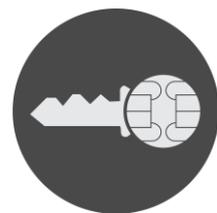


# U Signer



Servicios de implantación y Desarrollo de USignerRESTProxy

Propuesta para:

**BASE**  
GESTIÓ D'INGRESSOS

info@wsg127.com  
www.wsg127.com



<b>1</b>	<b>Introducción .....</b>	<b>3</b>
<b>2</b>	<b>Servicios de desarrollo de USigner RESTProxy .....</b>	<b>4</b>
2.1	Definición .....	4
2.2	Tareas a realizar y secuenciación de estas .....	6
2.3	Estimación y planificación.....	7
<b>3</b>	<b>Servicios de implantación .....</b>	<b>8</b>
3.1	Definición .....	8
3.2	Entornos .....	8
3.3	Protocolo de actuación.....	9
3.4	Ubicación de los diferentes tipos de actuación .....	10
3.5	Estimación y planificación.....	10
<b>4</b>	<b>Propuesta económica .....</b>	<b>11</b>
4.1	Servicios de implantación .....	11
4.2	Desarrollo de USignerRESTProxy.....	12
4.3	Notas .....	12
<b>5</b>	<b>Referencias de la oferta.....</b>	<b>13</b>

## 1 INTRODUCCIÓN

Base Gestió D'Ingressos dispone de la plataforma de Firma Electrónica ASF en su versión Advanced Edition, con la configuración de módulos siguiente:

- SignatureServer
- NonRepudiationService
- X509Validator
- PolicyManager
- EncryptionServer
- Consola de Administración
- WebSigner

La presente propuesta cubrirá incluirá los Servicios Profesionales necesarios para la implantación de la plataforma de Firma Electrónica **U Signer**, con la misma configuración de módulos que la actual plataforma implantada, a excepción de WebSigner que es sustituido por **U NetSigner**, componente de firma compatible con las integraciones existentes y que tiene como principal novedad la posibilidad de uso en navegadores que han dejado de soportar Applets.

Por otra parte, relacionado con esta implantación, surge la necesidad de construir una aplicación para facilitar el proceso de cambio e integración de las aplicaciones existentes, por ello la propuesta esta dividida a todos los niveles en 2 grandes partes, por un lado el desarrollo como tal del componente indicado y por otro la implantación en su misma de los nuevos productos.

## 2 SERVICIOS DE DESARROLLO DE USIGNER RESTPROXY

### 2.1 Definición

#### Contexto

Actualmente las integraciones de las diferentes aplicaciones contra la plataforma U Signer se hacen con un componente común de integración que actúa a modo de librería cliente, esto es, todas las aplicaciones tienen un punto común de integración que simplifica el proceso, este punto común limita las funcionalidades a utilizar permitiendo un control sobre que usa y que no cada una de estas aplicaciones, así mismo abstrae a estas aplicaciones de las dificultades de serializar/deserializar para la capa de servicios, pero todo esto es local, si hay que hacer un cambio este debe luego extenderse a cada una de estas aplicaciones, lo cual dificulta su evolución y mantenimiento.

#### Definición base de la evolución

Se pretende evolucionar la integración manteniendo la idea conceptual de un punto común de integración pero en el lado servidor.

U Signer tiene decenas de servicios a disposición con cientos de combinaciones de parámetros, y la tecnología de publicación de sus servicios web es SOAP, por lo que no nos sirve tal cual, se plantea la construcción de un proxy intermedio que hemos denominado “*USignerRESTProxy*”, una aplicación web con despliegue independiente a U Signer que publica una interfaz reducida de servicios web, de forma que controlamos aquí que pueden y que no pueden hacer las aplicaciones integradas, y estos servicios se publican en tecnología REST/JSON de forma que podemos prescindir de tener librerías cliente en cada una de las aplicaciones integradas.

La tecnología elegida para los servicios además de facilitar la integración nos facilitará futuras evoluciones de los servicios publicados, ya que permite añadir nuevos parámetros sin romper las integraciones existentes.

#### Funciones actuales del componente

Las actuales funciones del componente local servirán de base para establecer que servicios publicará el proxy, no obstante los hemos analizado y planteamos una adecuación y ampliación para dar cabida al menos a las evoluciones mas relevantes de U Signer, que resumiremos a alto nivel en lo que aquí aplica a “*dar soporte a las políticas europeas eIDAS*”.

Actuales funciones del componente local:

- `signar`
- `comprovarSignatura`
- `obtenerTimeStamp`
- `comprovarTimeStamp`

## Evolución de las actuales funciones

Hemos analizado el código de las funciones actualmente existentes, detectamos que la función `comprovarTimeStamp` no hace lo que su nombre hace presuponer, esta función no verifica un `TimeStamp`, no es la equivalente para verificar lo que ha hecho `obtenerTimeStamp`, realmente esta función lo que hace es validar y extraer información de un `X509Certificate`, por lo que proponemos cambio de nomenclatura para la misma.

En cuanto al número de funciones proponemos las mismas para estos nuevos servicios, donde planteamos la ampliación es en la parametrización de estas, así pues las nuevas funciones que proponemos con el cambio de nomenclatura serían:

- `signar`
- `comprovarSignatura`
- `obtenerTimeStamp`
- `comprovarCertificat`

Ampliación de parametrización en las funciones, se enumeran a alto nivel las nuevas funcionalidades que se proponen:

- Añadir posibles nuevos formatos de firma:
  - XAdES versiones 132 y 141 (el componente actual soporta XAdES pero no los esquemas actuales del formato)
  - PAdES (el componente actual no soporta el formato avanzado de la firma pdf)
- Añadir la posibilidad de uso de los Perfiles eIDAS para:
  - CAdES
  - XAdES
  - PAdES

A modo resumido diremos que eIDAS podemos considerarlo una política de firma, una forma concreta y específica de usar y generar cada uno de los formatos de firma avanzados, el soporte a esta política permite generar firmas que la cumplan y sean interoperables a nivel europeo sin necesidad de tener los conocimientos técnicos de que implican.

- Añadir la posibilidad de indicar subtipo de firma o nivel de eIDAS
- Añadir la posibilidad de indicar una política de firma
- Añadir la posibilidad de indicar la apariencia en las firmas PDF/PAdES

## Generación de paquete de test de USignerRESTProxy

Dentro del proyecto se incluye la generación de un paquete que contenga las clases necesarias para testear los servicios de USignerRESTProxy, este paquete tiene una doble función, por un lado servir de testeo y por otro servir a modo documental como código de ejemplo de integración contra los servicios.

La idea es generar un main que pueda ser ejecutado directamente desde el IDE, igualmente cualquier modificación/prueba de parámetros se deja recaer sobre el propio fuente.

Debe existir al menos un caso de prueba por cada uno de los servicios publicado y a ser posible varios ejemplos con diferentes juegos de parámetros para cada uno de dichos servicios.

## **Generación de un html de test de U NetSigner**

Este punto es independiente al resto, no tiene relación en si mismo con la construcción de USignerRESTProxy.

Este punto refleja la construcción de un html que haga de test contra el producto U NetSigner, algo similar al “*main*” que se ha definido en el punto anterior para USignerRESTProxy.

Desde el html debe poderse ejecutar un flujo completo de firma contra el producto cliente, esto es, debe permitir la instalación, debe permitir la extracción de certificados disponibles en local y debe permitir la firma electrónica con alguno de los certificados anteriores.

## **2.2 Tareas a realizar y secuenciación de estas**

Se enumeran a continuación las tareas de bajo nivel planificadas y su orden de ejecución para llevar a cabo el proyecto, no pretende ser un listado exacto y el único objeto es esbozar de manera informativa nuestro planteamiento del proyecto:

- Montar entorno de desarrollo:
  - VPN contra DPTa
  - SVN contra DPTa
  - Maven
  - Eclipse
  - Procesos de compilación y empaquetamiento
  
- Desarrollo:
  - Publicar servicios JSON, definición de los servicios la ya comentada, parametrización de los servicios la ya existente más las evoluciones planteadas
  - Comprobar y adaptar parámetros externos a la estructura de datos
  - Crear interfaz interna entre el JSON y el agente de seguridad
  - Crear implementación para conectar con los servicios web de U Signer
  - Adaptar la estructura de datos devuelta por U Signer a los resultados de JSON
  
- Generar paquete test U Signer:
  - main con integración a los servicios JSON:
    - `signar`
    - `comprovarSignatura`
    - `obtenerTimeStamp`
    - `comprovarCertificat`

- Generar paquete test U NetSigner:
  - html con integración (instalación + getcertificates + sign)
  
- Generar documentación y seguimiento:
  - Seguimiento por Redmine
  - Generación de Javadoc descriptivo de todos los servicios publicados
  - Generación de documentación en la Wiki de DPTa:
    - Descripción
    - Casos de Uso (por servicio con detalle de parámetros)
    - Diagrama de servicio (servicio + Estructura de datos)
    - Diagrama de despliegue

## 2.3 Estimación y planificación

Se estima la construcción de USignerRESTProxy en **216 horas** distribuidas de la siguiente forma:

Fase	Esfuerzo	Estimación ocupación según históricos				Esfuerzo por perfil		
		Tarea	JP	Analista	Programador	JP	Ana	Prog
<b>Gestión</b>	<b>11</b>	<b>100%</b>	<b>83%</b>	<b>17%</b>	<b>0%</b>	<b>9</b>	<b>2</b>	<b>0</b>
Arranque	3	30%	90%	10%		3	0	0
Seguimiento	5	45%	75%	25%		4	1	0
Cierre	2	15%	80%	20%		1	0	0
Calidad	1	10%	100%			1	0	0
<b>Análisis</b>	<b>43</b>	<b>100%</b>	<b>28%</b>	<b>72%</b>	<b>0%</b>	<b>12</b>	<b>31</b>	<b>0</b>
Requisitos	9	20%	60%	40%		5	3	0
Análisis	13	30%	20%	80%		3	10	0
Diseño	13	30%	20%	80%		3	10	0
Plan pruebas	9	20%	20%	80%		2	7	0
<b>Implementación</b>	<b>119</b>	<b>100%</b>	<b>2%</b>	<b>28%</b>	<b>70%</b>	<b>2</b>	<b>33</b>	<b>83</b>
Codificación	71	60%		20%	80%	0	14	57
Prueba unitarias	6	5%		10%	90%	0	1	5
Proc.Instalación	12	10%		75%	25%	0	9	3
Proc.Integración	6	5%	20%	80%		1	5	0
Documentación	24	20%	5%	20%	75%	1	5	18
<b>Pruebas</b>	<b>43</b>	<b>100%</b>	<b>9%</b>	<b>46%</b>	<b>46%</b>	<b>4</b>	<b>20</b>	<b>20</b>
Pruebas	17	40%	10%	20%	70%	2	3	12
Implantación	15	35%	10%	90%		2	14	0
Incidencias	11	25%	5%	25%	70%	1	3	8
<b>TOTAL</b>	<b>216</b>					<b>27</b>	<b>86</b>	<b>103</b>

Se planifica en aproximadamente **2 meses**.

## 3 SERVICIOS DE IMPLANTACIÓN

### 3.1 Definición

Implantación de U Signer en un entorno vacío, de 0, esto es, se implanta una versión “*limpia*” de la plataforma sin partir de la base de datos ni de los desplegados de la anterior.

En paralelo a la implantación de U Signer se implantará también USignerRESTProxy configurado para funcionar contra el U Signer que acompaña.

### 3.2 Entornos

El proceso se realizaría en 3 entornos, DES PRE y PRO que acarrean 5 servidores (se indican los nombres que conocemos de los servidores actuales, entendemos que realmente serán otros nombres para los nuevos):

- Entorno de DES:
  - Servidor: des-app-empleat2
- Entorno de PRE:
  - Servidor: pre-app-empleat3
  - Servidor: pre-app-empleat4
- Entorno de PRO:
  - Servidor: pro-app-empleat3
  - Servidor: pro-app-empleat4

A continuación se detalla el entorno para la implantación de la plataforma U Signer, se enumeran las diferentes aplicaciones/herramientas de software que intervienen en el proceso, aprovechando la implantación de 0 las aplicaciones aquí identificadas son las que recomendamos, se identifica el nombre y versión cada una de ellas:

- Sistema de Virtualización:  
Oracle Solaris Zones
- Sistema Operativo:  
Oracle Solaris 11.3
- Motor de Base de Datos:  
Oracle Database 12c Enterprise Edition

- Servidor de Aplicaciones:  
JBoss 6.4 EAP o JBoss 7.0 EAP
- JDK:  
Oracle JDK 1.7.0\_80 de 64 bits.
- Entornos de PRO y PRE clusterizados.
- No existe HSM

### 3.3 Protocolo de actuación

Se define a continuación el orden de los pasos a realizar:

- Implantación en entorno de TEST de Wise Security Global de:
  - U Signer 1.0.5.0 (última versión liberada del producto)
  - U NetSigner 3.0.3 (última versión liberada del producto)
  - USignerRESTProxy 1.0.0 (primera versión liberada no Release Candidate del componente)
- Pruebas de aceptación en el entorno de TEST.

El objeto de la implantación y pruebas en entorno de TEST de Wise Security Global es la validación previa del entorno tecnológico propuesto por Base Gestió D'Ingressos. El conjunto de pruebas se divide en 3 escenarios:

- Conjunto de pruebas de aceptación de la plataforma U Signer, incluye pruebas de firma, verificación...
  - Conjunto de pruebas definidas para el componente USignerRESTProxy
  - Conjunto de pruebas definidas para U NetSigner
- 
- Implantación en el servidor de DES de:
    - U Signer 1.0.5.0
    - U NetSigner 3.0.3
    - USignerRESTProxy 1.0.0
  - Pruebas de aceptación en el entorno de DES.
  - Implantación en los 2 servidores de PRE de:
    - U Signer 1.0.5.0
    - U NetSigner 3.0.3
    - USignerRESTProxy 1.0.0
  - Pruebas de aceptación en el entorno de PRE.
  - Implantación en los 2 servidores de PRO de:
    - U Signer 1.0.5.0
    - U NetSigner 3.0.3

- USignerRESTProxy 1.0.0
- Pruebas de aceptación en el entorno de PRO.

### **3.4 Ubicación de los diferentes tipos de actuación**

- Todas las actuaciones sobre el entorno de TEST serán llevadas a cabo localmente en las oficinas de Wise Security Global.
- Todas las actuaciones sobre el entorno de DES serán llevadas a cabo localmente en las oficinas de Wise Security Global.
- Todas las actuaciones sobre el entorno de PRE serán llevadas a cabo localmente en las oficinas de Wise Security Global.
- Todas las actuaciones sobre el entorno de PRO serán llevadas a cabo in situ, desplazándose un consultor a las oficinas de Base Gestió D'Ingressos.

### **3.5 Estimación y planificación**

Se estima el esfuerzo en **104h**, distribuidas de la siguiente manera:

- Entorno de TEST: 5 jornadas
- Entorno de DES: 2 jornadas
- Entorno de PRE: 3 jornadas
- Entorno de PRO: 3 jornadas

Se planifica en aproximadamente **4 semanas**.

## 4 PROPUESTA ECONÓMICA

### 4.1 Servicios de implantación

Concepto	Unidades	Importe unitario	Importe total
Implantación en TEST, jornadas en local	5	429,00€	2.145,00€
Implantación en DES, jornadas en local	2	429,00€	858,00€
Implantación en PRE, jornadas en local	3	429,00€	1.287,00€
Implantación en PRO, jornadas in situ	3	858,00€	2.574,00€
<b>TOTAL</b>			<b>6.864,00€</b>

## 4.2 Desarrollo de USignerRESTProxy

Concepto	Unidades	Importe unitario	Importe total
Jornadas de desarrollo en remoto	27	429,00€	11.583,00€
<b>TOTAL</b>			<b>11.583,00€</b>

## 4.3 Notas

- Los importes indicados NO incluyen IVA.
- Facturación: Factura emitida a la firma del contrato.
- Vencimiento de las facturas: 30 días f/f.
- Las condiciones económicas reflejadas en la presente oferta son de carácter estrictamente confidencial
- Validez de la oferta: 60 días naturales desde su fecha de recepción.

## 5 REFERENCIAS DE LA OFERTA

El identificador de la oferta es: **PRO21.166**

A efectos administrativos, se incluyen a continuación los datos de contacto de la empresa ofertante **Wise Security Global**, para la resolución de cualquier duda que pudiese surgir en relación con la presente oferta.

**Razón social** Wise Security Global SL

**CIF** B-95732087

**Dirección** Alameda Rekalde, 34  
48009 Bilbao

**Persona de contacto** Unai Azurmendi Sorriketa  
Desarrollo de negocio

**Móvil** (+34) 625.70.13.12

**Correo electrónico** [unai.azurmendi@wsg127.com](mailto:unai.azurmendi@wsg127.com)

