# ESPECIFICACIONES TÉCNICAS PARA SERVICIOS WEB

Servicio de Catálogo de Cuentas Bancarias

**Historial de Revisiones**

|  |  |  |
| --- | --- | --- |
| Fecha | Versión | Descripción |
| 29/11/2022 | 1.0 | El presente Documento expone los alcances para la generación del servicio web “Servicio de Catálogo de Cuentas Bancarias” necesarios para la sincronización con SIREGOB. |
| 25/01/2023 | 1.1 | Ajustes en la redacción del documento |

**Documento de Entendimiento**

1. **Introducción**

Actualmente se está desarrollando una plataforma para la distribución de los recursos a los municipios en donde se requiere la sincronización con el sistema SIREGOB, para la consulta del para la consulta del Catálogo de Cuentas Bancarias de Proveedor y tener disponible la información más reciente registrada en SIREGOB.

* 1. **Gerencia o Departamento Solicitante**

Coordinación de Gestión de la Información

* 1. **Propósito**

Sincronizar la información de la plataforma de distribución de recursos con el catálogo de cuentas bancarios de proveedor de SIREGOB.

* 1. **Definiciones, Acrónimos, y Abreviaturas**

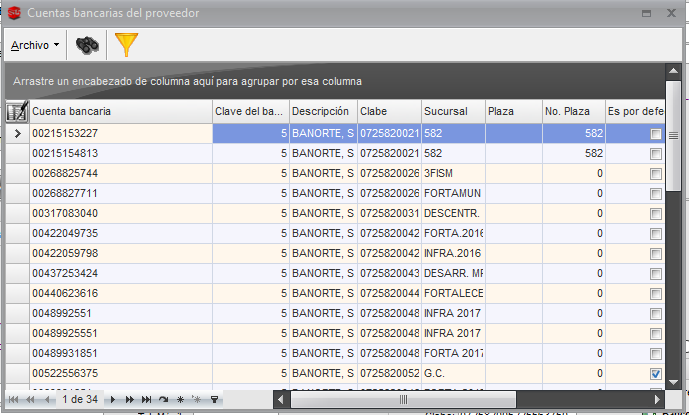
N/A: No aplica.

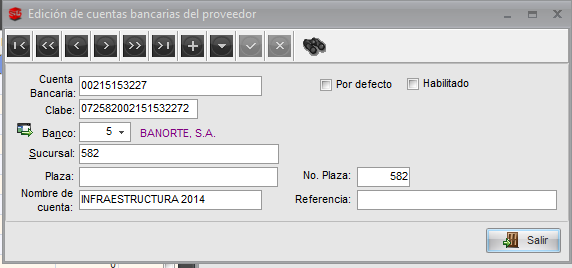
U. Resp: Unidad Responsable

1. **Descripción del Requerimiento**

Publicar Servicio Rest/API para lograr la sincronización del sistema SIREGOB, con la plataforma de Distribución de Recursos.

Cuando el usuario realice las operaciones de alta, modificación, en la plataforma de distribución, estas se sincronicen en el sistema SIREGOB.





* 1. **Protocolo Servicio Web**

se propone utilizar el protocolo de transferencia de datos REST (Representational State Transfer)

* 1. **Método de Servicio**

Para el uso del servicio debe ser mediante el método POST, ya que es el más recomendado para la trasferencia de información

* 1. **Formato de Salida**

Formato: JSON

Tipo MIME: application/json

* 1. **Parámetros de entrada**

A continuación, se describen los parámetros de entrada que debe considerar el servicio

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CAMPO | TIPO | OBLIGATORIO | DESCRIPCION | EJEMPLO |
| Usuario | Varchar(20) | Si | Describe el usuario con el que se está realizando la consulta | VCARDENAZ |
| U.Resp | Varchar(4) | Si | Describe la unidad Responsable que tiene relacionada el usuario | 0405 |
| Tipo de Operación | Int(1) | SI | Decide el tipo de operación que se realizara | 1. = inserción, 2. = modificación 3. = consulta |
| Clave de proveedor | Varchar(10) | SI | Describe la clave del proveedor | 000499 |
| Cuenta bancaria | Varchar(18) | SI | Describe el número de cuenta bancaria | 00215153227 |
| Clabe | Varchar(18) | SI | Describe la clave bancaria | 072582002151532272 |
| Banco | Int(5) | SI | Clave del Banco | 5 |
| Sucursal | Varchar(10) | NO | Describe la sucursal bancaria | 582 |
| Nombre de la cuenta | Varchar(100) | NO | Describe el identificador de la cuenta | INFRAESTRUCTURA 2014 |
| Por Defecto | Boolean | NO | Marca la cuenta por defecto | True / false |
| Habilitado | Boolean | NO | Marca la cuenta como habilitada | True / false |
|  |  |  |  |  |

**Ejemplo de parámetros en la solicitud**

**[{**

**Usuario:VCARDENAZ,**

**UResp:0405,**

**TipodeOperacion:1,**

**Clavedeproveedor:000499,**

**Cuentabancaria:00215153227,**

**Clabe:072582002151532272,**

**Banco:5,**

**Sucursal:582,**

**Nombredelacuenta:'INFRAESTRUCTURA 2014',**

**PorDefecto:false,**

**Habilitado:false**

**}]**

* 1. **Parámetros de Salida**

A continuación, la Salida estándar del Servicio Web,

Siempre regresando una estructura de la siguiente forma:

**ESTATUS**: debe regresar el número de estatus de la petición http tomando el siguiente criterio

200: si la petición fue un éxito

400: si existe error en conexión con el cliente

500: si existe error en conexión con el servidor

550: si existe un error que impida completar la operación, (falta de permisos, falta de datos, error en la operación)

**MENSAJE**: debe regresar el mensaje de la solicitud, por ejemplo, si hubo un error, si fue un éxito, o si existe algún mensaje que no permitió completar la solicitud.

**DATA**: debe regresar los datos de la cuenta afectada en forma de objeto en caso de error debe regresar el número y mensaje de error.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CAMPO | TIPO | OBLIGATORIO | DESCRIPCION | EJEMPLO |
| Estatus | Numérico | Si | Indica el estatus de la solicitud | 200: la acción se realizó correctamente (Succesful)  300: se ha tenido que tomar una acción adiconal(Redirección)  400: (Errores del cliente)  500: (Errores del servidor) |
| Mensaje | Varchar(2000) | Si | Indica el mensaje de respuesta de la solicitud | “Solicitud Exitosa” /  “Usuario no tiene permisos” /  “Error …”  Etc.. |
| Data | Json Object | SI | Indica la Respuesta de la petición , esta puede ser una lista de objetos en formato json | [{  Usuario:VCARDENAZ,  UResp:0405,  TipodeOperacion:1,  Clavedeproveedor:000499,  Cuentabancaria:00215153227,  Clabe:072582002151532272,  Banco:5,  Sucursal:582,  Nombredelacuenta:'INFRAESTRUCTURA 2014',  PorDefecto:false,  Habilitado:false  }]  O en caso de error  [{  Numero de error:56685,  Mensaje de error: ”No existe …”  }] |

1. **Limitaciones**

Se debe definir la forma de autenticación para poder consumir el servicio

1. **Glosario**

**Parámetros**: Los parámetros de una función son los valores que esta recibe por parte del código que la llama. Pueden ser tipos simples u objetos.

**Protocolo**: En redes, un protocolo de comunicaciones o protocolo de red es la especificación de una serie de reglas para un tipo particular de comunicación. La red Internet se basa en el modelo de referencia TCP/IP (Transmission Control Protocol/Internet Protocol) que toma su nombre de los dos principales protocolos que regulan la comunicación a través de esta red.

**Proxies**: Un proxy (representante) es un agente o sustituto autorizado para actuar en nombre de otra máquina que lo autoriza a hacerlo y puede utilizarse en el contexto de servidor proxy, que hace de intermediario en las peticiones de recursos que realiza un cliente (A) a otro servidor (C).

**REST:** Representational State Transfer - Arquitectura que, haciendo uso del protocolo HTTP, proporciona una API que utiliza cada uno de sus métodos (GET, POST, PUT, DELETE) para poder realizar diferentes operaciones entre la aplicación que ofrece el servicio web y el cliente.

**Servicio Web:** Conjunto de aplicaciones o de tecnologías con capacidad para interoperar en la Web. Estas aplicaciones o tecnologías intercambian datos entre sí con el objetivo de ofrecer unos servicios. Los proveedores ofrecen sus servicios como procedimientos remotos y los usuarios solicitan un servicio llamando a estos procedimientos a través de la Web.

**SOAP**: Protocolo Simple de Acceso a Objetos, el cual es un protocolo basado en XML, que permite la interacción entre varios dispositivos y que tiene la capacidad de transmitir información compleja.