Agencia de Tecnología
y Certificación Electrónica

Cliente de Arangí

Índice de contenido

1. Definiciones

- Firma PKCS#7: definida en la RFC 2315.
- Firma CMS: extensión de la firma PKCS#7 definida en la RFC-3852.
- Firma PAdES-LTV (PAdES Long Term Validation): definida en el estándar ETSI TS 102 778-4.
- Firma XAdES: definida en el estándar ETSI TS 101 903, actualmente en la versión 1.3.2.

2. Introducción

Las librerías Arangí son utilizadas para realizar diversas tareas relacionadas con la criptografía. Entre ellas se encuentra el realizar firmas y validaciones en diversos formatos.

El cliente de Arangí es un recubrimiento de las librerías Arangí para ser utilizado en entornos cliente. Consta de dos applets (firma y validación), siendo su utilización muy similar: cargar valores de entrada en el applet y llamar al método adecuado para obtener el resultado.

3. Los ejemplos

A lo largo de este documento se seguirán los ejemplos ofrecidos en la distribución del cliente de Arangí. Los ejemplos se encuentran el la carpeta 'examples'.

Para seguir los ejemplos:

- En la carpeta 'applet/arangi_client' se encuentran los jars utilizados por el cliente de Arangí, así como los JNLP que se utilizarán para crear los dos applets. Los jars se encuentran firmados por la 'Autoridad de Certificación de la Comunidad Valenciana', excepto los de Bouncy Castle, que están firmados por 'The Legion of the Bouncy Castle'.
- En la carpeta 'js/arangi_client' hay tres ficheros de javascript muy útiles a la hora de usar los applets:
    - deployJava.js: fichero proporcionado por SUN para el despliegue de aplicaciones Java en entornos cliente.
    - installer.js: proporciona métodos para cargar los dos applets del cliente de Arangí.
    - arangi_client.js: proporciona métodos para realizar firmas, cofirmas, contrafirmas y validaciones.
- En la carpeta 'js' se encuentran los ficheros javascript de JQuery, que se utiliza en los ejemplos para facilitar la programación javascript.
- En la raíz se encuentran los cinco ejemplos:
    - firma.html: realiza una firma XAdES-BES attached de un fichero o un contenido que hay que indicarle en los campos de texto dispuestos a tal efecto. Si se desea cambiar el tipo de firma sólo hay que modificar el valor de la variable 'signatureType'.
    - firmaDetached.html: realiza una firma XAdES-BES detached de un fichero o un contenido. Permite añadir una referencia del documento firmado a la firma.
    - firmaPadesVisible.html: realiza una firma PDF visible, en este caso un PadES-LTV.
    - cofirma.html: realiza la cofirma de una firma[1]. Únicamente hay que indicarle la firma, y en algún caso el documento que la originó.
    - contrafirma.html: realiza una contrafirma de una firma[1]. Únicamente hay que indicarle la firma, para ello hay que introducir el path a un fichero con la firma o escribir el contenido de la misma.
    - validacion.html: realiza la validación de una firma XAdES-BES attached. En este caso hay que introducir el path a un fichero con la firma o el contenido de la misma. Si la

[1] Sólo funciona con firmas XAdES

firma es detached se debería indicar el fichero firmado para poder realizar la validación. Si se desea cambiar el tipo de firma sólo hay que modificar el valor de la variable 'signatureType'.

4. Los applets

4.1. Requisitos

Los applets del cliente de Arangí requieren Java 1.6.0 Update 10 como mínimo para funcionar.

4.2. Librerías

Los applets utilizan los siguientes jars:

arangi_client-1.4.0.jar	Cliente de Arangí
arangi_base-1.4.1.jar, arangi-1.4.1.jar	Librerías Arangí
bcmail-jjdk15on-150b16.jar, bcprov-jdk15on-150b16.jar, bctsp-jdk15on-150b16.jar	Librerías Bouncy Castle: base criptográfica de Arangí
iaikPkcs11Wrapper-1.2.17.jar	Librería para trabajar con tarjetas criptográficas
log4j-1.2.13.jar	Librerías para tratar los logs de la aplicación
MITyCLibAPI-1.0.4_1.jar, MITyCLibXADES-1.0.4_2.jar, MITyCLibTSA-1.0.4_1 .jar	Librerías del Ministerio de Industria, Turismo y Comercio para tratar firmas XAdES
xmlsec-1.4.2-ADSI-1.0.jar, xalan-2.7.1.jar, serializer-2.7.1.jar	Librerías para trabajar con firma XML (base de XAdES)
itextpdf-5.1.2_1.jar	Librería para trabajar con el formato PDF
commons-lang-2.4.jar, commons-logging-1.0.4.jar, commons-httpclient-3.0.1.jar, commons- codec-1.2.jar	Utilidades

4.3. Errores

En los dos applets, si el método principal (sign o validateSignature) devuelve null es porque se ha producido algún error. Para obtener la pila del error hay que llamar al método getError(), que se encuentra implementado en los dos applets.

4.4. Internacionalización

El cliente de Arangí se ha realizado de forma que todos textos que se muestran al usuario y todos los mensajes que puede devolver se encuentran en ficheros de texto accesibles mediante un locale de Java. Para cambiar dicho locale basta con pasarle a los applets los parámetros 'language' y 'country'[2].

[2] De momento sólo se han incluido los textos en castellano.

4.5. Tipos de firma

Los tipos de firma soportados son los siguientes:

pkcs7	Firma PKCS#7 detached.
cms	Firma CMS detached.
pdf	Firma PDF con sello de tiempos y respuesta OCSP.
pades-ltv	Firma PDF longeva. Incluye sellos de tiempo y respuestas OCSP.
xades-bes-attached	Firma XAdES-BES attached.
xades-bes-detached	Firma XAdES-BES detached.
xades-t-attached	Firma XAdES-T attached. Incluye sellos de tiempo.
xades-t-detached	Firma XAdES-T detached. Incluye sellos de tiempo.
xades-xl-attached	Firma XAdES-X-L attached. Incluye sellos de tiempo y respuestas OCSP.
xades-xl-detached	Firma XAdES-X-L detached. Incluye sellos de tiempo y respuestas OCSP.

A la hora de validar se debe indicar uno de éstos tipos:

pkcs7	Firma PKCS#7 detached.
cms	Firma CMS detached.
pdf	Firma PDF (firma simple o PAdES-LTV).
xades	Firma XAdES

IMPORTANTE: Siempre hay que tener en cuenta que para las firmas que requieren de sellos de tiempo y/o respuestas OCSP el cliente de Arangí necesitará acceder a servidores, lo cual puede llegar a ser un problema si el ordenador cliente se encuentra protegido por un firewall demasiado restrictivo o necesita acceder a un proxy para salir a Internet. Por ello, en firmas longevas como XAdES se recomienda realizar en cliente firmas simples, como XAdES-BES, y promocionar dichas firmas a XAdES-T o XAdES-X-L en un entorno más controlado: al llegar al servidor de la aplicación.

4.6. El applet de firma

El applet de firma lista los certificados de los siguientes almacenes de claves:

- Almacén personal de Windows.
- Almacén de Firefox: funciona tanto en **Windows** como en **Linux**. En el caso de Linux también se listan los certificados de los dispositivos PKCS#11 asociados al almacén de Firefox.

El applet de firma acepta los siguientes métodos:

setSignatureType(String signatureType)	Indica al applet el tipo de firma a realizar.
setFileToSign(String filePath, String reference)	Le pasa al applet el path del fichero a firmar, así como una referencia.
setContentToSign (String content, String reference)	Le pasa al applet el contenido del fichero a firmar, así como una referencia.
setContentToSignBase64 (String contentB64, String reference)	Le pasa al applet el contenido del fichero a firmar en base64, así como una referencia[3]. El contenido que se firmará será el resultado de descodificar de base64 el parámetro 'contentB64'.
setSignatureFile(String filePath)	Le pasa al applet el path del fichero que contiene la firma. Se utiliza en el proceso de contrafirma.
setSignatureContent (String content)	Le pasa al applet el contenido de una firma. Se utiliza en el proceso de contrafirma.
setSignatureContentBase64 (String contentB64)	Le pasa al applet el contenido de una firma en base 64. Se utiliza en el proceso de contrafirma.
addSubjectDNFilterRule (String rule)	Añade una regla para seleccionar certificados al filtro Asunto (Subject DN).
addIssuerDNFilterRule (String rule)	Añade una regla para seleccionar certificados al filtro Emisor (Issuer DN).
addSerialNumberFilterRule (String serialNumber)	Añade una regla para seleccionar certificados al filtro de números de serie.
allKeyUsagesFilter (boolean allKeyUsages)	Le dice al filtro si se quieren todos los certificados independientemente de sus usos de clave. Por defecto el filtro tiene un valor false en este campo.
setMostrarSeleccionSiSoloUno(boolean mostrarSeleccionSiSoloUno)	Si al pasar el filtro sólo queda un certificado, se le puede indicar si mostrar la selección o directamente hacer la firma con ese certificado.
disableMozillaKeystore()	No se buscarán certificados en el almacén de Mozilla Firefox.
disableIExplorerKeystore()	No se buscarán certificados en los almacenes de Internet Explorer (Windows).
disablePkcs11Keystores()	No se buscarán certificados en los almacenes PKCS#11 (tarjetas)
useOnlyMozillaKeystore()	Sólo se buscarán certificados en el almacén de Mozilla Firefox.
useOnlyIExplorerKeystore()	Sólo se buscarán certificados en el almacén de Internet

	Explorer (Windows).
setFirmaPDFSinSelloTiempos (boolean firmaPDFSinSelloTiempos)	Si se firma PDF se puede indicar si se quiere con o sin sello de tiempos.
addParametroExtraFirma (String key, Object value)	Carga un parámetro extra para la firma.
setUrlTSA (String urlTSA)	Le pasa al applet la URL de la TSA con la que se desea realizar los sellos de tiempo.
resetFilter()	Elimina todas las reglas añadidas a los filtros.
getError()	En caso de error en la firma este método devuelve la pila del error producido.
sign ()	En base a los métodos utilizados para cargar el applet realiza la firma y devuelve ésta codificada en base64. Si se puede dar el caso de que la firma resultado sea un fichero grande (firmas attached o PDFs) es mejor utilizar el método 'signReturnStream'.
signReturnStream ()	En base a los métodos utilizados para cargar el applet realiza la firma. Esta se puede obtener por trozos llamando después al método 'getResultPiece' (de esta forma se pueden firmar ficheros grandes de varios megas).
coSign ()	En base a los métodos utilizados para cargar el applet realiza una cofirma y devuelve la firma codificada en base64. Si se puede dar el caso de que la firma resultado sea un fichero grande (firmas attached o PDFs) es mejor utilizar el método 'coSignReturnStream'.
coSignReturnStream ()	En base a los métodos utilizados para cargar el applet realiza una cofirma. Esta se puede obtener por trozos llamando después al método getResultPiece (de esta forma se pueden firmar ficheros grandes de varios megas).
counterSign ()	En base a los métodos utilizados para cargar el applet realiza una contrafirma y devuelve la firma contrafirmada codificada en base64. Si se puede dar el caso de que la firma resultado sea un fichero grande (firmas attached o PDFs) es mejor utilizar el método 'counterSignReturnStream'.
counterSignReturnStream ()	En base a los métodos utilizados para cargar el applet

	realiza una contrafirma. Esta se puede obtener por trozos llamando después al método getResultPiece (de esta forma se pueden firmar ficheros grandes de varios megas).
getResultPiece	Tras llamar a los métodos *ReturnStream se llamará a este método para obtener los trozos que compondrán la firma en base64. Cuando la llamada a este método devuelva una cadena vacía querrá decir que no queda por leer nada de la firma en base64.

El parámetro 'reference' que aparece en varios de los métodos indica el motivo de firma para PDFs y una referencia al documento firmado en las firmas XAdES. En este último caso estas son las posibilidades:

- Si se usa el método 'setFileToSign' puede dejarse el parámetro 'reference' a null, con lo que Arangí automáticamente ubicará como referencia dentro de la firma el path al fichero.
- Si se usa el método 'setContentToSign' o 'setContentToSignBase64' es obligatorio que el parámetro 'reference' tenga algún valor. Por ejemplo, si el documento se firma y se guarda en un gestor documental, la referencia podría ser la ubicación del documento en el gestor, o sea, algo que permita posteriormente obtener el documento que originó la firma

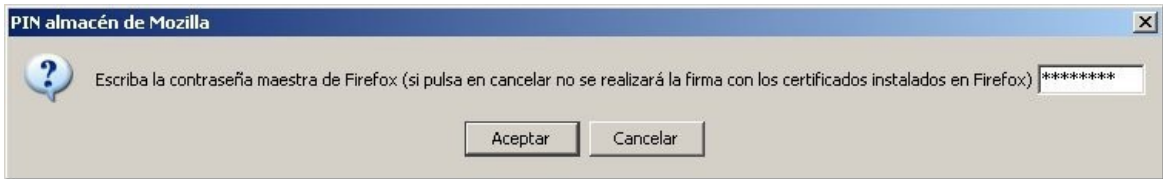
El método 'addParametroExtraFirma' puede cargar los siguientes valores:

- description (firmas XAdES) (String): será la descripción que irá en el tag DataObjectFormat de la firma.
- mime (firmas XAdES) (String): será el tipo MIME que irá en el tag DataObjectFormat de la firma.
- encoding (firmas XAdES) (String): será el encoding que irá en el tag DataObjectFormat de la firma. El String debe contener una URI.
- imagePath (firmas PDF visibles): path a la imagen que aparecerá en la firma. Es un campo opcional en las firmas PDF visibles, si no se incluye Acrobat Reader muestra una imagen por defecto.
- xBottomLeft, yBottomLeft (firmas PDF visibles): punto (x,y) que se encuentra en la esquina inferior izquierda de la firma visible. Es obligatorio si se quiere realizar una firma visible.
- xTopRight, xTopRight (firmas PDF visibles): punto (x,y) que se encuentra en la esquina superior derecha de la firma visible. Es obligatorio si se quiere realizar una firma visible.
- page (Firmas PDF visibles): página del PDF en el que se ubicará la firma. Es obligatorio si se quiere realizar una firma visible.

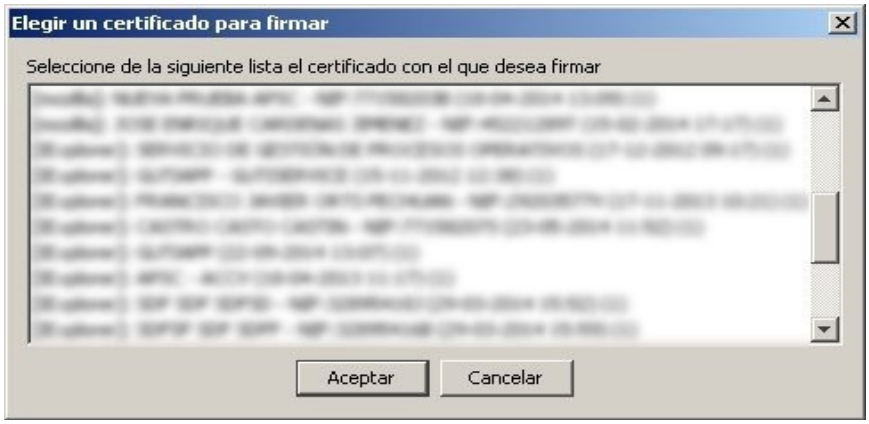
4.6.1.Firmar

Para realizar la firma el usuario deberá dar los siguientes pasos:

1. Si tiene instalado el navegador Firefox le aparecerá una pantalla para que introduzca su contraseña maestra. Si cancela la operación no se listarán los certificados de Firefox a la hora de realizar la firma.



2. Se le pedirá que elija un certificado de la lista. En la lista se muestran todos los certificados de Firefox, Internet Explorer y aquellos que se encuentren en tarjeta criptográfica y no se hayan propagado al almacén de claves de Windows. De cada certificado se muestra su common name y su fecha de caducidad. Si no elige ningún certificado se le avisará, y si realmente no quiere elegir ningún certificado se producirá un error.



3. Si el certificado se encuentra en tarjeta criptográfica se pedirá el PIN de la misma.

4.6.2. Cofirmar

El proceso de cofirma presenta la particularidad de que en algunos casos es necesario pasar al applet el documento que originó la firma y en otros no. Concretamente estas son las reglas que se deben seguir:

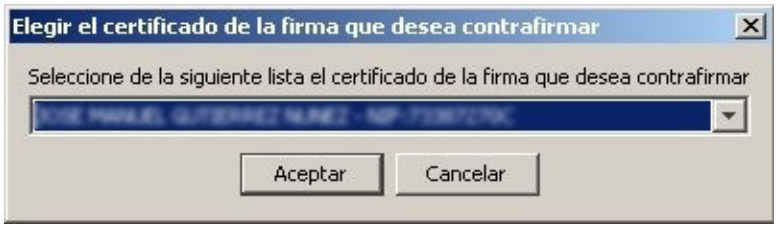
- Si la firma es attached no es necesario volver a pasarle el documento.
- Si la firma es detached:
    - Si la referencia no es a un fichero o a una URL es necesario pasarle el documento que originó la firma.

- Si la referencia es a un fichero o una URL no es necesario pasarle el documento que originó la firma, siempre que el fichero se encuentre accesible. Por ejemplo, si una firma que referencia a un fichero se lleva a otro ordenador, que no dispone del fichero en el path indicado, será necesario pasar el documento que originó la firma.

Por lo demás, el proceso es igual al de firma.

4.6.3. Contrafirmar

El proceso de contrafirma será muy similar al de firma. La única diferencia estriba en que si el objeto firma contiene varias firmas se mostrará la lista de certificados que han ocasionado dichas firmas para que el usuario elija cuál es la que desea contrafirmar.



4.6.4. Filtrando certificados

En ciertas ocasiones no se desea que aparezcan todos los certificados disponibles en el listado. Para ello se han definido 5 tipos de filtrados:

- Filtrado por el asunto (Subject DN): se pueden añadir las reglas que se quiera para seleccionar certificados por su subject DN. Hay que tener en cuenta que las reglas se evalúan mediante la conjunción O (OR). Si una regla incluye ',' se considerarán como varias reglas unidas mediante la conjunción Y (AND). Cada regla no es más que una cadena a buscar dentro del subject DN del certificado.
- Filtrado por emisor (Issuer DN): se pueden añadir las reglas que se quiera para seleccionar certificados por su issuer DN. El funcionamiento es el mismo que el filtrado por asunto.
- Filtrado por número de serie: cada regla es el número de serie buscado o parte de éste. Al igual que en los filtros anteriores cada regla se evalúa mediante la conjunción O (OR) con las restantes.
- Todos los certificados o sólo los de firma: por defecto sólo se muestran los certificados de firma, aunque este comportamiento puede cambiarse para que aparezcan todos los certificados. Para que un certificado sea de firma se pide que tenga alguno de estos usos de clave: Digital Signature o Non Repudiation.
- Filtrar por almacén de certificados: por defecto todos se encuentran activos. Si se desea se puede desactivar la búsqueda en algún almacén o se puede usar sólo uno de ellos.

Ejemplos:

Certificado 1:

Ubicación	Internet Explorer (Windows)
Subject DN	CN=PEPE MARTIN LOPEZ - NIF:77158203B,SERIALNUMBER=77158203B,GIVENNAME=PEPE,SURNAME=PRUEBA APSC,OU=Ciudadanos,O=Generalitat Valenciana,C=ES
Issuer DN	CN=SUB_CA_WINDOWS3,OU=PKIGVA,O=Generalitat Valenciana,C=ES
Número de serie	“8137467763903585750”

Certificado 2:

Ubicación	Internet Explorer (Windows)
Subject DN	CN= JOSE LUIS BUENAVENTURA RUBIO - NIF:72586545J,SERIALNUMBER=72586545J,GIVENNAME=JOSE LUIS, SURNAME=BUENAVENTURA RUBIO,OU=Ciudadanos,O=Generalitat Valenciana,C=ES
Issuer DN	CN=SUB_CA_WINDOWS3,OU=PKIGVA,O=Generalitat Valenciana,C=ES
Número de serie	“8137467763903585750”

Certificado 3:

Ubicación	Mozilla Firefox
Subject DN	CN=ROBERTA LOPEZ ALBERT – NIF:78546258H,SERIALNUMBER=78546258H,GIVENNAME=ROBERTA,SURNAME=LOPEZ ALBERT,OU=Ciudadanos,O=Generalitat Valenciana,C=ES
Issuer DN	CN=ACCV-CA2,OU=PKIGVA,O=Generalitat Valenciana,C=ES
Número de serie	“5423562418978522522”
Ubicación	

Filtrado 1:

Regla subject DN: “SERIALNUMBER=78546258H”

Regla subject DN: “SERIALNUMBER= 72586545J”

Resultado: Certificados 2 y 3.

Filtrado 2:

Regla subject DN: “LOPEZ”

Regla issuer DN: “SUB_CA_WINDOWS3”

Resultado: Certificado 1.

Filtrado 3:

Regla subject DN: “LOPEZ”

Regla número serie: “8137467763903585750”

Regla número serie: “8564852254657845236”

Regla número serie: “5423562418978522522”

Resultado: Certificados 1 y 3.

Filtrado 4:

Regla issuer DN: “SUB_CA_WINDOWS3,PKIGVA”

Resultado: Certificados 1 y 2.

Filtrado 5:

Regla subject DN: “LOPEZ,ROBERTA”

Resultado: Certificado 3.

Filtrado 6:

Regla subject DN: “SERIALNUMBER=78546258H”

Regla subject DN: “SERIALNUMBER= 72586545J”

Regla useOnlyMozillaKeystore: true

Resultado: Certificado 3.

4.7. El applet de validación

El applet de validación acepta los siguientes métodos:

setSignatureType(String signatureType)	Indica al applet el tipo de firma a validar.
setSignatureFile(String filePath)	Le pasa al applet el path del fichero que contiene la firma.
setSignatureContent (String content)	Le pasa al applet el contenido de la firma.
setSignatureContentBase64 (String contentB64)	Le pasa al applet el contenido de una firma en base 64.
setDetachedFile(String filePath)	Le pasa al applet el path del fichero firmado (sólo para firmas deteched).
setDetachedContent (String content)	Le pasa al applet el contenido del fichero firmado (sólo para firmas detached).
setDetachedContentBase64 (String contentB64)	Le pasa al applet el contenido del fichero firmado en base64 (sólo para firmas detached).
getError()	En caso de error en la firma este método devuelve la pila del error producido.
validateSignature ()	En base a los métodos utilizados para cargar el applet realiza la validación, devolviendo un XML con toda la información obtenida de ésta.

El método de validación devuelve un XML. Puede ver el esquema asociado y un ejemplo en el anexo 1.

El XML incluye información de cada una de las firmas:

- Validez de la firma: en caso de ser 'false' se escribe también el texto explicativo de porqué la firma no es válida.
- Fecha de realización de la firma.
- Información del certificado de firma: Common name, fecha de inicio y de fin de su periodo de validez y common name del certificado emisor.

- Datos del firmante: dependiendo del tipo de certificado se obtendrán unos datos u otros. Más abajo se muestra una tabla con la información obtenida de cada certificado.
- Datos del sello de tiempos: información del certificado de firma del sello y fecha proporcionada por el mismo.
- Datos de cada una de las respuestas OCSP: información del certificado de firma de la respuesta OCSP.
- Información de las contrafirmas realizadas a la firma.

En el ejemplo 'validacion.html' se puede ver cómo parsear la respuesta del applet de validación para mostrar todos los campos.

La información obtenida para cada tipo de certificado como 'datos del firmante' son:

Certificado de ciudadano de la Agencia de Tecnología y Certificación Electrónica	• *nif*: NIF/NIE • *name*: nombre y apellidos • *email*: e-mail
Certificados de empleado público o pertenencia a empresa de la Agencia de Tecnología y Certificación Electrónica	• *nif*: NIF/NIE • *name*: nombre y apellidos • *email*: e-mail • *cif*: CIF del organismo / empresa al que pertenece el propietario del certificado • *entity*: nombre del organismo / empresa al que pertenece el propietario del certificado • *organizational-unit* (opcional): unidad en la que trabaja el propietario del certificado • *nrrpnip* (opcional): Número de identificación del propietario del certificado
Certificado de entidad de la Agencia de Tecnología y Certificación Electrónica	• *cif*: CIF de la entidad • *name*: nombre de la entidad • *email*: e-mail • *agent-nif*: NIF del representante • *agent-name*: Nombre y apellidos del representante
Certificado de aplicación de la Agencia de Tecnología y Certificación Electrónica	• *name*: nombre de la aplicación • agent-name: nombre y apellidos del representante • email: e-mail del representante

DNIe	• *nif*: NIF/NIE • *name*: nombre y apellidos
Resto de certificados	

5. Utilizando el javascript de ejemplo

La creación y utilización de los applets en las páginas de una aplicación web puede ser implementada por cualquier desarrollador con experiencia en desarrollos web. De todas formas, y con objeto de facilitar la integración del cliente de Arangí, se pueden utilizar los ficheros javascript de los ejemplos. A continuación se intentará explicar cómo utilizar estos ficheros.

Si se utilizan los tres ficheros de javascript del ejemplo para incorporar el cliente de Arangí a una aplicación web, éstas son las variables que se pueden añadir para crear los applets e inicializarlos antes de realizar la acción (firmar o validar):

baseDownloadURL	Determina la URL de la carpeta donde se encuentran los jars del cliente. Es obligatorio, si no se define se produce un error.
localeLanguage (opcional)	Determina el lenguaje del locale que utilizará el cliente (ver apartado de internacionalización).
localeCountry (opcional)	Determina el país del locale que utilizará el cliente (ver apartado de internacionalización).
signatureType	Determina el tipo de firma. Es obligatorio, si no se define se produce un error.

En el fichero HTML es necesario que durante la carga se haga una llamada a la instalación del applet:

- loadArangiClientSignatureApplet(): carga el applet de firma.
- loadArangiClientValidationApplet(): carga el applet de validación.

IMPORTANTE: la llamada a estas funciones javascript debe hacerse dentro del BODY de la página HTML para que el applet pueda incluirse en él. Si la llamada se realiza antes el applet no se cargará y además no se verá ningún error que pueda dar alguna pista de lo que sucede.

Los métodos de carga de los applets comprueban que haya una versión de Java instalada y sea suficiente para ejecutar los applets. Si no es así avisa y redirige a una URL donde el usuario puede descargarse la última versión de Java.

Variables exclusivas para el applet de firma (ver el apartado *Filtrando certificados*):

arraySubjectFilterRules	Array javascript con las reglas para el filtro de selección por el asunto del certificado (Subject DN).
arrayIssuerFilterRules	Array javascript con las reglas para el filtro de selección por el emisor del certificado (Issuer DN).

arraySerialNumberFilterRules	Array javascript con las reglas para el filtro de selección por el número de serie del certificado.
allKeyUsagesFilter	Valor booleano que indica si se quieren listar todos los certificados (true) o sólo los de firma (false). Si no se define sólo se listarán los de firma.
disableMozillaKeystore	Valor booleano que indica si no se quiere buscar los certificados en el almacén de Mozilla Firefox.
disableIExplorerKeystore	Valor booleano que indica si no se quiere buscar los certificados en el almacén de Internet Explorer (Windows).
disablePkcs11Keystores	Valor booleano que indica si no se quiere buscar los certificados en los almacenes PKCS#11 (Tarjetas).
useOnlyMozillaKeystore	Valor booleano que indica que se quiere buscar los certificados sólo en el almacén de Mozilla Firefox.
useOnlyIExplorerKeystore	Valor booleano que indica que se quiere buscar los certificados sólo en el almacén de Internet Explorer (Windows).
mostrarSeleccionSiSoloUno	Valor booleano que indica si se debe mostrar el diálogo de selección cuando sólo hay un certificado para seleccionar.

Por otra parte, para elegir los ficheros o contenidos a firmar o validar se van a buscar campos HTML con unos determinados IDs. Para el caso de firma:

fileToSign	Path al fichero a firmar. Si este campo existe y tiene valor se usará para firmar aunque también exista y tenga valor el campo 'contentToSign'.
contentToSign	Contenido a firmar.
contentToSignBase64	Contenido a firmar en base64.
signatureReference	Referencia al documento firmado incluida en las firmas XAdES detached o motivo de firma para los PDFs. Si se deja a null en una firma XAdES detached la referencia de la firma será el path del fichero firmado.

Para el caso de cofirma:

signatureFile	Path al fichero que contiene la firma. Si este campo existe y tiene valor se usará para firmar aunque también exista y tenga valor el campo 'signatureContent'.
signatureContent	Contenido de la firma.

signatureContentBase64	Contenido de la firma en base64.
fileToSign	Path al fichero a firmar. Si este campo existe y tiene valor se usará para firmar aunque también exista y tenga valor el campo 'contentToSign'.
contentToSign	Contenido a firmar.
contentToSignBase64	Contenido a firmar en base64.

Para el caso de contrafirma:

signatureFile	Path al fichero que contiene la firma. Si este campo existe y tiene valor se usará para firmar aunque también exista y tenga valor el campo 'signatureContent'.
signatureContent	Contenido de la firma.
signatureContentBase64	Contenido de la firma en base64.

Para el applet de validación:

fileToValidate	Path al fichero a que contiene la firma. Si este campo existe y tiene valor se usará para firmar aunque también exista y tenga valor el campo 'contentToValidate'.
contentToValidate	Contenido de la firma.
contentToValidateBase64	Contenido de la firma en base64.
detachedFile	Path al fichero firmado. Se usa únicamente en las firmas detached (en las attached no tiene ninguna repercusión). Si este campo existe y tiene valor se usará para firmar aunque también exista y tenga valor el campo 'detachedContent'.
detachedContent	Contenido del fichero firmado. Se usa únicamente en las firmas detached (en las attached no tiene ninguna repercusión)
detachedContentBase64	Contenido del fichero firmado codificado a base64. Se usa únicamente en las firmas detached (en las attached no tiene ninguna repercusión)

6. Anexo 1. XSD y ejemplo del XML de respuesta en las validaciones

6.1. XSD para los XML de respuesta en las validaciones

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

<xs:element name="validation_result">

	<xs:complexType>

		<xs:sequence minOccurs="1" maxOccurs="1">

			<xs:element name="signature_type" type="xs:string" />

			<xs:element name="date" type="xs:string" />

			<xs:element name="signatures" type="signatures"/>

		</xs:sequence>

	</xs:complexType>

</xs:element>


<xs:complexType name="signatures">

	<xs:sequence minOccurs="1">

		<xs:element name="signature" type="signature"/>

	</xs:sequence>

</xs:complexType>


<xs:complexType name="signature">

	<xs:sequence maxOccurs="1" minOccurs="1">

		<xs:element name="valid" type="xs:boolean" />

		<xs:element name="result" type="xs:int" />
```

```
            <xs:element name="result_message" type="xs:string" />
            <xs:element name="date" type="xs:string" />
            <xs:element name="certificate" type="certificate" />
            <xs:element ref="signer_information" />
            <xs:element ref="timestamp" minOccurs="0" />
            <xs:element ref="ocsps" minOccurs="0" />
            <xs:element ref="countersignatures" minOccurs="0" />
        </xs:sequence>
    </xs:complexType>

    <xs:complexType name="certificate">
        <xs:sequence maxOccurs="1" minOccurs="1">
            <xs:element name="common_name" type="xs:string" />
            <xs:element ref="validity" />
            <xs:element name="issuer" type="xs:string" />
        </xs:sequence>
    </xs:complexType>

    <xs:element name="validity">
        <xs:complexType>
            <xs:sequence maxOccurs="1" minOccurs="1">
                <xs:element name="not_before" type="xs:string" />
                <xs:element name="not_after" type="xs:string" />
            </xs:sequence>
        </xs:complexType>
    </xs:element>
```

```
<xs:element name="signer_information">
        <xs:complexType>
                <xs:sequence maxOccurs="1" minOccurs="1">
                        <xs:element name="type" type="xs:string" minOccurs="0" />
                        <xs:element name="nif" type="xs:string" minOccurs="0" />
                        <xs:element name="cif" type="xs:string" minOccurs="0" />
                        <xs:element name="name" type="xs:string" minOccurs="0" />
                        <xs:element name="email" type="xs:string" minOccurs="0" />
                        <xs:element name="position" type="xs:string" minOccurs="0" />
                        <xs:element name="entity" type="xs:string" minOccurs="0" />
                        <xs:element name="organizational_unit" type="xs:string"
 minOccurs="0" />
                    <xs:element name="nrpnip" type="xs:string" minOccurs="0" />
                    <xs:element name="agent_nif" type="xs:string" minOccurs="0" />
                    <xs:element         name="agent_nombre"         type="xs:string"
minOccurs="0"/>
                </xs:sequence>
        </xs:complexTy
              pe>
 </xs:element>

         <xs:element
      name="timestamp">
        <xs:complexTy
              pe>
                <xs:sequence minOccurs="0">
                        <xs:element name="timestamp_certificate" type="certificate"
 minOccurs="1" maxOccurs="1"
              />
```

```
                    <xs:element name="date" type="xs:string" minOccurs="1"
maxOccurs="1" />
             </xs:sequence>

       </xs:complexType>

</xs:element>

<xs:element name="ocsps">
       <xs:complexTy
            pe>
             <xs:sequence maxOccurs="unbounded" minOccurs="0">

                    <xs:element ref="ocsp" minOccurs="1" />

             </xs:sequence>

       </xs:complexTy
            pe>

</xs:element>


<xs:element name="ocsp">

       <xs:complexTy
            pe>

             <xs:sequence minOccurs="1">

                    <xs:element name="ocsp_certificate" type="certificate"
minOccurs="1" maxOccurs="1"
             />

             </xs:sequence>

       </xs:complexTy
            pe>

</xs:element>


<xs:element name="countersignatures">

       <xs:complexTy
            pe>
```

```
            <xs:sequence maxOccurs="unbounded" minOccurs="0">
                  <xs:element name="countersignature" type="signature"
minOccurs="1" />
            </xs:sequence>
      </xs:complexType>
</xs:element>
</xs:schema>
```

6.2. Ejemplo de XML de respuesta en las validaciones

```
<?xml version="1.0" encoding="UTF-8"?>
<validation_result>
<signature_type>XAdES-X-L</signature_type>
<date>25-05-2011 14:02</date>
<signatures>
<signature>
      <valid>true</valid>
      <result>0</result>
      <result_message></result_message>
      <date>18-04-2011 16:05</date>
      <certificate>
            <common_name>JOSE MANUEL GUTIERREZ NUNEZ - NIF:73387270C
            </common_name>
```

```
        <validity>
                <not_before>18-10-2010 07:52</not_before>
                <not_after>17-10-2013 08:02</not_after>
        </validity>
        <issuer>ACCV-CA2</issuer>
</certificate>
<signer_information>
        <type>Ciudadano</type>
        <nif>73387270C</nif>
        <name>JOSE MANUEL Gutierrez NuÃ±ez</name>
        <email>jgutierrez@accv.es</email>
</signer_information>
<timestamp>
        <timestamp_certificate>
                <common_name>TSA1 ACCV</common_name>
                <validity>
                        <not_before>21-11-2006 18:52</not_before>
                        <not_after>18-11-2016 17:52</not_after>
                </validity>
                <issuer>Root CA Generalitat Valenciana</issuer>
        </timestamp_certificate>
        <date>18-04-2011 16:05</date>
</timestamp>
<ocsps>
        <ocsp>
                <ocsp_certificate>
```

```
                        <common_name>Servidor OCSP ACCV-CA2</common_name>
                        <validity>
                                <not_before>04-05-2010 15:36</not_before>
                                <not_after>08-04-2015 15:46</not_after>
                        </validity>
                        <issuer>ACCV-CA2</issuer>
                </ocsp_certificate>
        </ocsp>
        <ocsp>
                <ocsp_certificate>
                        <common_name>ocsp-gva</common_name>
                        <validity>
                                <not_before>11-09-2007 17:14</not_before>
                                <not_after>09-09-2012 16:14</not_after>
                        </validity>
                        <issuer>Root CA Generalitat Valenciana</issuer>
                </ocsp_certificate>
        </ocsp>
</ocsps>
<countersignatures>
        <countersignature>
                <valid>false</valid>
                <result>3</result>
                <result_message>El certificado está revocado</result_message>
                <date>17-05-2011 12:17</date>
                <certificate>
                        <common_name>JOSE MANUEL GUTIERREZ NUNEZ - NIF:73387270C
```

```
                </common_name>
                <validity>
                        <not_before>12-08-2008 09:21</not_before>
                        <not_after>12-08-2011 09:31</not_after>
                </validity>
                <issuer>ACCV-CA2</issuer>
        </certificate>
        <signer_information>
                <type>Ciudadano</type>
                <nif>73387270C</nif>
                <name>JOSE MANUEL GUTIERREZ NUNEZ</name>
                <email>jgutierrez@accv.es</email>
        </signer_information>
        <timestamp>
                <timestamp_certificate>
                        <common_name>TSA1 ACCV</common_name>
                        <validity>
                                <not_before>21-11-2006 18:52</not_before>
                                <not_after>18-11-2016 17:52</not_after>
                        </validity>
                        <issuer>Root CA Generalitat Valenciana</issuer>
                </timestamp_certificate>
                <date>17-05-2011 12:17</date>
        </timestamp>
    </countersignature>
</countersignatures>
```

```
</signature>

</signatures>

</validation_result>
```