienda hacer uso de la funcionalidad de validación que dispone la Herramienta Centralizada de Verificación que pone a disposición la Junta de Andalucía. A continuación se indican las URLs de acceso a los entornos de pruebas y producción de esta herramienta:

Entorno de pruebas: https://cancanaprun1.chap.junta-andalucia.es/verificarFirma_des/

Entorno de producción: <https://ws050.juntadeandalucia.es/verificarFirma>

7 REFERENCIAS

Objeto	Referencia
NTI de Documento Electrónico	http://www.boe.es/diario_boe/txt.php?id=BOE-A-2011-13169
JVM	http://en.wikipedia.org/wiki/Java_virtual_machine
REST	http://en.wikipedia.org/wiki/REST
Jersey	https://jersey.java.net/
Complemento Firefox	https://addons.mozilla.org/es/firefox/addon/restclient/