



SISTEMA INTEGRADO DE GESTIÓN DE EXPEDIENTES MODULAR (SIGM)

ESPECIFICACIÓN SERVICIO TERCEROS EXTERNOS

SIGM v3



Administración Local Soluciones

Control de versiones

Versión	Fecha edición	Cambio producido	Autor
01	05-02-2009	Versión inicial	IECISA
02	Octubre 2012	Actualización para SIGM 3.0	IECISA

ÍNDICE

1	INTRODUCCIÓN	4
1.1	VISIÓN GENERAL DEL SISTEMA	4
1.2	FINALIDAD DEL DOCUMENTO	4
1.3	DEFINICIONES Y ABREVIATURAS.....	4
2	SERVICIO WEB DE SIGM CON TERCEROS DE UN SISTEMA EXTERNO	5
2.1	FUNCIONALIDAD DEL SERVICIO	5
2.1.1	<i>Dar tercero de Alta</i>	5
2.1.2	<i>Modificar tercero</i>	5
2.1.3	<i>Buscar terceros</i>	6
2.1.4	<i>Contar los terceros encontrados</i>	6
2.1.5	<i>Obtener datos tercero</i>	6
2.1.6	<i>Obtener direcciones del tercero</i>	6
2.1.7	<i>Obtener direcciones telemáticas del tercero</i>	7
2.1.8	<i>Obtener lista de provincias</i>	7
2.1.9	<i>Obtener lista de ciudades</i>	7
2.1.10	<i>Clases</i>	7

1 Introducción

1.1 Visión general del sistema

AL SIGM es la plataforma de Tramitación Electrónica del MINETUR, solución integral para la tramitación electrónica de los procedimientos administrativos, que fomenta la interoperabilidad entre administraciones mediante su adaptación a estándares de comunicación así como la reutilización de recursos e información pública.

1.2 Finalidad del documento

En este documento se describe el servicio web de SIGM de acceso al sistema de Terceros de un sistema externo, usado en la Gestión de Expedientes.

Nota: SIGM cuenta con otro servicio de Terceros, propio de SIGEM, que permite su utilización tanto por las aplicaciones de Registro y Gestión de Expedientes, como por cualquier aplicación ajena a SIGM, que lo quiera utilizar.

1.3 Definiciones y Abreviaturas

A continuación se expone una tabla con los diferentes acrónimos y abreviaturas utilizados a lo largo del documento, con su correspondiente definición.

Acrónimo / Abreviatura	Definición
MINETUR	Ministerio de Industria, Energía y Turismo
IECISA	Informática El Corte Inglés S.A.
SIGM	Sistema Integrado de Gestión de Expedientes Modular
AL	Administración Local

2 Servicio web de SIGM con Terceros de un sistema externo

2.1 Funcionalidad del servicio

2.1.1 Dar tercero de Alta

Descripción: da de alta un tercero en la base de datos de Terceros.

public String create (PersonInfo persona, Entidad entidad)

Parámetros:

- PersonInfo persona: objeto con la información de la persona que se quiere dar de alta.
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- String: Identificador del tercero dado de alta.

2.1.2 Modificar tercero

Descripción: modifica los datos de un tercero, en la base de datos de Terceros.

public String update(PersonInfo persona, Entidad entidad)

Parámetros:

- PersonInfo persona: objeto con la información del tercero / persona que se quiere modificar
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- String: Identificador del tercero modificado.

2.1.3 Buscar terceros

Descripción: realiza una búsqueda de terceros en la base de datos de terceros, en base a los criterios de búsqueda indicados.

public Persona[] search (Criterios busqueda, Entidad entidad)

Parámetros:

- Criterios búsqueda: parámetros de búsqueda
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- Persona[]: array de objetos *Tercero* con información de los que cumplen el criterio de búsqueda.

2.1.4 Contar los terceros encontrados

Descripción: realiza una búsqueda de terceros en la base de datos de terceros, en base a los criterios de búsqueda indicados y devuelve el número de terceros que cumplen los criterios indicados.

public Integer count (Criterios busqueda, Entidad entidad)

Parámetros:

- Criterios búsqueda: parámetros de búsqueda
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- Integer: número de terceros encontrado, según el criterio de búsqueda.

2.1.5 Obtener datos tercero

Descripción: obtiene información de un tercero.

public PersonInfo getInfo (Integer id, Entidad entidad)

Parámetros:

- Integer id: identificador del tercero del que se quiere recuperar los datos
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- PersonInfo: información del tercero.

2.1.6 Obtener direcciones del tercero

Descripción: obtiene las direcciones de un determinado tercero.

public Direccion[] getDirecciones (Integer id, Entidad entidad)

Parámetros:

- Integer id: Identificador del tercero del que quieren recuperarse los datos
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- Direccion[]: direcciones del tercero.

2.1.7 Obtener direcciones telemáticas del tercero

Descripción: obtiene las direcciones telemáticas de un determinado tercero.

public EDirecciones[] getEDirecciones (Integer id, Entidad entidad)

Parámetros:

- Integer id: Identificador de la persona de la que quieren recuperarse los datos
- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- EDireccion[]: direcciones telemáticas del tercero que se buscaba.

2.1.8 Obtener lista de provincias

Descripción: obtiene la lista de provincias.

public Provincia[] getProvincias (Entidad entidad)

Parámetros:

- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- Provincia[]: provincias que se buscaban.

2.1.9 Obtener lista de ciudades

Descripción: obtiene la lista de ciudades.

public Ciudad[] getCiudades (Entidad entidad)

Parámetros:

- Entidad entidad: objeto con la información del Ayuntamiento/ Diputación al que pertenece la base de datos.

Retorna:

- Ciudad[]: ciudades que se buscaban.

2.1.10 Clases

PersonInfo. Objeto con la información de una persona

- id: String con el código del usuario
- tipo: String 1(Persona física) o 2 (Persona Jurídica)
- nombre: String Nombre de la persona física o Razón social de la jurídica
- apellido1: String Primer apellido de la persona física
- apellido2: String Segundo apellido de la persona física
- tipoDoc: String Tipo de documentación (N (NIF), P(Pasaporte), E(NIE), X(Otros), C(CIF))
- nif:

- direcciones: Array de objetos Direccion.
- eDirecciones: Array de objetos EDireccion

Direccion. Objeto con los datos de una dirección

- Id: String con el identificador de la dirección
- direccion, ciudad, codPostal, Provincia: Strings
- Preferencia: String con valor "1" si es la dirección principal.

EDireccion. Objeto con los datos de una dirección telemática

- Id: String con el identificador de la dirección
- Dirección: String con todos los datos necesarios para la localización
- Tipo: Tipo de dirección telemática. TE(Teléfono), CE(Correo electrónico), FX(Fax).
- Preferencia: String con valor "1" si es la dirección principal.

Criterios. Criterios de búsqueda

- Criterio []: Array con objetos de tipo criterio
- where: Clausula where adicional
- order: Clausula order.
- personType: Tipo de persona, 1(física), 2(jurídica)
- inicio: Registro inicial a devolver.
- fin: Registro final a devolver.

Criterio. Clausula de búsqueda

- Campo: Campo por el que buscar
- Operador: Operador a aplicar a la busqueda
- Valor: Valor a buscar

Persona. Objeto con la información de una persona

- id: String con el código del usuario
- tipo: String 1(Persona física) o 2 (Persona Jurídica)
- nombre: String Nombre de la persona física o Razón social de la jurídica
- apellido1: String Primer apellido de la persona física
- apellido2: String Segundo apellido de la persona física
- tipoDoc: String Tipo de documentación
- nif:

Provincia. Objeto con la información de una provincia

- id: String con el identificador de la provincia
- codigo: String con el código de la provincia
- nombre: String Nombre de la provincia

Ciudad. Objeto con la información de una ciudad

- id: String con el identificador de la ciudad
- codigo: String con el código de la ciudad
- nombre: String Nombre de la ciudad
- idProvincia: String con el identificador de la provincia a la que pertenece la ciudad.

Entidad. Objeto que almacena la información de un ayuntamiento

- identificador: Código del ayuntamiento
- nombre: Nombre de la población