



Manual de CallBack de PortaFIB

**Manual que mostra les diferents formes de Gestionar
els CallBack disponibles a PortaFIB.**



G VICEPRESIDÈNCIA
O I CONSELLERIA
I INNOVACIÓ,
B RECERCA I TURISME
/ DIRECCIÓ GENERAL
DESENVOLUPAMENT
TECNOLÒGIC



G VICEPRESIDÈNCIA
O I CONSELLERIA
I INNOVACIÓ,
B RECERCA I TURISME
/ FUNDACIÓ BIT

Informació general del document.

Descripció.

Títol:	Manual de CallBack de PortaFIB
Estat:	Esborrany/Aprovat
Versió:	1.0
Autor/s:	Antoni Nadal Bennasar
Creat:	14/01/2014
Modificat	13/02/2018
Fitxer:	Manual_de_CallBack_de_PortaFIB.odt

Històric de modificacions.

Comentari:	Autor/s:	Data:
Més informació sobre CallBack	A. Nadal	26/01/2016
CallBack REST	A. Nadal	18/08/2017
Separar la part de Callback del document Manual_de_WebServices_i_CallBack_de_PortaFIB.odt a aquest document.	A. Nadal	05/09/2017

Font documental.




Index de Contingut

1.-Introducció.....	4
2.-Crear WebService.....	5
3.-API de CallBack.....	7
3.1.-Classes WS CallBack.....	8
3.1.1.-Classe PortalFIBEvent.....	8
3.1.2.-Classe SigningRequest.....	9
3.1.3.-Classe Sign.....	10
3.1.4.-Classe Actor.....	10
4.-CallBack REST PortaFIB v1.....	11
4.1.-Introducció.....	11
4.2.-Implementar servei REST en aplicació externa.....	11
4.2.1.-Dependències.....	11
4.2.2.-Controlador de Servei REST.....	12
4.2.3.-Modificacions en el web.xml.....	13

1.- Introducció

L'API de CallBack serveix al servidor PortaFIB per enviar notifikacions via webservices dels events que ocorren en peticions relacionades amb els usuaris aplicació, és a dir, les peticions de firma enviades via WS emprant un usuari-aplicació. El funcionament normal és que des del lloc (aplicació externa) on es fa la petició de firma via webservices sigui el lloc on es posi un servidor webservices escoltant amb aquesta API.

Com és lògic en el servidor PortaFIB s'haurà definit en l'usuari aplicació el camp "URL de CallBack" que apuntarà a aquest servidor de webservices.

fundaciobit_usrapp	PortaFIB Callback v1.0	http://localhost:8080/portafib/cb/v1/PortaFIBCallBack	✓	PFI_USER	  
--------------------	---------------------------	---	---	----------	---

A continuació és mostra una captura de l'entorn web associat al CallBack d'exemple, que l'únic que fa es guardar els events rebuts i mostrar-los via web. En el meu cas tenc el servidor en localhost:8080, llavors la URL a aquest entorn d'exemple és la següent: "http://localhost:8080/portafib/portafibcb/":

Llistat de Callbacks Rebuts [\(WSDL\)](#) [Fer Net](#)

Data	Ver.	Entitat	Applicació	Tipus	Petició	Firma	Actor
26/01/2016 08:35:38	1	fundaciobit	fundaciobit_usrapp	REBUTJAT	ID:174934 Titol: Peticio de Test amb Custodia - 93715 Estat: REBUTJAT Info: null Rebuig: No és correcte	-	ID: fundaciobit_anadal NIF: 68456845C Name: Antoni Nadal Bennasar
26/01/2016 08:35:22	1	fundaciobit	fundaciobit_usrapp	ENPROCES	ID:174934 Titol: Peticio de Test amb Custodia - 93715 Estat: EN_PROCES Info: null	-	-
26/01/2016 08:29:03	1	fundaciobit	fundaciobit_usrapp	FIRMADA	ID:174865 Titol: Peticio de Test amb Custodia - 91323 Estat: FIRMAT Info: null Custodia: http://localhost:8080/custodia/index.jsp?hash=a3c007f144f0bdb6ae3d085a2f40cd9	-	-
26/01/2016 08:29:03	1	fundaciobit	fundaciobit_usrapp	FIRMA PARCIAL	ID:174865 Titol: Peticio de Test amb Custodia - 91323 Estat: EN_PROCES Info: null Rebuig:	ID: 174863 S/N: 1023931205 Issuer: C=ES,O=FNMT,OU=FNMT Clase 2 CA Subject: C=es,O=FNMT,OU=FNMT Clase 2 CA,OU=500070015,CN=NOMBRE NADAL ANTONI - NIF 68456845C	ID: fundaciobit_anadal NIF: 68456845C Name: Antoni Nadal
26/01/2016 08:28:35	1	fundaciobit	fundaciobit_usrapp	ENPROCES	ID:174865 Titol: Peticio de Test amb Custodia - 91323 Estat: EN_PROCES Info: null	-	-
26/01/2016 08:28:08	1	fundaciobit	fundaciobit_usrapp	PAUSAT	ID:174865 Titol: Peticio de Test amb Custodia - 91323 Estat: PAUSAT Info: null	-	-
26/01/2016 08:27:36	1	fundaciobit	fundaciobit_usrapp	ENPROCES	ID:174865 Titol: Peticio de Test amb Custodia - 91323 Estat: EN_PROCES Info: null	-	-

2.- Crear WebService

El servidor webservices es pot crear a partir del projecte ws\portafib_callback_server del codi font de PortaFIB que genera un jar amb webservices per un servidor JBoss amb CXF. Només s'hauran de modificar alguns noms de projecte dins el pom i dels paquets java, així com la implementació dels mètodes.

Podeu crear també el servei webservice a partir del wsdl que trobareu en la següent ruta del codi font de PortaFIB ws\portafib_callback_api\PortaFIBCallBack_v1.wsdl. Adjuntam una còpia per si no es té accés a aquest arxiu.

```
<?xml version='1.0' encoding='UTF-8'?><wsdl:definitions name="PortaFIBCallBackWsService"
targetNamespace="http://v1.server.callback.ws.portafib.caib.es/"
xmlns:ns1="http://schemas.xmlsoap.org/soap/http"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://v1.server.callback.ws.portafib.caib.es/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <xs:schema attributeFormDefault="unqualified" elementFormDefault="unqualified"
targetNamespace="http://v1.server.callback.ws.portafib.caib.es/"
xmlns:tns="http://v1.server.callback.ws.portafib.caib.es/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:element name="event" type="tns:event" />
      <xs:element name="eventResponse" type="tns:eventResponse" />
      <xs:element name="getVersionWs" type="tns:getVersionWs" />
      <xs:element name="getVersionWsResponse" type="tns:getVersionWsResponse" />
      <xs:complexType name="getVersionWs">
        <xs:sequence />
      </xs:complexType>
      <xs:complexType name="getVersionWsResponse">
        <xs:sequence>
          <xs:element name="return" type="xs:int" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="event">
        <xs:sequence>
          <xs:element minOccurs="0" name="event" type="tns:portaFIBEvent" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="portaFIBEvent">
        <xs:sequence>
          <xs:element minOccurs="0" name="actor" type="tns:actor" />
          <xs:element minOccurs="0" name="applicationID" type="xs:string" />
          <xs:element minOccurs="0" name="entityID" type="xs:string" />
          <xs:element minOccurs="0" name="eventDate" type="xs:dateTime" />
          <xs:element name="eventTypeID" type="xs:int" />
          <xs:element minOccurs="0" name="sign" type="tns:sign" />
          <xs:element minOccurs="0" name="signingRequest" type="tns:signingRequest" />
          <xs:element name="version" type="xs:int" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="actor">
        <xs:sequence>
          <xs:element minOccurs="0" name="administrationID" type="xs:string" />
          <xs:element minOccurs="0" name="ID" type="xs:string" />
          <xs:element minOccurs="0" name="name" type="xs:string" />
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="sign">
```

```

<xs:sequence>
<xs:element name="ID" type="xs:long" />
<xs:element minOccurs="0" name="issuer" type="xs:string" />
<xs:element minOccurs="0" name="serialNumber" type="xs:integer" />
<xs:element minOccurs="0" name="subject" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="signingRequest">
<xs:sequence>
<xs:element minOccurs="0" name="additionalInformation" type="xs:string" />
<xs:element minOccurs="0" name="custodyURL" type="xs:string" />
<xs:element name="ID" type="xs:long" />
<xs:element minOccurs="0" name="rejectionReason" type="xs:string" />
<xs:element name="state" type="xs:int" />
<xs:element minOccurs="0" name="title" type="xs:string" />
</xs:sequence>
</xs:complexType>
<xs:complexType name="eventResponse">
<xs:sequence />
</xs:complexType>
<xs:element name="CallBackFault" type="tns:CallBackFault" />
<xs:complexType name="CallBackFault">
<xs:sequence />
</xs:complexType>
</xs:schema>
</wsdl:types>
<wsdl:message name="getVersionWs">
  <wsdl:part element="tns:getVersionWs" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="CallBackException">
  <wsdl:part element="tns:CallBackFault" name="CallBackException">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="eventResponse">
  <wsdl:part element="tns:eventResponse" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="event">
  <wsdl:part element="tns:event" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="getVersionWsResponse">
  <wsdl:part element="tns:getVersionWsResponse" name="parameters">
  </wsdl:part>
</wsdl:message>
<wsdl:portType name="PortaFIBCallBackWs">
  <wsdl:operation name="getVersionWs">
    <wsdl:input message="tns:getVersionWs" name="getVersionWs">
    </wsdl:input>
    <wsdl:output message="tns:getVersionWsResponse" name="getVersionWsResponse">
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="event">
    <wsdl:input message="tns:event" name="event">
    </wsdl:input>
    <wsdl:output message="tns:eventResponse" name="eventResponse">
    </wsdl:output>
    <wsdl:fault message="tns:CallBackException" name="CallBackException">
    </wsdl:fault>
  </wsdl:operation>
</wsdl:portType>

```

```
<wsdl:binding name="PortaFIBCallBackWsServiceSoapBinding" type="tns:PortaFIBCallBackWs">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="getVersionWs">
    <soap:operation soapAction="" style="document" />
    <wsdl:input name="getVersionWs">
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="getVersionWsResponse">
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="event">
    <soap:operation soapAction="" style="document" />
    <wsdl:input name="event">
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output name="eventResponse">
      <soap:body use="literal" />
    </wsdl:output>
    <wsdl:fault name="CallbackException">
      <soap:fault name="CallbackException" use="literal" />
    </wsdl:fault>
  </wsdl:operation>
</wsdl:binding>
<wsdl:service name="PortaFIBCallBackWsService">
  <wsdl:port binding="tns:PortaFIBCallBackWsServiceSoapBinding" name="PortaFIBCallBackWs">
    <soap:address location="http://localhost:8080/portafib/cb/v1/PortaFIBCallBack" />
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

3.- API de CallBack

Aquesta API de CallBack només consta de dos mètodes:

```
@WebService
public interface PortaFIBCallBackWs {

    @WebMethod
    public int getVersionWs();

    @WebMethod
    public void event(@WebParam(name = "event") PortaFIBEvent event) throws CallbackException;

}
```

El primer, *getVersionWs()*, és només un mètode per versionar els WS. Ha de retornar 1.

El segon, *event(...)*, és el que utilitza PortaFIB per notificar als clients canvis en el procés de firma que els Usuaris-Aplicació han enviat a PortaFIB:

Prototipus:
<code>public void event (@WebParam(name = "event") PortaFIBEvent event) throws CallbackException;</code>
Descripció:
Rep les notificacions de canvis en algun procés de firma que ha enviat algun Usuari-Aplicació al que se li ha associat aquest WS per rebre informació.
Paràmetres:
<ul style="list-style-type: none"> PortaFIBEvent event: Bean que conté tota la informació d'algun event o canvi en el procés de firma. Veure punt "3.1.1.-Classe PortalFIBEvent" per més detalls.
Resposta:
<ul style="list-style-type: none"> No respon res al servidor. Això indica que tot ha anat correctament. Si és vol informar a PortaFIB que aquest CallBack no accepta la notificació o no ha pogut processar-la, llavors s'ha de realitzar llançant una excepció de tipus CallbackException (Veure següent apartat)
Excepcions:
<ul style="list-style-type: none"> Per informar a PortaFIB que aquest CallBack no accepta la petició o durant el procés d'aquesta notificació s'ha produït un error, llavors s'ha de realitzar llançant una excepció de tipus CallbackException. Exemple d'ús: <code>throw new CallbackException("Error processant notificació");</code>

3.1.- Classes WS CallBack

3.1.1.- Classe PortalFIBEvent

Els camps de l'estructura 5.2.1.- PortalFIBEvent són els següents:

Nom	Tipus	Descripció
eventTypeID	int	Indica el tipus d'event. El llistat de tipus es descriu en el punt "3.1.1.1.-Taula de tipus d'events"
version	int	Versió del WS del Servidor
eventDate	Date	Data en que s'ha produït l'event
applicationID	String	És l'aplicació a qui va dirigida la informació, és a dir, la que ha realitzat la Petició de Firma
entityID	String	És l'entitat o organització
signingRequest	SigningRequest	Informació sobre la Petició De Firma. Els detalls d'aquesta classe es mostren al punt "3.1.2.-Classe SigningRequest"
sign	Sign	Informació sobre la firma. Aquest camp té valor només quan el tipus d'event és NOTIFICACIOAVIS_FIRMA_PARCIAL. Podeu trobar més detalls al punt "3.1.3.-Classe Sign"

actor	Actor	És la persona que ha realitzat l'acció. Aquest camp té valor només quan el tipus d'event és NOTIFICACIOAVIS_FIRMA_PARCIAL o NOTIFICACIOAVIS_PETICIO_REBUTJADA, que indica qui ha realitzat la firma o qui ha rebutjat la petició. Més informació al següent punt "3.1.4.-Classe Actor"
-------	-------	--

3.1.1.1.- Taula de tipus d'events

A continuació es descriuen els tipus d'events que existeixen. Es important comentar que aquests events només es llancen per part del servidor de PortaFIB sobre les Peticions de Firma que han sigut enviades a traves de WS enprant Usuaris-Aplicació.

Valor	Nom/Descripció
0	NOTIFICACIOAVIS_PETICIO_EN_PROCES: Indica que una Petició de Firma s'ha posat s'ha iniciat o reiniciat després de ser pausada.
50	NOTIFICACIOAVIS_FIRMA_PARCIAL: Indica si un usuari ha realitzat una firma. Com que en una petició de Firma hi pot haver múltiples destinataris (i els delegats associats), aquesta notificació només informa que un d'aquest destinataris ha signat.
60	NOTIFICACIOAVIS_PETICIO_FIRMADA: Indica que la petició de firma ha finalitzat correctament, és a dir, s'han realitzat totes les firmes requerides.
70	NOTIFICACIOAVIS_PETICIO_REBUTJADA: Indica que una petició ha sigut rebutjada per algun dels destinataris o delegats.
80	NOTIFICACIOAVIS_PETICIO_PAUSADA: Indica que una petició ha sigut pausada.

3.1.2.- Classe SigningRequest

Els camps d'aquesta classe són els següents:

Nom	Tipus	Descripció
iD	long	Identificador de la Petició de Firma
title	String	Títol de la petició de firma
state	int	Estat de la petició de firma. Els valors poden ser: <ul style="list-style-type: none"> TIPUSESTATPETICIODEFIRMA_NOINICIAT = 0; TIPUSESTATPETICIODEFIRMA_ENPROCES = 1; TIPUSESTATPETICIODEFIRMA_PAUSAT = 2; TIPUSESTATPETICIODEFIRMA_REBUTJAT = 3; TIPUSESTATPETICIODEFIRMA_FIRMAT = 4;
rejectionReason	String	Quan la petició es rebutjada aquest camp conté la raó de rebuig.
additionalInformation	String	Camp per a futurs usos.
custodyURL	String	Si la petició ha definit custòdia, llavors aquest camp conté la URL al fitxer custodiat.

3.1.3.- Classe Sign

Representa una firma. Els camps d'aquesta classe són els següents:

Nom	Tipus	Descripció
iD	long	Identificador emprat pel sistema PortaFIB per identificar aquesta firma
serialNumber	BigInteger	Identificador del certificat emprat per firmar.
issuer	String	Informació de la persona extreta del certificat
subject	String	Informació del tipus de certificat o de l'autoritat certificadora que ha emès el certificat.

3.1.4.- Classe Actor

Els camps d'aquesta classe són els següents:

Nom	Tipus	Descripció
iD	String	Identificador de l'usuari-entitat dins PortaFIB.
name	String	Nom de la persona
administrationID	String	NIF de la persona

4.- CallBack REST PortaFIB v1

Nou a v1.1.3

4.1.- Introducció

CallBack REST funciona de forma similar a l'API de CallBack WS, però fa servir serveis REST per a la comunicació amb les Aplicacions. Com hem dit abans, el CallBack serveix al servidor PortaFIB per enviar notifikacions via REST dels events que ocorren en peticions relacionades amb els usuaris aplicació, és a dir, les peticions de firma enviades via WS emprant un usuari-aplicació. Aquest nou Servei s'ha implementat ja que és molt més senzill fer feina amb REST, ja que es pot incorporar directament en la capa WEB enlloc de fer tot un muntatge per escoltar peticions WS.

Únicament hem de configurar l'Usuari-Aplicació per a usar RETS en les notifikacions i definir la ruta o URL on l'aplicació externa ha definit el servei REST.

Callback Versió (*)	PortaFIB Callback REST v1.0
Callback URL (*)	http://localhost:8080/portafib/cbrest/v1/event

En l'exemple anterior la ruta apunta al Servei Rest de proves que incorpora PortaFIB.

4.2.- Implementar servei REST en aplicació externa

A continuació es descriuen les passes per a que una aplicació externa implementi el servei REST i d'aquesta forma pugui rebre les notifikacions de les peticions que ha enviat a PortaFIB.

Podeu trobar un exemple del que s'explicarà en el codi font de PortaFIB en el directori [portafib-1.1]\ws\portafib_callback_server_rest.

4.2.1.- Dependències

Aquí teniu les dependències a afegir al pom.xml del mòdul Web:

```
<dependency>
  <artifactId>portafib-callback-beans</artifactId>
  <groupId>es.caib.portafib</groupId>
  <version>1.1</version>
</dependency>

<dependency>
  <groupId>com.sun.jersey</groupId>
  <artifactId>jersey-server</artifactId>
  <version>1.8</version>
</dependency>
```

```
<dependency>
  <groupId>com.sun.jersey</groupId>
  <artifactId>jersey-json</artifactId>
  <version>1.8</version>
  <exclusions>

    <exclusion>
      <groupId>stax</groupId>
      <artifactId>stax-api</artifactId>
    </exclusion>

    <exclusion>
      <groupId>com.sun.xml.bind</groupId>
      <artifactId>jaxb-impl</artifactId>
    </exclusion>

  </exclusions>
</dependency>
```

4.2.2.- Controlador de Servei REST

Aquí teniu un exemple de “receptor” de peticions REST. En aquesta classe hem de tenir en compte el package (que es modificarà pel propi de l'aplicació) i la ruta o Path relatiu a aquest servei definit en la classe i en el mètode. En l'exemple següent són /v1 i /event, que conformen una ruta relativa /v1/event. Aquestes dues anotacions les hem de tenir en compt per gestionar els punts següents.

```
package es.caib.portafib.callback.rest.server.v1;

import javax.ws.rs.Consumes;
import javax.ws.rs.POST;
import javax.ws.rs.Path;
import javax.ws.rs.core.MediaType;
import javax.ws.rs.core.Response;
import es.caib.portafib.callback.beans.v1.PortaFIBEvent;
import es.caib.portafib.callback.beans.v1.testers.PortaFIBEventStore;

/**
 * @author anadal
 */
@Path("/v1")
public class PortaFIBRestServiceV1 {

    @POST
    @Path("/event")
    @Consumes(MediaType.APPLICATION_JSON)
    public Response event(PortaFIBEvent event) {

        try {
            // Fer el que sigui !!!!!
            return Response.status(200).entity("OK").build();
        } catch (Throwable th) {
            String msg = "Error desconegut processant event de Peticio de Firma REST: "
                + th.getMessage();
            th.printStackTrace();
            return Response.status(500).entity(msg).build();
        }
    }
}
```

```
}
```

Les classes i la gestió de les mateixes es descriuen en l'apartat de API Callback de WS.

4.2.3.- Modificacions en el web.xml

Ara toca associar el servlet de REST amb el controlador que hem programat en el punt anterior:

- (1) Propietat “com.sun.jersey.config.property.packages”: aquesta ha de constar de dos packages: el primer és el package d'on es troba el controlador i el segon sempre ha de ser “es.caib.portafib.callback.beans.v1”. S'han de separar per un ','.
- (2) “Url-pattern”: Aquesta propietat indica la url base per tots els serveis definits en els controladors. En l'exemple de web.xml aquesta URL val /callbackrest. Si concatenem aquest valor amb el definit en el controlador, la ruta d'accés a aquest servei seria `http://HOST:PORT/{aplicació_context}/callbackrest/v1/event`, que és la que d'ha de posar en la definició de l'Usuari-Aplicació (Veure Introducció)

```
<servlet>
  <servlet-name>jersey-servlet</servlet-name>
  <servlet-class>com.sun.jersey.spi.container.servlet.ServletContainer</servlet-class>
  <init-param>
    <param-name>com.sun.jersey.config.property.packages</param-name>
    <param-value>
es.caib.portafib.callback.rest.server.v1;es.caib.portafib.callback.beans.v1
    </param-value>
  </init-param>
  <init-param>
    <param-name>com.sun.jersey.api.json.POJOMappingFeature</param-name>
    <param-value>true</param-value>
  </init-param>
  <load-on-startup>1</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>jersey-servlet</servlet-name>
  <url-pattern>/callbackrest/*</url-pattern>
</servlet-mapping>
```