PORTADA

Contenido

[Índice de figuras 3](#_Toc530486480)

[Objeto 4](#_Toc530486481)

[Alcance 4](#_Toc530486482)

[Siglas 4](#_Toc530486483)

[Documentos de referencia 4](#_Toc530486484)

[Diseño de la arquitectura 4](#_Toc530486485)

[Arquitectura del sistema 4](#_Toc530486486)

[Directorios utilizados por la aplicación y ficheros de log 4](#_Toc530486487)

[Diseño de componentes 5](#_Toc530486488)

[Consideraciones previas 6](#_Toc530486489)

[Definición de plataformas 6](#_Toc530486490)

[Definición de servicios 6](#_Toc530486491)

[Monitoriz@ 7](#_Toc530486492)

[Arquitectura del componente 7](#_Toc530486493)

[Componente clave-status-app 8](#_Toc530486494)

[Operativa del agente de monitorización 8](#_Toc530486495)

[Parametrización peticiones 8](#_Toc530486496)

[Componente clave-web 15](#_Toc530486497)

[Base de datos 15](#_Toc530486498)

[Esquema 15](#_Toc530486499)

[Tablas 16](#_Toc530486500)

[Secuencias 16](#_Toc530486501)

[Carga inicial de datos 16](#_Toc530486502)

[Parámetros del sistema 16](#_Toc530486503)

[Monitoriz@ 16](#_Toc530486504)

[SAMLEngine 16](#_Toc530486505)

# Índice de figuras

[Ilustración 1 Diseño de la arquitectura 5](#_Toc530554633)

[Ilustración 2 Diagrama de componentes 6](#_Toc530554634)

[Ilustración 3 Diagrama de componente Monitoriza 7](#_Toc530554635)

# Objeto

El objeto del presente documento es detallar el diseño técnico del nuevo módulo de monitorización, basado en la solución base de Monitoriz@, mediante la que se producirán agentes de monitorización de las plataformas de eIDAS y @Firma.

# Alcance

Los objetivos que cubre este documento son los siguientes:

* La arquitectura del sistema.
* Información de compilación y despliegue del sistema.
* Diseño y parametrización de los principales módulos del sistema.
* Diseño del modelo de datos.
* Parametrización y configuración del sistema.
* Interacción con otros sistemas de información.

# Siglas

HTTP Hypertext Transfer Protocol

HTTPS Hypertext Transfer Protocol Secure

URL Uniform Resource Locator

SAML Security Assertion Markup Language

MVC Modelo-Vista-Controlador

# Documentos de referencia

| Id Documento | Nombre Documento | Descripción |
| --- | --- | --- |
| Manual Monitoriz@ |  |  |

# Diseño de la arquitectura

## Arquitectura del sistema

El Sistema es implementado como una aplicación Java para entornos Web, y cuya arquitectura es definida por el **patrón MVC**, apoyándose en el framework de desarrollo Spring, y principalmente en Spring MVC para la gestión de eventos y comunicaciones entre las distintas capas que conforman el Sistema.

El siguiente diagrama presenta la pila tecnológica del Sistema.

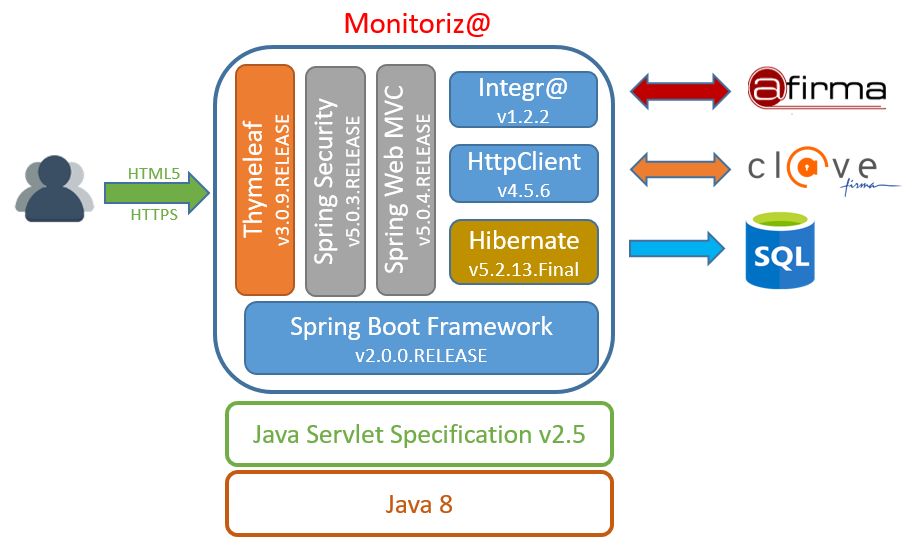


Ilustración 1 Diseño de la arquitectura

* **Capa de presentación:** La interfaz gráfica de usuario está desarrollada en HTML5, empleándose para ello el motor de plantillas **Thymeleaf v3.0.9.RELEASE**, y apoyándose adicionalmente los siguientes componentes:
  + **jQuery v3.2.1**. Biblioteca JavaScript multiplataforma que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos y agregar interacción AJAX.
  + **Datatables v1.10.16.** (http://datatables.net). Componente jQuery que facilita y amplía la agregación de controles y la interacción avanzada sobre tablas HTML.
  + **Bootstrap v4.0.0**. Framework para el desarrollo y diseño de aplicaciones web con HTML, CSS y JavaScript.
* **Capa de control**. Esta capa se encarga de responder y validar las peticiones efectuadas por los usuarios finales desde la capa de presentación, para finalmente interactuar con el modelo y resolver la solicitud efectuada. La capa de control se apoya en los componentes que aporta el framework Spring para el desarrollo web, empleándose los siguientes componentes:
  + **Spring Web MVC v5.0.4.RELEASE**. Habilita la aplicación del patrón MVC en el Sistema, gestionando principalmente la comunicación entre la vista de presentación y la capa de control, y otras funcionalidades sobre la vista del Sistema, como por ejemplo, la funcionalidad de multidioma.
  + **Spring Security v5.0.3.RELEASE**. Incorpora mecanismos de seguridad, autenticación y autorización al Sistema.
  + **Spring Boot Framework v2.0.0.RELEASE**. Este componente actúa como framework base para el desarrollo del Sistema, facilitando la reutilización de componentes y el acceso a archivos de configuración, por ejemplo.
* **Modelo**. Capa responsable de trabajar con los datos o información, ofreciendo al resto de capas la lógica necesarios para su consulta, actualización, inserción y borrado. La implementación de esta capa emplea los siguientes componentes:
  + **Hibernate v5.2.13.Final**. Componente utilizado para resolver el mapeo objeto-relacional (ORM) y el acceso al sistema de persistencia.
* **Servicios y lógica de negocio**. Esta capa ofrece los gestores y fachadas necesarios para responder a las peticiones, realizadas desde la capa de control y servicios publicados, sobre el modelo del Sistema. Para ello se apoya en el componente **Spring Boot Framework v2.0.0.RELEASE.**

La integración con los servicios de @firma y cl@ve se realizan a través de **Integr@ v1.2.2.** y **HttpClient v4.5.6.**.

## Directorios utilizados por la aplicación y ficheros de log

Los directorios empleados por el Sistema son los siguientes:

* **Publicación web**: El contexto web que aloja al Sistema es monitoriza-web. Desde este contexto es posible acceder a los servicios publicados y a la interfaz gráfica de usuario.
* **Sistema de archivos**:
  + **Despliegue**: El binario del Sistema se corresponde con dos archivos WAR, los cuales incluyen inicialmente todos los recursos requeridos para las invocaciones para los agentes de monitorización y para la ejecución del proyecto Monitoriz@ para su correcto funcionamiento.
  + **Archivos de trazas de ejecución:** Directorio donde son alojados los archivos de trazas de ejecución del Sistema, generados mediante la librería log4j. Este directorio puede ser configurado en el fichero “monitoriza-log4j.xml“.
  + **Almacenes de certificados electrónicos**. La securización de las comunicaciones y la invocación de ciertos servicios web pueden requerir la configuración de almacenes de certificados que alojen los certificados electrónicos empleados para tal propósito.
  + **Almacenes de archivos comprimidos.** Los archivos comprimidos en ZIP almacenados desde los servicios publicados vía web, serán almacenados en una ruta que será configurada en el properties “staticMonitorizaConfig.properties” en el atributo “directories.request.root.path”.
* **Variables requeridos por SAML:**
  + **assertionConsumerServiceURL:** Dirección del servicio de confianza de la petición Saml.
  + **SPApplication:** Dirección destino de la petición Saml.
  + **providerName:** Nombre del proveedor de la petición Saml.
  + **SPType:** Tipo del proveedor del servicio de la petición Saml. Es una enumerado que soporta las cadena de texto, “public” y “private”.
  + **Attributes:** Listado de atributos. Es una enumerado que soporta las cadena de texto, “familyName”, “firstName”, “dateOf Birth”, “personIdentifier”, “legalName” y “legalPersonIdentifier”.

# Diseño de componentes

La siguiente imagen ilustra gráficamente el diagrama de componentes del activo Cl@ve-Monitori@.

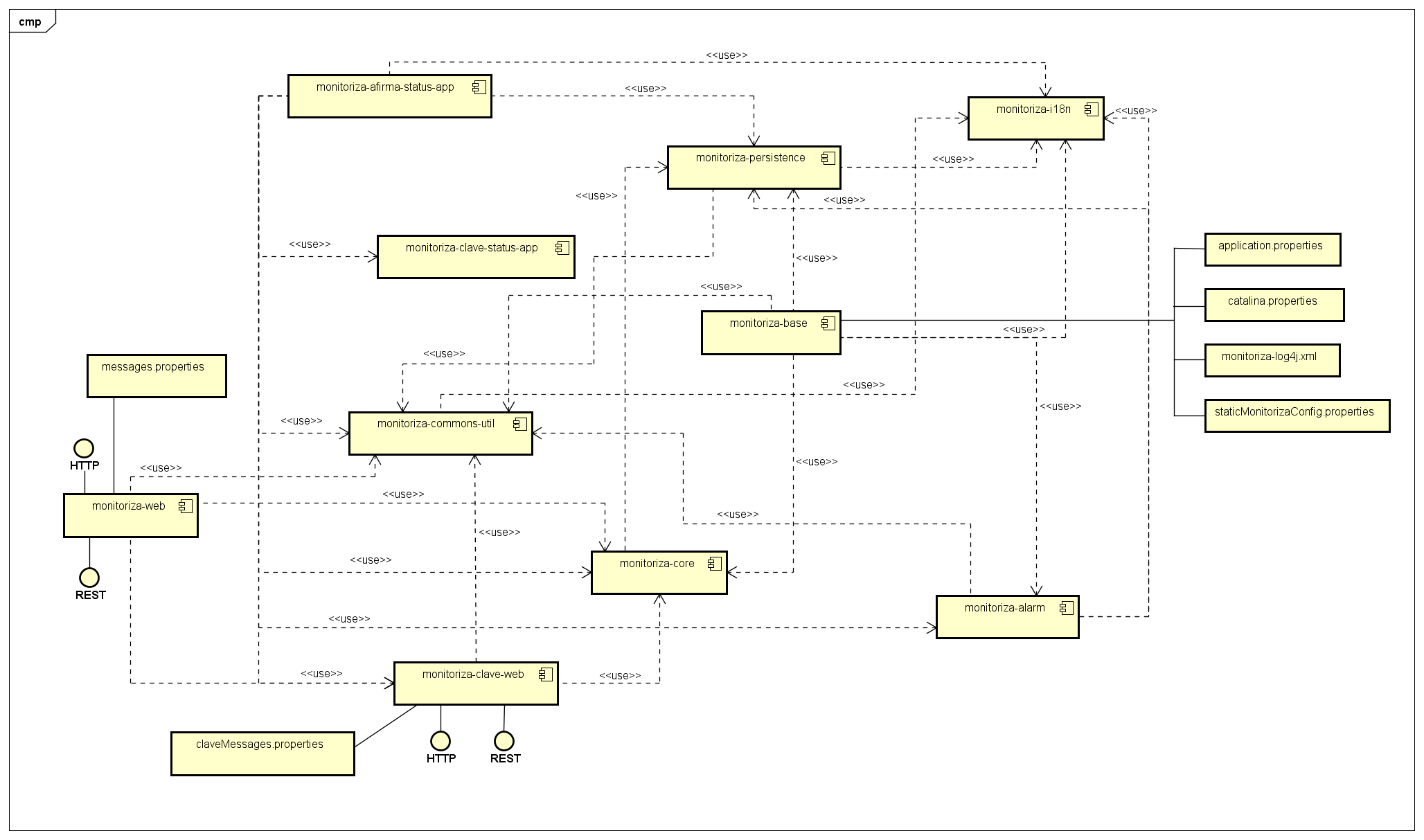


Ilustración 2 Diagrama de componentes

## Consideraciones previas

### Definición de plataformas

Mediante la herramienta de monitorización se van a incorporar las plataformas que conectará con la pasarela de eIDAS y con el IdP de @firma, para es necesario lo que se introducirá en la carga inicial del sistema. Cada uno de los agentes de monitorización apunta a una plataforma distinta y presenta la siguiente información:

* Nombre: Denominación de la plataforma.
* Host: Dirección base a donde apunta el agente de monitorización.
* Puerto: Puerto de la dirección base.
* Securizar peticiones: Indica si la plataforma utiliza HTTP o HTTPS.
* Contexto del servicio: Contexto a donde apunta el agente de monitorización.

### Definición de servicios

Mediante la herramienta de monitorización se van a incorporar aquellos elementos necesarios para el funcionamiento de los servicios e incorporando las plataformas.

Se crea un servicio por cada agente de monitorización a realizar y presenta la siguiente información:

* **Nombre:** Se indica el nombre que tendrá el servicio.
* **Plataforma:** Identifica la plataforma a la que se va a conectar el servicio.
* **Tipo de Servicio:** Identifica el tipo de servicio según la plataforma seleccionada.
* **Endpoint:** Punto de la dirección URL en el que se encuentra el agente de monitorización.
* **Timeout:** Tiempo de conexión del servicio.
* **Timer:** Identifica el temporizador con el que se va a controlar el servicio.
* **Umbral degradado (milisegundos):** Establece el tiempo en milisegundos el umbral máximo del tiempo de la petición del servicio.
* **Umbral perdido (%):** Establece el porcentaje máximo de peticiones perdidas de la petición de servicio.
* **Alarma:** Identifica la alarma que saltara en el caso de producirse un problema en la petición de servicio.
* **Archivo de peticiones en formato ZIP:** Se establece el archivo ZIP que contiene la petición XML del servicio.

## Monitoriz@

Sistema Base de monitorización desarrollado en principio para la herramienta @firma y Ts@. Su responsabilidad es suministrar funcionalidades, utilidades y recursos de uso común a los componentes desarrollados que conforman la solución final.

Este componente presenta una arquitectura compleja, estando a su vez conformado por otros componentes, como se ilustra en el siguiente diagrama:

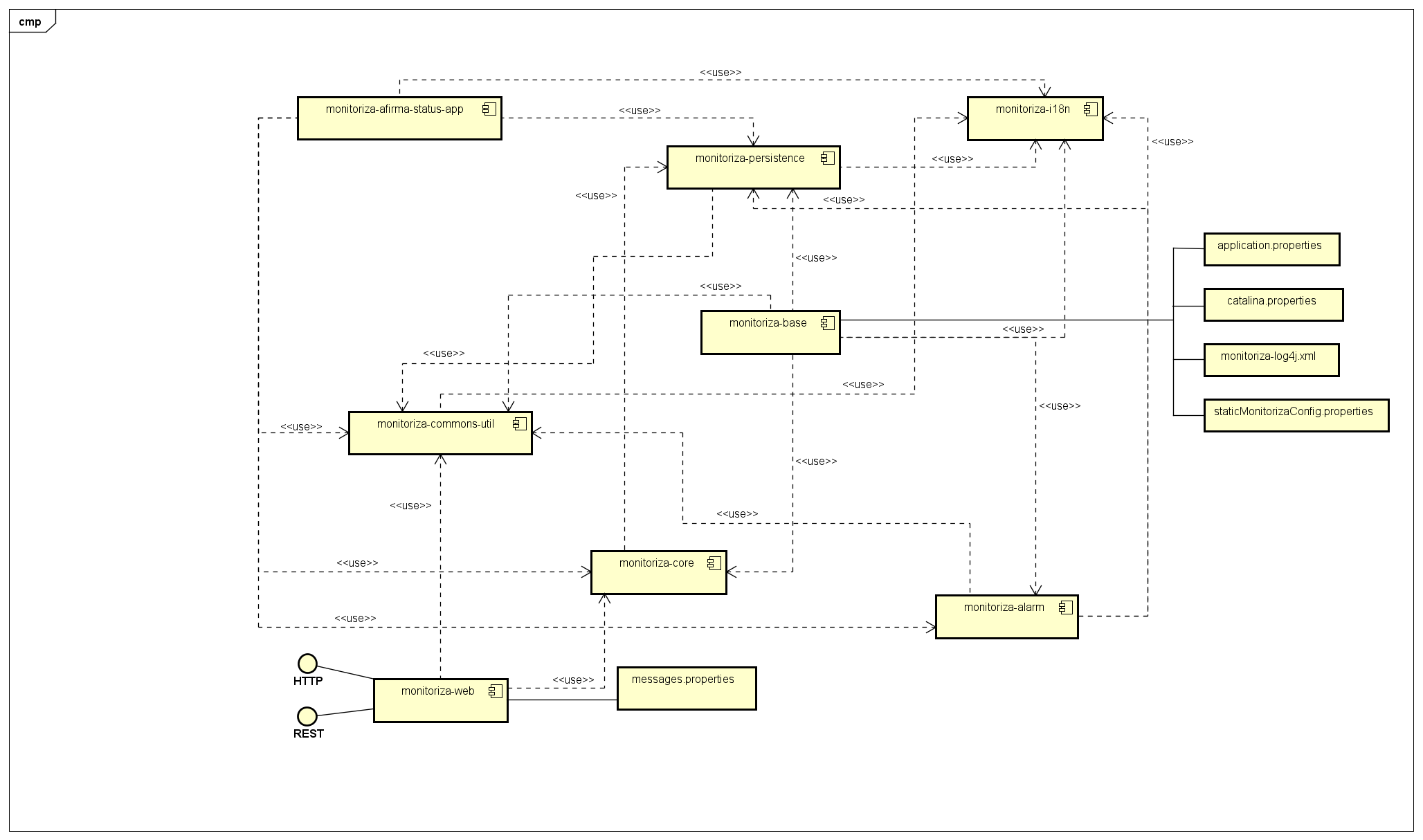


Ilustración 3 Diagrama de componente Monitoriza

### Arquitectura del componente

En los siguiente apartados se describe detalladamente cada uno de los componentes que han sido modificados para la incorporación de los componentes necesarios para el desarrollo necesario. Las modificaciones han sido lo menos intrusivas con el proyecto base de Monitoriz@. Para el resto de componentes se hace referencia al documento de Monitoriz@.

#### Componente monitoriza-afirma-status-app

Se modifica la clase *RequestProcessorThread* añadiendo al método run() una nueva condición en el caso en el que el tipo de servicio sea HTTP.

#### Componente monitoriza-core

Se añade dos nuevos métodos para guardar la nueva plataforma desarrollada en el nuevo complemento:

* **findAllByPlatformType:** Devuelve todas las plataformas según el tipo de plataforma seleccionada.
* **savePlatformClave:** Almacenala configuración de la plataforma de clave en la base de datos.

#### Componente monitoriza-web

Se añaden nuevas condiciones en dos métodos ya creados en el proyecto base de Monitoriz@:

* **loadServiceType:** Se añade una nueva condición para los servicios de tipo HTTP.
* **loadBaseEndpoint:** Se añade una nueva condición para los servicios de tipo HTTP.

Al añadir el nuevo complemento *monitoriza-clave-web*, se modifica la forma de obtener la internalización de las vistas, por lo que se modifica el directorio que tenía el proyecto base de Monitoriz@ y su vez para que esto funcione se añade al *application.properties* la referencia a esa nueva dirección.

Se añade un nuevo enlace en la vista del *sidebar.html* para la gestión de las plataformas de Cl@ve.

#### Componente monitoriza-persistence

Se añade en los scripts de base de datos una nueva inserción para el tipo de servicio para cl@ve.

#### Componente monitoriza-parent

Se modifica el fichero de configuración*catalina.properties* de tomcatse añade la referencia de los nuevos directorios de internalización.

Se añade al repositorio de maven los JAR correspondientes de la herramienta de eIDAS.

## Componente clave-status-app

Este componente Java aporta las clases mínimas para la invocación del método utilizado para los agentes de la monitorización. Cabe destacar las siguientes clases:

* Paquete **es.gob.monitoriza.invoker.http:**
  + Clase **AbstractHttpInvoker:**

Clase abstracta que implementa algunas clases propias del método de invocación de la clase HttpInvoker.

* + Interfaz **Constants:**

Interfaz con todas las constantes utilizadas en el componente.

* + Clase **HttpInvoker:**

Clase extensible de AbstractHttpInvoker y contiene el método principal del envío de la petición.

* Paquete **es.gob.monitoriza.invoker.http.conf.messages:**

Paquete que contiene todas las clases de la conversión de XML a JAXB y viceversa.

* Paquete **es.gob.monitoriza.invoker.http.conf.util**
  + Clase **Utilities**

Clase agrupadora de métodos y utilidades globales en el componente.

* Paquete **es.gob.monitoriza.invoker.http.saml**

Paquete con clases procedentes del proyecto cl@ve y necesarias para la creación del encapsulado SAML.

### Operativa del agente de monitorización

La elaboración de una petición a través de la herramienta Monitoriz@ se enumera de la siguiente forma:

1. Obtención de las configuraciones correspondientes del servicio del agente a monitorizar:
   1. Datos obtenidos de la base de datos.
   2. Datos obtenidos del ZIP descomprimido con información dinámica del agente a monitorizar.
2. Validación por medio de JAXB del fichero XML obtenido del ZIP descomprimido del punto anterior.
3. Generación de la petición SAML con todos los datos recopilados anteriormente.
4. Creación de la petición HTTP.
   1. Esta petición puede ir securizada, para ello se obtendrá del fichero XML la ruta del keyStore para la autenticación mutua.
5. Se añaden la cabecera y el cuerpo en la petición con los atributos estáticos.
6. Envío de la petición según se indica en las configuración obtenida.

### Parametrización peticiones

A continuación se indica el fichero XSD con la normalización que debe tener el fichero XML final.

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <xsd:schema xmlns:xsd=*"http://www.w3.org/2001/XMLSchema"*  xmlns:mcl=*"urn:es:gob:monitoriza:invoker:http:conf:messages:1.0.0"*  targetNamespace=*"urn:es:gob:monitoriza:invoker:http:conf:messages:1.0.0"*  elementFormDefault=*"qualified"* attributeFormDefault=*"unqualified"*>  <xsd:annotation>  <xsd:documentation>  Esquema XML que define la petición a realizar para el servicio de la pasarela de cl@ve.  </xsd:documentation>  </xsd:annotation>  <!-- Configuración de los atributos de conexión de la petición -->    <xsd:complexType name=*"ProxyConfType"*>  <xsd:annotation>  <xsd:documentation>Configuración del proxy  </xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"host"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Identificador del host.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"port"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Identificador del puerto.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"user"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Identificador del usuario.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"password"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Identificador del contraseña.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"AuthenticationMutualType"*>  <xsd:annotation>  <xsd:documentation>Configuración de la autenticación mutua  </xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"path"* type=*"xsd:string"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Dirección del almacenamiento de las claves. Opcional</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"base64"* type=*"xsd:string"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Base64 del almacenamiento de las claves. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"passwordKeyStore"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Contraseña del almacenamiento de las claves.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"typeKeyStore"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Tipo del almacenamiento de las claves.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"ConnectionType"*>  <xsd:annotation>  <xsd:documentation>Configuración de la conexión de la petición. Opcional.  </xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"proxy"* type=*"mcl:ProxyConfType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración del proxy. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"authenticationMutual"* type=*"mcl:AuthenticationMutualType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración de la autenticación mutua. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <!-- Fin configuración de los atributos de conexión de la petición -->    <!-- Configuración de la petición -->    <!-- Enumerados de la petición -->    <xsd:simpleType name=*"SPRequestType"*>  <xsd:annotation>  <xsd:documentation>Tipo de peticion a la SP.</xsd:documentation>  </xsd:annotation>  <xsd:restriction base=*"xsd:string"*>  <xsd:enumeration value=*"public"*/>  <xsd:enumeration value=*"private"*/>  </xsd:restriction>  </xsd:simpleType>    <xsd:simpleType name=*"AttributeType"*>  <xsd:annotation>  <xsd:documentation>Tipo que define los atributos que pueden ser solicitados en una petición SAML.</xsd:documentation>  </xsd:annotation>  <xsd:restriction base=*"xsd:string"*>  <xsd:enumeration value=*"familyName"*/>  <xsd:enumeration value=*"firstName"*/>  <xsd:enumeration value=*"dateOfBirth"*/>  <xsd:enumeration value=*"personIdentifier"*/>  <xsd:enumeration value=*"legalName"*/>  <xsd:enumeration value=*"legalPersonIdentifier"*/>  </xsd:restriction>  </xsd:simpleType>    <!-- Fin enumerados de la petición -->    <!-- Configuración de los parámetros de la petición -->    <xsd:complexType name=*"ParamType"*>  <xsd:annotation>  <xsd:documentation>Tipo que define un parámetro mediante su identificador y su valor.</xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"id"* type=*"xsd:string"* >  <xsd:annotation>  <xsd:documentation>Identificador del parámetro.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"value"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Valor del parámetro.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"ParamsType"*>  <xsd:annotation>  <xsd:documentation>Tipo que define la relación de parámetros.</xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"param"* type=*"mcl:ParamType"* maxOccurs=*"unbounded"*>  <xsd:annotation>  <xsd:documentation>Un parámetros de la relación de parámetros.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"AttributesType"*>  <xsd:annotation>  <xsd:documentation>Tipo que define la relación de atributos de una peticion SAML.</xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"attribute"* type=*"mcl:AttributeType"* maxOccurs=*"unbounded"*>  <xsd:annotation>  <xsd:documentation>Un atributo de la relación de atributos.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"SamlRequestType"*>  <xsd:annotation>  <xsd:documentation>Configuración de la petición Saml  </xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"assertionConsumerServiceURL"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Direccion del servicio de confianza de la petición Saml.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"SPApplication"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Dirección destino de la petición Saml.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"providerName"* type=*"xsd:string"*>  <xsd:annotation>  <xsd:documentation>Nombre del proveedor de la petición Saml.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"SPType"* type=*"mcl:SPRequestType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Tipo del proveedor del servicio de la petición Saml.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"attributes"* type=*"mcl:AttributesType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Listado de atributos.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"HttpRequestType"*>  <xsd:annotation>  <xsd:documentation>Configuración de la conexión de la petición. Opcional.  </xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"headers"* type=*"mcl:ParamsType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración de los otros parámetros. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"params"* type=*"mcl:ParamsType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración de los otros parámetros. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <xsd:complexType name=*"RequestType"*>  <xsd:annotation>  <xsd:documentation>Configuración de la conexión de la petición. Opcional.  </xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"httpRequest"* type=*"mcl:HttpRequestType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración de la peticion http. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"samlRequest"* type=*"mcl:SamlRequestType"*>  <xsd:annotation>  <xsd:documentation>Configuración de la petición Saml encapsulada en la patición HTTP.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>    <!-- Fin configuración de los parámetros de la petición -->    <!-- Fin configuración de la petición -->  <xsd:complexType name=*"ClaveAgentConfType"*>  <xsd:annotation>  <xsd:documentation>Define la configuración para el envio de una petición SAML sin metadatos, mediante HTTP, al sistema de identificación Cl@ve (Pasarela o IdP).</xsd:documentation>  </xsd:annotation>  <xsd:sequence>  <xsd:element name=*"connection"* type=*"mcl:ConnectionType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración de la conexión de la petición. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  <xsd:element name=*"request"* type=*"mcl:RequestType"* minOccurs=*"0"*>  <xsd:annotation>  <xsd:documentation>Configuración de la petición. Opcional.</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:sequence>  </xsd:complexType>  <xsd:element name=*"claveAgentConf"* type=*"mcl:ClaveAgentConfType"*>  <xsd:annotation>  <xsd:documentation>Configuración para el envio de una petición SAML sin metadatos, mediante HTTP, al sistema de identificación Cl@ve (Pasarela o IdP).</xsd:documentation>  </xsd:annotation>  </xsd:element>  </xsd:schema> |

A continuación se indica un tipo de XML válido:

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <mcl:ClaveAgentConf  xmlns:mcl="urn:es:gob:monitoriza:invoker:http:conf:messages:1.0.0"  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  xsi:schemaLocation="urn:es:gob:monitoriza:invoker:http:conf:messages:1.0.0 generationRequestClaveTypes.xsd ">  <mcl:connection>  <mcl:proxy>  <mcl:host>localhost</mcl:host>  <mcl:port>8888</mcl:port>  <mcl:user>usuario</mcl:user>  <mcl:password>12345</mcl:password>  </mcl:proxy>  <mcl:authenticationMutual>  <mcl:passwordKeyStore>G5cp,fYC9gje</mcl:passwordKeyStore>  <mcl:typeKeyStore>jks</mcl:typeKeyStore>  <mcl:path>C:\\Trabajo\\Software\\CertificadosPruebas\\PF\_ACTIVO\_EIDAS.p12</mcl:path>  </mcl:authenticationMutual>  </mcl:connection>  <mcl:request>  <mcl:httpRequest>  <mcl:headers>  <mcl:param>  <mcl:id>parametroH1</mcl:id>  <mcl:value>valorH1</mcl:value>  </mcl:param>  <mcl:param>  <mcl:id>parametroH2</mcl:id>  <mcl:value>valorH2</mcl:value>  </mcl:param>  </mcl:headers>  <mcl:params>  <mcl:param>  <mcl:id>parametro1</mcl:id>  <mcl:value>valor1</mcl:value>  </mcl:param>  <mcl:param>  <mcl:id>parametro2</mcl:id>  <mcl:value>valor2</mcl:value>  </mcl:param>  </mcl:params>  </mcl:httpRequest>  <mcl:samlRequest>  <mcl:assertionConsumerServiceURL>http://localhost:8080/Proxy2/ServiceProvider</mcl:assertionConsumerServiceURL>  <mcl:SPApplication>SPApp</mcl:SPApplication>  <mcl:providerName>DEMO-SP\_NIF</mcl:providerName>  <mcl:SPType>public</mcl:SPType>  <mcl:attributes>  <mcl:attribute>familyName</mcl:attribute>  </mcl:attributes>  </mcl:samlRequest>  </mcl:request>  </mcl:ClaveAgentConf> |

A continuación se explica cada uno de los atributos que contiene el fichero XSD, todas estas clases se ubican en el paquete **es.gob.monitoriza.invoker.http.conf.messages:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nombre** | **Tipo** | **Padre** | **Campo obligatorio** | **Descripción** |
| **claveAgentConf** | String | **-** | Si | Petición hacia la pasarela de cl@ve. |
| **connection** | **ConnectionType** | **claveAgentConf** | No | Configuración de la conexión de la petición. Opcional. |
| **request** | **RequestType** | **claveAgentConf** | No | Configuración de la petición. Opcional. |
|  | | | | |
| **proxy** | **ProxyConfType** | **connection** | No | Configuración del proxy. Opcional. |
| **authenticationMutual** | **authenticationMutualType** | **connection** | No | Configuración de la autenticación mutua. Opcional. |
|  | | | | |
| **host** | String | **proxy** | Si | Identificador del host. |
| **port** | String | **proxy** | Si | Identificador del puerto. |
| **user** | String | **proxy** | Si | Identificador del usuario. |
| **password** | String | **proxy** | Si | Identificador de la contraseña. |
|  | | | | |
| **path** | String | **authenticationMutual** | No | Dirección del almacenamiento de las claves. Opcional. |
| **base64** | String | **authenticationMutual** | No | Base64 del almacenamiento de las claves. Opcional. |
| **passwordKeystore** | String | **authenticationMutual** | Si | Contraseña del almacenamiento de las claves. |
| **typeKeystore** | String | **authenticationMutual** | Si | Tipo del almacenamiento de las claves. |
|  | | | | |
| **httpRequest** | **httpRequestType** | **request** | No | Configuración de la petición http. Opcional. |
| **samlRequest** | **samlRequestType** | **request** | Sí | Configuración de la petición Saml encapsulada en la petición HTTP. |
|  | | | | |
| **headers** | **ParamsType** | **httpRequest** | No | Configuración de los otros parámetros. Opcional. |
| **params** | **ParamsType** | **httpRequest** | No | Configuración de los otros parámetros. Opcional. |
|  | | | | |
| **param** | **paramsType** | **headers** | Si | Un parámetros de la relación de parámetros. |
|  | | | | |
| **id** | String | **param** | Si | Identificador del parámetro. |
| **value** | String | **param** | Si | Valor del parámetro. |
|  | | | | |
| **assertionConsumerServiceURL** | String | **samlRequest** | Si | Dirección del servicio de confianza de la petición Saml. |
| **SPApplication** | String | **samlRequest** | Si | Dirección destino de la petición Saml. |
| **providerName** | String | **samlRequest** | Si | Nombre del proveedor de la petición Saml. |
| **SPType** | **SPRequestType** | **samlRequest** | No | Tipo del proveedor del servicio de la petición Saml. |
| **attributes** | **AttributesType** | **samlRequest** | No | Listado de atributos. |
|  | | | | |
| **attribute** | **AttributeType** | **attributes** | Si | Un atributo de la relación de atributos. |
|  | | | | |
| **SPRequestType** | public, private | | | |
| **AttributeType** | familyName, firstName, dateOfBirth, personIdentifier, legalName, legalPersonIdentifier | | | |

## Componente clave-web

# Base de datos

## Esquema

Usaremos los establecidos en la herramienta Monitoriz@ base.

## Tablas

Usaremos los establecidos en la herramienta Monitoriz@ base.

## Secuencias

Usaremos los establecidos en la herramienta Monitoriz@ base.

## Carga inicial de datos

En este caso debemos añadir a lo ya introducido por la herramienta de Monitoriz@ base, un nuevo tipo de plataforma para su identificación.

# Parámetros del sistema

## Monitoriz@

Los parámetros para configurar y utilizar la herramienta de Monitoriz@:

* Archivo **staticMonitorizaConfig.properties** situado en la configuración del tomcat.

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Descripción** | **Ejemplo** |
| aes.padding.alg | Padding del algoritmo | AES/ECB/PKCS5Padding |
| aes.algorithm | Nombre del algoritmo | AES |
| aes.password | Contraseña del algoritmo | ABCDEFGHIJKLMNOP |
| directories.request.root.path | Ruta raíz donde se situarán los directorios con las peticiones de cada servicio. | C:\\Trabajo\\Software\\apache-tomcat-8.5.24-Monitoriza\\webapps\\ZIPs |
| directories.grupo.principal.path | Ruta donde se situarán las peticiones del grupo principal de un servicio. | \\grupoPrincipal |
| directories.grupo.confirmacion.path | Ruta donde se situarán las peticiones del grupo de confirmación x de un servicio. | \\grupoConfirmacion |
| requestThreadPoolSize | Tamaño del pool de hilos para lanzar las peticiones | 5 |
| alarm.active |  | false |
| confirmation.wait.time | Tiempo de espera para ejecutar siguiente grupo de confirmación (milisegundos) | 60000 |
| character.specia | Keystore | /,^,\*,\=,(,),&,%,$,\u00B7,",\u00A1,\!,[,],{,},\u00E7,\u00C7,\u00BF,?,\u00AC,\# |
| fixedRate.in.milliseconds | Spring scheduled task | 60000 |
| monitoriza.vip.status.servlet | Address for the VIP status servlet | http://localhost:8080/monitoriza-afirma-status-app/afirmaServicesStatus?admin=true |
| monitoriza.spie.status.servlet | Address for the SPIE status servlet | http://localhost:8080/monitoriza-afirma-status-app/spieStatus |
| invoker.http.connect.timeout | Tiempo de conexión del servicio | 30000 |
| invoker.http.connection.request.timeout | Tiempo de conexión de la petición del servicio | 30000 |
| invoker.http.socket.timeout | Tiempo de conexión del socket del servicio | 30000 |

## SAMLEngine

Los parámetros estáticos para configurar las peticiones realizadas por los agentes de monitorización, todos los archivos están situados en el directorio *config*, los cuales se añaden en el despliegue de la herramienta:

* Archivo **EncryptModule\_Monitoriza.xml:**

|  |  |  |
| --- | --- | --- |
| **Parámetro** | **Descripción** | **Ejemplo** |
| check\_certificate\_validity\_period | Validación del periodo del certificado | false |
| disallow\_self\_signed\_certificate | Rechazar certificado autofirmado | false |
| signature.algorithm | Firma del algoritmo | http://www.w3.org/2001/04/xmldsig-more#rsa-sha512 |
| signature.algorithm.whitelist | Lista de algoritmos de firma entrantes de la lista blanca separados por ; | http://www.w3.org/2001/04/xmldsig-more#rsa-sha256; http://www.w3.org/2001/04/xmldsig-more#rsa-sha384; http://www.w3.org/2001/04/xmldsig-more#rsa-sha512; http://www.w3.org/2001/04/xmldsig-more#rsa-ripemd160; http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256; http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha384; http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha512; http://www.w3.org/2007/05/xmldsig-more#sha256-rsa-MGF1 |
| response.sign.assertions | Afirmaciones de signo de respuesta | false |
| keyStorePath | Ruta del almacén de certificados | ../KeyStore.jks |
| keyStorePassword | Contraseña del almacén de certificados | local-demo |
| keyPassword | Clave del almacén de certificados | local-demo |
| issuer | Editor | CN=local-demo-cert, OU=DIGIT, O=European Comission, L=Brussels, ST=Spain, C=ES |
| serialNumber | Número de serie | 58764336 |
| keyStoreType | Tipo del almacén de certificados | JKS |