Estado Plurinacional de Bolivia

IOP-001

Lineamientos para la implementación de servicios de interoperabilidad para las entidades del sector público

ctic CONSEJO PARA LAS TECNOLOGÍAS DE INFORMACIÓN Y COMUNICACIÓN

Lineamientos para la implementación de servicios de interoperabilidad para las entidades del sector público

Lineamientos para la implementación de servicios de interoperabilidad para las entidades del sector público

IOP-001

Este documento ha sido elaborado por los miembros del Consejo para las Tecnologías de Información y Comunicación del Estado Plurinacional de Bolivia (CTIC-EPB).

Coordinación Secretaria Técnica del CTIC - EPB: Cristina Loma y Carolina Ovale.

Alan Sandoval Pereira, Alberto Guillermo Arnez Flores, Alejandro Salamanca, Alejandro Gozalves, Alejandro Javier Sánchez Roa, Alicia Estrada Cava, Álvaro Darío Murillo Flores, Ana María Durán, Armin Mesa Sánchez, Arsenio Marcial Castellón Quisbert, Arturo Farfan, Bernardo Valdivia Baldomar, Branko Matijasevic, Cándido Cabrera Jiménez, Carlos Macuchapi Parisaca, Carlos Alberto Guisbert Salazar, Carlos Arratia Paz, César Álvarez Antezana, Christian Cuéllar Ramírez, Claudio Cuellar, Corali Betty Condori Bernabé, Cynthia Helen Rodríguez Conde, David Esaú Dávila Chapana, Dennis Saravia Beltrán, Diego Cristian Melendres Argote, Dulfred Gutiérrez, Edgar Monrroy, Edson Vallejos Pacheco, Edwin Zarate Aduviri, Eitner Montero Churata, Elizabeth Garcia, Danny Elvis Quenta Apaza, Ervin Mario Flores Fernández, Esteban Calisaya, Fabricio Góngora, Fernando Fernández, Fernando Fuentes, Franco Morales Valdivia, Franz Kenny Quintanilla Arancibia, Franz Rojas Castillo, Freddy Mendoza, Freddy Acarapi, Gabriel Vallejos, Gary Ramos Cuela, George Cervantes, Gerardo Luna Arguata, Germán Molina, Germán Quisbert, Gladys Alanoca, Gonzalo Medina, Gregory Pekynov Bustamante Rojas, Helmuth Alberto Pardo Salinas, Horacio López Justiniano, Huberth Vargas, Hugo Gutiérrez Espada, Hugo Zubieta Iriarte, Israel Jiménez Antezana, Ivan Gabriel Espinoza Marin, Ivan Rojas Vega, Jacqueline Danitza Sánchez, Jan Pérez, María Jannett Ibañez Flores, Javier Antonio Arduz López, Jenny Castro Quenta, Jhery Sanjinez, Jhonny Ruben Monrroy Casillo, Joaquín Heredia Molina, Jorge Cox, Jorge Pablo Tordoya Rojas, José Cortez, José Luis Aruquipa Hilari, José Luis Baldivieso Portela, José Luis Batuani Rivamontán, José Luis Escarcha, José Miguel Pacoricona, Juan Carlos Chejo, Juan Fernando Yañez Bernal, Juan Marcelo Rocabado Alanez, Julia Luz Tarifa, Laura Quisbert Bustamante, Lelis Jenny Padilla Vedia, Lizeth Mendoza, Luis Rejas Alurralde, Luis Miguel Quispe Nina, Luis Villanueva Cabrera, Marcelino René Ergueta Illanes, Marcelo Pinto Macedo, Marcelo Romero, Marco Antonio Laura Avendaño, María Arismendi, María del Pilar Mamani, María Elizabeth Quispe, María Estefani Huanca Huanca, Mario Rodolfo Silva Cabrera, Marlene Ordoñez, Gonzalo Carvajal Sumi, Martín Meruvia, Mauricio Bellota, Mayra Sotes, Melvy Rodríguez, Michael Castillo Campo, Miguel Medina, Miguel Franz Jallaza Castro, Milenka Altamirano Ferreira, Nelson Huanca Pinto, Oliver Camacho Giacoman, Oscar Murillo Cardozo, Oscar Durán, Paola Calderón Escalante, Patricia Noemí Guisbert Sánchez, Pedro Zambrana Rivera, Pedro Damian Vásquez Calle, Peter Rodríguez Zapata, Rainer Gutiérrez, Ramiro Ariel Bellido Carranza, Ramiro Jesús Vásquez Quiñones, Ramiro Oña, Raúl Fernando Molina Rodríguez, Raúl García, Renán Luis Layme Yucra, René Erick Mollinedo Silva, Reynaldo Rodrigo Choque Vicente, Roberto Santivañez Loayza, Rocío Campos, Roger Chuquimia Mamani, Rómulo Calderón, Ronny Marcelo Espinoza Arias, Rosa Ramos Monasterios, Rosmery Ana Zegarra Deheza, Rosse Mary Gonzales, Sergio Sánchez, Sergio Martínez, Shirley Salazar Montoya, Skarleth Saavedra, Stephanie Ferreira Paravicini, Teodoro Nina, Virginia Kama, Vladimir Yugar Pinto, Wilber López Castro, Wilfredo Alarcón, Wilfredo Callisaya, Wilfredo Nacho, Willyams Yujra, Ximena Soto Salvador, Sylvain Damien Lesage, Sergio A. Criales Yujra, Ariel Alvarado Atahuichi, Ever Favio Argollo Ticona.

Edición, diseño y diagramación: Natalia Antezana y Orestes Sotomayor.

Depósito Legal: 4-1-426-17 P.O.

La Paz Bolivia

2017


ctic
CONSEJO PARA LAS
TECNOLOGÍAS DE INFORMACIÓN
Y COMUNICACIÓN

Contenido

1 INTRODUCCIÓN

El Consejo para las Tecnologías de Información y Comunicación del Estado Plurinacional de Bolivia (CTIC-EPB) se constituye en una instancia de coordinación técnica para la implementación de Gobierno Electrónico y para el uso y desarrollo de Tecnologías de Información y Comunicación en el país.

Entre las principales tareas asignadas al CTIC-EPB se encuentran:

- Formular propuestas de políticas y normativa relacionada con Gobierno Electrónico, a ser presentadas a la AGETIC;
- Presentar proyectos y programas de Gobierno Electrónico y Tecnologías de Información y Comunicación en el ámbito gubernamental a la AGETIC para su gestión;
- Generar mecanismos de participación de instituciones y organizaciones de la sociedad civil en la proposición y formulación de políticas y acciones relacionadas con Gobierno Electrónico y Tecnologías de Información y Comunicación en el ámbito gubernamental;
- Establecer espacios de coordinación entre las entidades del sector público para el desarrollo conjunto de programas, proyectos o acciones de Gobierno Electrónico y Tecnologías de Información y Comunicación en el ámbito gubernamental;
- Desarrollar y proponer estándares abiertos oficiales del Estado Plurinacional de Bolivia en materia de Gobierno Electrónico y Tecnologías de Información y Comunicación aplicables a las entidades del sector público;
- Establecer espacios de coordinación de comunidades de desarrollo informático, dentro del Estado, con la ciudadanía y a nivel internacional.

El 5 de mayo de 2016 se llevó a cabo la inauguración y la primera Reunión del Pleno del CTIC-EPB, en la que se conformaron seis grupos temáticos de trabajo: Interoperabilidad, Software Libre, Seguridad, Infraestructura, Desarrollo de Software y Datos Abiertos.

Cada Grupo de Trabajo estuvo integrado por servidores públicos de las entida-

des del nivel central del Estado: Órgano Ejecutivo, Legislativo, Judicial y Electoral, incluyendo sus instituciones descentralizadas, autárquicas, empresas públicas y autoridades de regulación sectorial; Ministerio Público y Procuraduría General del Estado.

Adicionalmente, se invitó a participar, en calidad de miembros adjuntos, a representantes de entidades territoriales autónomas, universidades públicas e indígenas y sociedad civil, a fin de trabajar y elaborar propuestas a ser presentadas al Consejo para su posible implementación a nivel estatal.

Cabe mencionar que el desarrollo de los Grupos de Trabajo y del Consejo se enmarca en el Reglamento de Funcionamiento del CTIC-EPB, aprobado mediante la Resolución Administrativa N° 024/2016 de la AGETIC, de fecha 31 de mayo de 2016.

El Grupo de Trabajo de Interoperabilidad se planteó como objetivo la formulación de los lineamientos para que las entidades o instituciones públicas del Estado Plurinacional de Bolivia, puedan elaborar e implementar servicios de interoperabilidad.

El Grupo estuvo conformado por los representantes de las siguientes entidades:

- Administración de Aeropuertos y Servicios Auxiliares a la Navegación Aérea (AASANA).
- Aduana Nacional de Bolivia (ANB).
- Agencia Nacional de Hidrocarburos (ANH).
- Agencia para el Desarrollo de la Sociedad de la Información en Bolivia (ADSIB).
- Agencia de Gobierno Electrónico y Tecnologías de Información y Comunicación (AGETIC).
- Autoridad de Fiscalización y Control de Pensiones y Seguros (APS).
- Autoridad de Fiscalización y Control Social de Electricidad (AE).
- Autoridad de Impugnación Tributaria (AIT).
- Autoridad de Regulación y Fiscalización de Telecomunicaciones y Transportes (ATT).

- Autoridad de Supervisión del Sistema Financiero (ASFI).
- Autoridad Plurinacional de la Madre Tierra (APMT).
- Banco Central de Bolivia (BCB).
- Banco Unión S.A. (BUSA).
- Consejo de la Magistratura.
- Dirección del Notariado Plurinacional (DIRNOPLU).
- Empresa de Apoyo a la Producción de los Alimentos (EMAPA).
- Empresa de Correos de Bolivia (ECOBOL).
- Empresa Pública QUIPUS.
- Escuela de Gestión Pública Plurinacional (EGPP).
- Gobernación del Departamento de Santa Cruz.
- Instituto Nacional de Estadística (INE).
- Instituto Nacional de Reforma Agraria (INRA).
- Ministerio de Defensa.
- Ministerio de Desarrollo Productivo y Economía Plural (MDPyEP).
- Ministerio de Economía y Finanzas Públicas (MEFP).
- Ministerio de Educación.
- Ministerio de Gobierno.
- Ministerio de Justicia y Transparencia Institucional.
- Ministerio de Obras Públicas Servicios y Vivienda (MOPSyV).
- Ministerio de Salud.
- Ministerio de Trabajo, Empleo y Previsión Social (MTEyPS).

- Servicios de Desarrollo de las Empresas Públicas (SEDEM).
- Servicio General de Identificación Personal (SEGIP).
- Servicios de Impuestos Nacionales (SIN).
- Sistema Nacional de Información en Salud y Vigilancia Epidemiológica del Ministerio de Salud (SNIS-VE).
- Servicio Nacional de Verificación de Exportaciones (SENAVEX).
- Servicios Geológico Minero (SERGEOMIN).
- Tribunal Supremo Electoral (TSE) - Servicio de Registro Cívico (SERECI).
- Unidad de Análisis de Políticas Sociales Económicas (UDAPE).
- Unidad de Proyectos Especiales (UPRE).
- Universidad Mayor de San Andrés (UMSA).
- Vicepresidencia del Estado Plurinacional – Geobolivia.
- Yacimientos Petrolíferos Fiscales Bolivianos (YPFB).
- Alicia Heydy Estrada Cava (Comunidad Software Libre Bolivia).
- Alejandro Gozálvez (Sociedad civil).
- Alejandro Salamanca (Comunidad Software Libre Bolivia).
- Gonzalo Carvajal (Más y Mejor Internet para Bolivia).
- Luis Rejas (Más y Mejor Internet para Bolivia).
- Virginia Kama (Sociedad civil).

Asimismo, es importante resaltar que otras entidades u órganos del Estado participaron a través de sugerencias y acotaciones al documento inicial elaborado por el Grupo. Entre estas entidades se encuentran:

- Dirección General de Migración (DIGEMIG).

- Empresa Estatal de Transporte por Cable “Mi Teleférico”.
- Escuela Militar de Ingeniería (EMI).
- Fondo de Desarrollo del Sistema Financiero y de Apoyo al Sector Productivo (FONDESIF).
- Programa Bono Juana Azurduy (BJA) del Ministerio de Salud.
- Servicio Nacional de Registro y Control de la Comercialización de Minerales y Metales (SENARECOM).
- Servicio Nacional de Sanidad Agropecuaria e Inocuidad Alimentaria (SENASAG).
- Instituto Boliviano de Metrología (IBMETRO).
- Registro Único para la Administración Tributaria Municipal (RUAT).

Como resultado de las reuniones de trabajo del Grupo de Interoperabilidad, sus respectivas discusiones y exposiciones, se elaboró el documento “Lineamientos para la implementación de servicios de interoperabilidad para las entidades del sector público”, del Estado Plurinacional de Bolivia.

2 MARCO NORMATIVO REFERENCIAL

El marco normativo concerniente a la temática incluye:

- El Parágrafo I del Artículo 103 de la Constitución Política del Estado, que establece que: " el Estado garantizará el desarrollo de la ciencia y la investigación científica, técnica y tecnológica en beneficio del interés general. Se destinarán los recursos necesarios y se creará el sistema estatal de ciencia y tecnología".

- El Parágrafo II del mencionado Artículo, establece que el Estado asumirá como política la implementación de estrategias para incorporar el conocimiento y aplicación de nuevas tecnologías de información y comunicación.

- El Artículo 85 de la Ley Nº 031, de 19 de julio de 2010, Marco de Autonomías y Descentralización "Andrés Ibáñez", dentro las competencias exclusivas del nivel central del Estado, determina lo siguiente: "I. Formular y aprobar el régimen general y las políticas de comunicaciones y telecomunicaciones del país, incluyendo las frecuencias electromagnéticas, los servicios de telefonía fija y móvil, radiodifusión, acceso al internet y demás Tecnologías de Información y Comunicaciones (TIC)" (...).

- El Artículo 72 (Rol del Estado) de la Ley Nº 164, de 8 de agosto de 2011, Ley General de Telecomunicaciones, Tecnologías de Información y Comunicación, señala en el parágrafo II: "Las entidades públicas deberán adoptar todas las medidas necesarias para garantizar el máximo aprovechamiento de las tecnologías de información y comunicación en el desarrollo de sus funciones".

- El parágrafo III señala que el Estado promoverá de manera prioritaria el desarrollo de contenidos, aplicaciones y servicios de las tecnologías de información y comunicación en las siguientes áreas:

    - ◊ En educación, como medio para la creación y difusión de los saberes de las bolivianas y los bolivianos en forma universal y equitativa.

    - ◊ En salud, como mecanismo para desarrollar el sistema de alerta temprana, bases de administración de recursos en salud y plataformas de acceso a la información y consultas del sector.

- ◊ En gestión gubernamental, como mecanismo para optimizar los sistemas existentes y crear nuevos para atender la demanda social, facilitar el acceso y uso intensivo de estos sistemas a nivel interno de cada unidad gubernamental, entre entidades gubernamentales, entre las ciudadanas y ciudadanos con las entidades gubernamentales.

- ◊ En lo productivo, como mecanismo para optimizar, hacer eficiente y reducir los costos de la economía plural debiendo desarrollarse aplicaciones de tecnologías de la información y comunicación.

- ◊ En comunicación e información, como mecanismo que permita garantizar los derechos a la libre expresión, a la diversidad de la palabra y a la participación activa, plural e informada de las bolivianas y los bolivianos.

- El Artículo 7 (Regulación de Excepciones) del Decreto Supremo N° 28168, de fecha 17 de mayo de 2005, señala: "I. El acceso a la información sólo podrá ser negado de manera excepcional y motivada, únicamente respecto a aquella información que con anterioridad a la petición y de conformidad a leyes vigentes se encuentre clasificada como secreta, reservada o confidencial. Esta calificación no será, en ningún caso, discrecional de la autoridad pública".

- El Artículo 18 (Lineamientos del Plan de Implementación) del Decreto Supremo N° 1793, de 13 de noviembre de 2013, Reglamento de la Ley N° 164, de 8 de abril de 2011, Ley Telecomunicaciones, Tecnologías de Información y Comunicación, señala en su inciso "d) Proponer mecanismos para lograr eficiencia en el uso de los recursos tecnológicos de las entidades públicas, además de la interoperabilidad de los sistemas de información y de servicios gubernamentales desarrollados por cada una de ellas, a través de la aplicación y uso de estándares abiertos; " (...).

- El Artículo 19 (Interoperabilidad, Datos e Información) del Decreto Supremo N° 2514, de 9 de septiembre de 2015, establece lo siguiente:

  - ◊ La AGETIC coordinará con las entidades del sector público la implementación de servicios de interoperabilidad de Gobierno Electrónico así como los datos e información que deben estar disponibles.

- ◊ Se autoriza a las entidades públicas proporcionar a la AGETIC los datos e información que hubieran producido, recolectado o generado, por medios electrónicos o mecanismos de interoperabilidad, que ésta solicite mediante nota formal de su MAE, en el marco de la política general de Gobierno Electrónico, simplificación de trámites, transparencia, participación y control social y tecnologías de la información y comunicación.

- ◊ El ente rector de Gobierno Electrónico determinará la política general y normativa específica de interoperabilidad e intercambio de información y datos entre las entidades del sector público.

- El Decreto Supremo N° 3251, de 12 de julio de 2017, que aprueba el Plan de Implementación de Gobierno Electrónico y el Plan de implementación de Software Libre y Estándares Abiertos.

- Asimismo, el Parágrafo I del Artículo 4 del mencionado Decreto Supremo señala: "El COPLUTIC en coordinación con la Agencia de Gobierno Electrónico y Tecnologías de Información y Comunicación – AGETIC, podrá determinar la obligatoriedad por parte de las entidades públicas para compartir información mediante interoperabilidad, en el marco de las leyes y normas vigentes, así como disposiciones específicas de sectores estratégicos".

3 OBJETIVO

El presente documento tiene como objetivo establecer los lineamientos para la implementación de servicios de interoperabilidad por parte de las entidades del sector público del Estado Plurinacional de Bolivia.

4 ALCANCE Y ÁMBITO DE APLICACIÓN

En este documento se formulan los lineamientos para estandarizar la implementación de servicios de interoperabilidad en las entidades del sector público de nivel central, instituciones descentralizadas, instituciones desconcentradas, autárquicas, empresas públicas estratégicas y mixtas, autoridades de regulación sectorial, Ministerio Público y Procuraduría General del Estado.

En el caso de entidades privadas que brinden servicios de interoperabilidad al sector público, en el marco de relaciones contractuales o de concesiones de servicios y bienes, las entidades del sector público deberán solicitar la aplicación y cumplimiento de los puntos 7 (Servicios Web), 8 (Seguridad), 9 (Disponibilidad), 10 (Accesibilidad) y 11 (Políticas) mínimamente.

Los lineamientos contenidos en este documento establecen estándares técnicos mínimos a ser implementados por todas las entidades del sector público, sin perjuicio del trabajo desarrollado por aquellas que ya hayan asumido como parámetros rectores, normas y estándares nacionales e internacionales vigentes o de otra naturaleza en materia de interoperabildad.

Tomando en cuenta la naturaleza dinámica de la tecnología, las entidades podrán implementar versiones superiores y/o con mejores características, sin dejar de considerar todos los puntos que se detallan en el documento.

La AGETIC analizará periódicamente la necesidad de actualización de este documento para su consideración en el CTIC-EPB.

5 NATURALEZA DEL SERVICIO DE INTEROPERABILIDAD

Cada servicio de interoperabilidad presenta ciertas características que definen su naturaleza de acuerdo al tipo de datos que proporcionan. En este marco, se recomienda que la entidad que desee implementar un servicio de interoperabilidad identifique primero el tipo de servicio que proveerá con la finalidad de conocer el comportamiento que este tendrá sobre los datos[1]. Los tipos de servicios de interoperabilidad son de lectura o transaccional, individual o masivo, y agregado estadístico o desagregado[2], los cuales se describen a continuación:

5.1 Tipo de servicio de lectura/consulta o transaccional

Cuando el servicio de interoperabilidad permite solamente la lectura de los datos y no realiza modificación de los mismos en la fuente primaria, el tipo de servicio de interoperabilidad es sólo de lectura. Un tipo de servicio de interoperabilidad de consulta puede incluir el procesamiento y la formación de una respuesta dedicada.

Cuando el servicio de interoperabilidad permite la creación, actualización o eliminación de los datos, el tipo de servicio es transaccional.

Es posible que un servicio de interoperabilidad de tipo transaccional también incluya la lectura/consulta.

Es importante mencionar que en el caso de identificar un servicio de tipo transaccional se aplicarán mayores medidas de seguridad en su implementación, como se verá más adelante en el punto 8.2 (Seguridad de los datos).

5.2 Tipo de servicio individual o masivo

Cuando el servicio de interoperabilidad devuelve solamente un conjunto de datos relacionados y estructurados entre sí (por ejemplo, los datos de una persona como su nombre, número de cédula de identidad, fecha de nacimiento), el tipo del servicio de interoperabilidad es individual.

En cambio, cuando el servicio de interoperabilidad devuelve un listado de un conjunto de datos relacionados y estructurados, se considera a este tipo de servicio

[1] Sobre los datos, a su vez, se aplican las operaciones de creación, lectura, modificación y eliminación o CRUD (Create, Read, Update and Delete).
[2] Un servicio de interoperabilidad puede presentar todas las características de los tipos mencionados.

como masivo (por ejemplo, un listado de personas).

Es importante mencionar que en el caso de implementarse un servicio de interoperabilidad de tipo masivo, se sugiere aplicar los criterios de paginación, de filtro y búsqueda y otros de disponibilidad sobre el servicio de interoperabilidad, debido a la gran cantidad de datos que serán solicitados por los consumidores. Este tema será descrito con mayor detalle en el punto 9.3 (Balanceo de carga).

5.3 Tipo de servicio agregado estadístico o desagregado

Un servicio agregado es aquel que devuelve los datos como información procesada y con estadísticas (por ejemplo, promedios, sumas u otra forma de agrupación); de esta forma, el consumidor no necesita procesarlos.

Un servicio desagregado, en cambio, expone los datos sin ningún tipo de procesamiento y de manera individual.

En caso de tratarse de un tipo de servicio agregado estadístico, se toma en cuenta la cantidad de datos a ser procesados, ya que si se trata de una cantidad grande de datos y el cálculo se realiza en ese instante, se sugiere aplicar los criterios de disponibilidad. Para mayor detalle véase el punto 9.4 (Cacheado).

6 SEMÁNTICA

La interoperabilidad requiere que todos los participantes hablen y apliquen un lenguaje común para intercambiar los datos, de modo que estos se entiendan de la misma manera.

Es necesario definir las características de los datos que se desean intercambiar mediante el servicio de interoperabilidad:

6.1 Objeto

Un objeto es la abstracción de un elemento físico o conceptual del que pueden identificarse los atributos, los metadatos de esos atributos, las relaciones y sus ámbitos (los contextos en que tiene sentido este objeto).

Los objetos se definirán de manera consensuada entre las entidades del Estado Plurinacional de Bolivia con el fin de otorgarles un mismo significado.

6.2 Código

Para poder identificar a los objetos de una misma manera se necesita definir un código de identificación único y consensuado o utilizar los códigos ya definidos, si existieran.

En caso de crearse un código nuevo, se necesita definir tanto su estructura como sus características.

La finalidad de este código es la identificación unívoca de un objeto que permite relacionarlo con otro para posibilitar el cruce de información entre bases de datos.

La fuente primaria es la encargada de establecer el valor a este código para cada objeto sobre el que tiene tuición.

6.3 Recomendaciones

Todos los objetos consensuados se documentarán, pudiendo ser publicados en un catálogo de objetos de uso general para las entidades del Estado Plurinacional de Bolivia.

Se dará prioridad a los datos que se producen en la entidad que se considera fuen-

te primaria, ya que es una mala práctica recopilar datos de segunda mano.

Los cambios o adiciones en la semántica deben ser reportados a las entidades consumidoras y debidamente documentados.

Los datos que se publiquen a través de un servicio de interoperabilidad deberán utilizar la semántica consensuada en el Estado, permitiendo tanto la comprensión de los datos por las distintas entidades como la posibilidad de cruzar datos de diversas fuentes.

7 SERVICIOS WEB

Se recomienda utilizar los siguientes lineamientos al implementar servicios de interoperabilidad.

7.1 Definición de parámetros de entrada/salida

Es recomendable que la entidad establezca los parámetros de entrada y salida del servicio de acuerdo a su naturaleza. Cada parámetro será definido claramente y describirá su propósito; esta definición facilitará, posteriormente, la generación de la documentación respectiva.

7.2 Buenas prácticas

Al momento de implementar un servicio de interoperabilidad es importante tomar en cuenta experiencias de implementación con resultados positivos que ayuden a mejorar o solucionar problemas al poner en funcionamiento los servicios. La sistematización de estas experiencias es lo que se denomina una buena práctica.

Se recomienda ser consistentes en la implementación de servicios de interoperabilidad, manteniendo las buenas prácticas en todos los servicios de acuerdo a la tecnología utilizada.

7.3 Validación de los mensajes

La validación de los mensajes se realiza para verificar que la estructura de los objetos que se utilizan en el servicio de interoperabilidad sea la correcta.

Validar los mensajes evita problemas de funcionamiento no previstos cuando los clientes consumen el servicio.

Se recomienda validar la estructura de los mensajes, tanto de los parámetros de entrada como los de salida.

7.4 Versionamiento

En ocasiones es necesario tener varias versiones de un mismo servicio. Esto puede ocurrir cuando el servicio presenta nuevas características de funcionamiento (o en el caso de necesitar otros parámetros de entrada/salida).

Se recomienda siempre versionar los servicios de interoperabilidad para evitar inconvenientes a los consumidores que ya usan el servicio. El versionamiento debe considerarse desde el inicio de la implementación.

Los cambios o modificaciones entre versiones se documentan y se publican de acuerdo a lo establecido en el punto 10.3 (Documentación).

7.5 Manejo de errores

Todos los problemas que presenten los servicios de interoperabilidad deben ser identificados, de modo que la entidad consumidora de los mismos interprete los motivos de las fallas.

Se recomienda que el mensaje de error tenga como mínimo un código y una descripción que permita comprenderlo de manera clara.

Los códigos de errores y descripciones establecidos para el servicio de interoperabilidad siempre serán documentados de acuerdo a lo establecido en el punto 10.3 (Documentación), de modo que el consumidor pueda consultar la lista de códigos para utilizarlos en la implementación de su cliente y para comunicarse con el productor del servicio para conocer la causa del error.

7.6 Codificación

Las entidades que participen en el intercambio de datos de un servicio de interoperabilidad (aún si estuviera en otro idioma) tienen que interpretar de la misma manera los caracteres utilizados para la transmisión de mensajes.

Por este motivo, se recomienda siempre utilizar la codificación (encoding) UTF8 en todos los servicios de interoperabilidad.

7.7 Zona horaria

Se recomienda que todos los servicios web de interoperabilidad utilicen la misma zona horaria, esto con el fin de que todas las entidades conciban los tiempos de la misma manera. Para el caso del Estado Plurinacional de Bolivia se utilizará GMT-4.

7.8 Recomendación general

Al momento de la redacción del documento, las dos tecnologías más utilizadas son

REST y SOAP, las cuales se detallan en el Anexo 15.2 (Tipos de servicios web).

En general se sugiere utilizar REST para contar con la misma tecnología en todas las entidades del sector público u organismos del Estado Plurinacional de Bolivia. Véase el Anexo 15.2.1 (REST).

8 SEGURIDAD

Seguridad de la información es la preservación de la confidencialidad, integridad y disponibilidad de la información; también pueden estar involucradas otras propiedades como la autenticidad, responsabilidad, no repudio y confiabilidad[3].

8.1 Seguridad en la capa de transporte

Es necesario contar con protocolos de seguridad en el canal o medio por el cual se transmitirá la información, esto para evitar que ajenos puedan verla o modificarla. Existen diversas tecnologías para este fin, como ser:

8.1.1 TLS (Transport Layer Security o seguridad de la capa de transporte)

El protocolo TLS permite la identificación y autenticación de las entidades a nivel del protocolo de transporte, consiguiendo que la comunicación sea confidencial y que la información enviada y/o recibida esté íntegra.

Este protocolo provee un medio seguro para comunicarse, siendo su principal propósito la confidencialidad de la información transferida. La integridad de la información intercambiada se logra autenticando los mensajes en cada transmisión.

Es recomendable siempre utilizar TLS en procesos de interoperabilidad que requieren algún tipo de autenticación y/o confidencialidad, ya sea con claves obtenidas de una entidad certificadora o con claves autogeneradas, dado que su implementación en general es sencilla de realizar, comparada con otras tecnologías.

8.1.2 VPN (Virtual Private Network o red privada virtual)

Una VPN es un canal privado cifrado de comunicación. Se trata de un ambiente de comunicaciones donde el ingreso es restringido y habilitado solo para las partes que quieren comunicarse; es decir, que el canal no puede ser visto ni entendido por ajenos.

Una VPN ofrece un canal seguro de transmisión sobre una red no segura, con las siguientes propiedades: confidencialidad, integridad y autenticación. Todas estas propiedades son añadidas a la seguridad propia del servicio de interoperabilidad (el servicio de interoperabilidad podría tener autenticación y autorización).

■

[3] Decreto Supremo N°1793, artículo 3, párrafo 6, inciso b).

Esta tecnología se utiliza cuando se requiere transmitir información sensible, dado que al existir diversas tecnologías para la implementación de VPNs, la conexión entre productos de distintos proveedores suele ser compleja.

8.1.3 Recomendaciones generales

Se recomienda siempre usar TLS. Sin embargo si los datos necesitan un mayor grado de seguridad en el transporte, se debe utilizar una VPN.

Es importante tomar en cuenta que el uso de estas tecnologías no es exclusivo, es decir que se puede utilizar una VPN con TLS.

8.2 Seguridad de los datos

8.2.1 Autenticación y autorización

Al publicar datos a través de un servicio de interoperabilidad es necesario identificar quién puede acceder al servicio, salvo en los servicios de interoperabilidad que son de acceso público. La autenticación es la verificación de la identidad de la entidad consumidora, mientras que la autorización es la verificación de los permisos que tiene dicha entidad sobre un recurso específico.

Siempre se debe realizar primero la autenticación y luego la autorización. Esto debe verificarse en cada petición de la entidad consumidora hacia el servicio.

La autenticación y autorización pueden efectuarse de diversas maneras, a continuación se detallan las principales:

8.2.1.1 Autenticación básica (Basic auth)

La autenticación básica es el método más simple y utiliza un usuario y contraseña para identificar a la entidad consumidora.

Se recomienda usar este tipo de autenticación en los servicios de interoperabilidad cuando los datos que se quiere intercambiar no tengan muchas restricciones de uso. Además, este tipo de autenticación permitirá identificar mínimamente el usuario de la entidad consumidora. Cabe destacar que este tipo de autenticación puede ser utilizado tanto con un tipo de servicio de interoperabilidad REST como SOAP.

Es recomendable implementar procedimientos de uso y ciclo de vida de las contraseñas.

8.2.1.2 Autenticación con certificados digitales

La autenticación por medio de certificados se realiza solicitando el certificado digital del consumidor del servicio de interoperabilidad (cliente) y verificando cada mensaje enviado contra dicho certificado, lo que garantiza que el consumidor del servicio es quien dice ser.

Si bien es posible realizar la autenticación por este medio, la administración de certificados para un servicio de interoperabilidad con una gran cantidad de clientes es compleja, por lo que no se recomienda su uso existiendo otros medios de autenticación más sencillos e igualmente seguros.

Si se va a implementar la autenticación con certificados digitales, es necesario comprobar que el certificado haya sido entregado por una Autoridad Certificadora, que no haya expirado y no haya sido revocado.

8.2.1.3 Autenticación con JWT (JSON Web Tokens)

JWT[4] es el medio principal de autenticación en servicios de tipo REST, es un medio compacto y autocontenido para enviar datos de manera segura entre las partes en formato JSON.

Un JWT puede ser firmado tanto con un algoritmo simétrico como uno asimétrico.

Los JWT consisten de tres partes separadas por puntos (.), las cuales son: cabecera (header), cuerpo (payload) y la firma (signature).

Se debe considerar el cifrado de los datos del cuerpo si estos se consideran sensibles, véase el punto 8.2.3 (Confidencialidad).

Por su facilidad de implementación al trabajar con un servicio REST, se recomienda siempre utilizar JWT, sin olvidar que muchas veces es necesario revocar un token, por lo cual se sugiere que como mínimo esta implementación soporte la revocación además del tiempo de expiración del JWT.

■

[4] https://tools.ietf.org/html/rfc7519

8.2.1.4 Autenticación y autorización con OAuth (OAuth 1.0a y OAuth2)

Existen dos versiones: OAuth 1.0a y OAuth2, siendo esta última la más utilizada.

OAuth2 es un protocolo de autorización y la manera de realizar la autenticación va más allá del alcance de la especificación[5], por lo que se puede implementar el proceso de autenticación que se desee. Una gran mayoría de las implementaciones del estándar ya ofrecen mecanismos de autenticación.

En OAuth2 existen varios roles: el cliente (o la aplicación que intenta tener acceso a la información del usuario), el servidor de recursos, el servidor de autorización (es el que pregunta al usuario si realmente quiere dar los permisos al cliente) y el usuario. Existen, además varios flujos o maneras para obtener un token, las principales son:

- Otorgamiento de credenciales de cliente (Client Credentials Grant): Este método se utiliza comúnmente para comunicaciones de servidor a servidor, el cliente envía sus credenciales al servidor de autorización y este retorna un token firmado (el cliente y el usuario son los mismos).
- Otorgamiento de código de autorización (Authorization Code Grant): Usado comúnmente en aplicaciones web cuando el cliente quiere acceder a recursos protegidos en favor del usuario. El flujo es el siguiente: el cliente redirecciona al usuario al servidor de autorización; entonces se solicita al usuario ingresar sus credenciales en el servidor de autorización y aprobar la solicitud del cliente; luego el cliente —con el permiso que ya dio el usuario—, negocia un token de acceso con el servidor de autorización (es una llamada en favor de un usuario, el usuario y el cliente son distintos).

OAuth resulta complejo en su implementación; sin embargo, destaca cuando se espera contar con una gran cantidad de servicios de interoperabilidad, ya que al tratarse de un servicio centralizado, su administración resulta más sencilla de aplicar transversalmente.

■

[5] https://tools.ietf.org/html/rfc6749#section-3.1

8.2.1.5 Autenticación y autorización con OpenID Connect

OpenID Connect[6] es un protocolo que al basarse sobre los protocolos de OAuth2 y OpenID provee tanto autenticación como autorización, por lo que se considera más completo.

En la actualidad, su especificación se realiza con mensajes JSON sobre HTTPS y se recomienda su uso sobre versiones anteriores de OpenID que se marcaron como obsoletas[7].

OpenID Connect se basa en el concepto de identidad, que se define como un conjunto de atributos que identifican a los usuarios de forma exclusiva y que permite a las aplicaciones cliente confiar en la autenticación realizada por un proveedor de OpenID Connect para verificar la identidad de un usuario.

Es necesario recalcar que su implementación es compleja con relación a otros mecanismos como JWT.

8.2.1.6 Recomendaciones generales

Si la entidad carece de un mecanismo de autenticación y autorización ya definido, se recomienda utilizar JWT para servicios REST y BASIC para SOAP.

Si la entidad ya cuenta con un mecanismo de autenticación y autorización, se recomienda mantener este mecanismo y solo realizar acciones de documentación y respaldo de su funcionamiento.

Se recomienda la implementación de OAuth u OpenID Connect a mediano o largo plazo.

Se sugiere utilizar un solo mecanismo de autenticación y autorización, esto para evitar complejidad administrativa en el largo plazo.

Es necesario asegurarse que a las funciones administrativas de autenticación y autorización del servicio de interoperabilidad solamente puedan acceder administradores del servicio y no así consumidores.

Para REST se sugiere implementar un Administrador de APIs (Api Gateway). Esto ■

[6] http://openid.net/specs/openid-connect-core-1_0.html
[7] https://openid.net/developers/specs/

posibilita aplicarlo transversalmente a todos los servicios de interoperabilidad (tanto la autenticación como la autorización), además de que provee otros mecanismos de control como listas de acceso, logs, límite de solicitudes, etc. Dependiendo de la herramienta que se utilice, las características pueden variar, sin embargo se sugiere el uso de Kong[8] por ser una herramienta completa.

Si la naturaleza del servicio es transaccional, se recomienda necesariamente aplicar un mecanismo de autorización.

Se recomienda realizar un inventario de vulnerabilidades técnicas respecto a la implementación del tipo de autenticación y autorización adoptada previo al despliegue en producción. Estas vulnerabilidades pueden corresponder al tipo de cifrado que se utilice, manejo de contraseñas entre otras.

8.2.2 Integridad

Usualmente los mismos protocolos de transmisión de datos otorgan integridad pero sólo en el nivel de transporte, esto no siempre es suficiente al implementar un servicio de interoperabilidad, ya que sólo se garantiza que la información no haya sido alterada en el canal, aunque luego puede haber sido alterada por otros procesos.

Los servicios de interoperabilidad deben garantizar la "exactitud" y "completitud" de los datos utilizando mecanismos como la Firma Digital, que evitan la alteración de los datos tanto en el canal como en el lugar donde se procesarán o almacenarán.

A continuación se detallan los mecanismos que nos garantizan la integridad de los datos:

8.2.2.1 Registro centralizado de hashes

Se puede utilizar una base de datos centralizada para garantizar la integridad de los mensajes. Esto se realiza distribuyendo un resumen (hash) de los datos intercambiados a una base de datos de confianza donde se almacenan. Junto con estos resúmenes se almacena el tiempo (timestamp) a fin de verificar el momento en que se haya efectuado una solicitud.

■

[8] https://getkong.org/

Es necesario que esta base de datos sea accesible para las entidades participantes, de tal forma que cada una pueda obtener un respaldo de la misma, si así lo deseara.

Este medio de verificación de la integridad se utilizará solamente si las entidades que requieren intercambiar datos establecen un acuerdo entre ellas, donde aceptan la validez de dicho registro.

8.2.2.2 Firma digital

La Firma Digital asocia al firmante con un documento (un mensaje en interoperabilidad) brindando autenticidad, integridad y no repudio.

La Firma Digital es parte de una infraestructura denominada PKI que permite identificar un certificado digital con la identidad de una persona o entidad. Con este mecanismo es posible verificar si la información ha sido modificada, incluso si solamente se hubiera cambiado un caracter.

Es posible incluso tener una infraestructura PKI propia, si todas las entidades que intervienen aceptan su validez.

Para realizar la interoperabilidad entre las partes es necesario que estas intercambien las claves públicas, manteniendo las claves privadas en lugar seguro (esto solo si son certificados autofirmados, ya que si los certificados son generados por una Entidad Certificadora Pública, esta se encarga de verificar las claves).

Los certificados para realizar la Firma Digital entre servidores se pueden obtener de varias maneras. A continuación detallamos las más utilizadas:

- Con una Entidad Certificadora Pública (CA o Certificate Authority): entidad a la cual se acude para obtener certificados digitales para utilizarlos en el intercambio de datos.
- Con Certificados Autofirmados: en este caso, los certificados son generados por las entidades que desean intercambiar datos. El intercambio de las claves públicas puede efectuarse por cualquier método entre las partes; sin embargo, para darle un carácter legal, el intercambio de claves puede realizarse ante un notario de fe pública.

8.2.2.2.1 Sellado de tiempo

El sellado de tiempo se utiliza para especificar el momento en el cual se ha aplicado la Firma Digital. Usualmente el sellado de tiempo es parte de la infraestructura PKI y a la autoridad de sellado de tiempo se le denomina TSA (Time Stamping Authority).

El sellado de tiempo indica que los contenidos firmados digitalmente existieron en un momento dado y que no han cambiado desde ese instante. Sin la existencia de sellado de tiempo, no es posible saber cuándo se ha utilizado la Firma Digital o si ésta ha sido aplicada con un certificado válido en el momento del firmado.

Se sugiere usar el sellado de tiempo en conjunto con la Firma Digital, ya que permite saber cuándo se ha efectuado la firma de los datos, si se ha realizado con una clave válida o si la información es reciente.

8.2.2.3 Cadena de bloques (Blockchain)

La cadena de bloques es un registro público en orden cronológico de las transacciones que se realizan. Esta cadena se comparte entre los usuarios, lo que permite verificar que un evento ha ocurrido, garantizando la integridad. En otras palabras, es un registro de transacciones descentralizado y distribuido, donde cada cliente puede tener su propia copia, lo cual evita modificaciones por parte de terceros.

En interoperabilidad se podría utilizar generando un resumen (hash) del mensaje que contenga los datos a intercambiar e insertando este resumen en la cadena de bloques, que al ser distribuida es muy difícil de alterar.

8.2.2.4 Recomendaciones generales

Se recomienda el uso de la Firma digital en todos los servicios de interoperabilidad que intercambien datos sensibles, tomando en cuenta la complejidad que esto implica.

Si se va a utilizar la firma digital es conveniente establecer mecanismos seguros para el intercambio de las claves públicas entre las entidades (si fuera necesario) y el almacenamiento de las claves privadas en lugar seguro, garantizando así la integridad de los datos y evitando que estas claves se pierdan en caso de alguna

eventualidad. Además es necesario verificar que los certificados no hayan expirado y que no hayan sido revocados.

En SOAP se sugiere utilizar un XML Signature[9] de acuerdo a la W3C o WS-Security; en REST, el RFC 7515[10] de JWS (JSON Web Signature) para la firma de JSON.

8.2.3 Confidencialidad

Cuando los datos fluyen de un lado a otro, muchas veces a través de varios intermediarios, se debe evitar que sean entendidos por terceros; más aún, es necesario que dichos datos no sean entendibles por administradores de la infraestructura y otros con acceso al mismo servicio de interoperabilidad. El método más común para mantener la confidencialidad de los datos es el cifrado.

El cifrado es el proceso de convertir los datos a un formato no legible por ajenos no autorizados (terceros que observen el canal de transmisión, administradores del servicio de interoperabilidad u otros). Los datos solamente puede ser entendidos por aquellos que posean la clave para descifrarlos.

Los métodos utilizados para el cifrado son el simétrico y el asimétrico, que detallamos a continuación:

8.2.3.1 Cifrado simétrico

El cifrado simétrico se realiza con una clave secreta y es seguro siempre que dicha clave sea mantenida en buen resguardo. En interoperabilidad, basta que las partes intercambien la clave de manera segura para establecer una comunicación con un nivel aceptable de cifrado y a un costo computacional bastante bajo (tiempo de procesamiento).

El algoritmo más utilizado para realizar el cifrado simétrico es AES (Advanced Encryption Standard o estándar avanzado de cifrado), que es bastante eficiente y al procesarse consume pocos recursos. Puede utilizarse con claves de longitud 128, 192, 256 o de una mayor cantidad de bits, siendo la mayor diferencia entre estas el tiempo que toma descifrar los datos.

Una de las principales desventajas de este tipo de cifrado es el intercambio de

[9] https://www.w3.org/2008/xmlsec/
[10] https://tools.ietf.org/html/rfc7515

la clave, que se convierte en un riesgo cuando deja de ser secreta (cualquiera en posesión de la clave puede descifrar los datos).

8.2.3.2 Cifrado asimétrico

El cifrado asimétrico usa dos claves: una para el cifrado y otra para el descifrado. Se utiliza una clave pública para aplicar el cifrado y una clave privada para descifrar los datos (solamente se puede realizar el descifrado de los datos con la clave privada).

En este tipo de cifrado no existe la necesidad de realizar el intercambio de claves, evitando el problema de distribución de claves.

Se recomienda el uso de RSA con una clave de longitud mínima de 2048.

8.2.3.3 Recomendaciones generales

En ningún caso es recomendable implementar un algoritmo de cifrado propio.

Se recomienda utilizar un cifrado simétrico siempre que no se encuentre sobre un canal seguro (TLS o VPN) para evitar que terceros puedan entender los datos enviados.

También se recomienda utilizar el cifrado asimétrico si los datos que se quieren intercambiar son sensibles y solamente pueden ser conocidos por usuarios específicos.

En general, si se requiere de mayor seguridad para los datos, se sugiere usar el cifrado asimétrico.

8.3 Auditoría

La auditoría es la revisión y comprobación de las acciones realizadas para reconstruir una serie de eventos que generaron un hecho específico. La manera más común para efectuar estas auditorías es a través de registros de eventos (logs) en servidores seguros y solamente accesibles por personal autorizado.

Se recomienda utilizar siempre los registros de eventos con toda la información posible. En este marco, lo mínimo que deberían contener son: información temporal, información de la aplicación que está realizando el registro del evento, quién

realizó el evento (datos del cliente), el tipo de evento que se haya realizado y la solicitud y respuesta del servicio.

No se debe incluir en los registros de eventos información sensible del cliente (contraseña por ejemplo), rutas a archivos o cadenas de conexión a la base de datos, ya que esto constituye un riesgo de seguridad.

Se debe documentar en informes postmortem las acciones correctivas que se tomaron en caso de cualquier suceso que haya afectado al servicio, con la finalidad de que si los mismos sucesos se repitieran, estos se resuelvan fácilmente.

Se recomienda utilizar un centralizador de registros de eventos para facilitar su administración y posterior análisis (ELK[11], Apache Kafka[12] u otro).

También se puede realizar la auditoría con el registro centralizado de hashes en caso de haberse implementado este mecanismo; véase el punto 8.2.2.1 (Registro centralizado de hashes).

Además, se pueden utilizar tecnologías de sincronización de relojes de sistemas (NTP, Network Time Protocol) con la finalidad de posibilitar un seguimiento cronológico de los eventos.

■

[11] https://www.elastic.co/
[12] https://kafka.apache.org/

9 DISPONIBILIDAD

La entidad publicadora del servicio de interoperabilidad se esforzará para que el servicio esté disponible de forma continua y sin interrupción. A continuación se detallan las características más importantes sobre la disponibilidad de los servicios de interoperabilidad:

9.1 Redundancia

Para garantizar el estado de los servicios de interoperabilidad es recomendable eliminar los puntos únicos de fallo. Esto se logra incorporando componentes redundantes (servidores, enrutadores, energía, software, microservicios, refrigeración, IPs, nombre de dominio y otros), para que en caso de catástrofes o incidencias, el componente pueda reemplazar las tareas del otro que hubiese presentado fallas. Es recomendable poner en práctica este punto para que así se tolere la pérdida o desperfectos funcionales de algún componente de la infraestructura.

No siempre es posible eliminar todos los puntos únicos de fallo debido al costo que esto implica (hacer una evaluación de riesgo contra costo); sin embargo, de acuerdo a la criticidad del servicio de interoperabilidad, se recomienda contar con un inventario de toda la infraestructura necesaria para el funcionamiento del servicio y evaluar qué componentes son los que tienen más probabilidad de fallo, para comenzar a aplicar la redundancia.

Cabe mencionar que existen arquitecturas de software que facilitan la redundancia de los elementos más utilizados de un servicio de interoperabilidad, como los microservicios.

9.2 Pruebas de rendimiento

Es recomendable que los servicios de interoperabilidad sean sometidos a un conjunto de pruebas de rendimiento para verificar su escalabilidad, fiabilidad y uso de recursos. Se recomienda utilizar los siguientes tipos:

- Pruebas de carga: realizadas para monitorear el comportamiento de la aplicación bajo una cantidad de peticiones alta. Esto nos permite conocer el límite de peticiones que pueden ser respondidas, lo que a su vez permitirá establecer un límite de consultas o utilizar un balanceador de carga, para

mejorar el rendimiento del servicio de interoperabilidad evitando su interrupción para los consumidores.

- Pruebas de estrés: con estas pruebas se puede verificar que el servicio de interoperabilidad responda de manera adecuada cuando sobrepasa las condiciones normales de consumo.

En base a las mediciones de los resultados de las pruebas de rendimiento (carga y estrés), tomar decisiones en cuanto a los elementos que necesitarán redundancia, ya sean de software o hardware. Véase el punto 9.1 (Redundancia).

Además es posible verificar el rendimiento de los servicios de interoperabilidad realizando un análisis de los tiempos de respuesta en los registros de eventos. Véase el punto 8.3 (Auditoría).

9.3 Balanceo de carga

La infraestructura encargada de proveer los servicios de interoperabilidad debe tener la capacidad de distribuir el trabajo entre los recursos con los que cuenta, a fin de evitar problemas en el momento de gestionar una cantidad considerable de solicitudes. Esta necesidad puede ser satisfecha con servidores de proxy reverso (reverse proxy), balanceadores de carga, enrutadores o grupos de servidores (clusters).

Se recomienda aplicar el balanceo de carga en los servicios de interoperabilidad que se espera tengan una cantidad alta de entidades consumidoras. Hay que evaluar la cantidad de solicitudes que el servicio de interoperabilidad puede manejar en base a pruebas de rendimiento y de acuerdo a ese límite establecer nuevos recursos que permitan manejar una mayor cantidad de solicitudes.

Si bien esto aumentará la cantidad de solicitudes que se puede manejar, es conveniente revisar si el servicio de interoperabilidad no tiene algún problema que evite que pueda responder a más solicitudes, como conexiones no cerradas a la base de datos, no limpiar recursos en la memoria, consultas lentas a la base de datos u otros que pudieran estar afectando el rendimiento del servicio de interoperabilidad. La principal desventaja es que se necesita una mayor configuración para poner en marcha el balanceo de carga y el costo que esto implica.

9.4 Cacheado

Es necesario que al identificar los recursos de interoperabilidad consumidos con una frecuencia considerablemente alta, estos sean almacenados provisionalmente en una memoria extra para aligerar el proceso de consulta y respuesta en los servidores.

El cacheado se emplea comúnmente en operaciones idempotentes y puede efectuarse dentro del mismo servicio de interoperabilidad (cargando datos que se soliciten con frecuencia a una estructura en memoria como redis[13]) o como parte de su arquitectura (un servicio extra entre el cliente y el servicio de interoperabilidad como varnish[14]).

La arquitectura REST, al ser multicapa, permite la inclusión de manera sencilla de una caché.

Se recomienda el uso del cacheado en servicios de interoperabilidad de lectura que requieran un alto rendimiento (se necesita responder a una alta cantidad de solicitudes al mismo tiempo).

Es importante tomar en cuenta que, en caso de modificación de datos, la caché también debe actualizarse ya que una cache no actualizada respondería con datos incorrectos.

9.5 Monitoreo

Un factor importante a considerar en los servicios de interoperabilidad para garantizar su disponibilidad es el constante monitoreo de su estado, esto facilita tomar las acciones pertinentes en caso de problemas, registrándolos y alertando a los administradores del servicio para poder aplicar acciones correctivas y tener un informe estadístico de incidencias en un lapso determinado tiempo.

Las estadísticas contemplan: número total de peticiones por cliente, número total de peticiones correctas y número total de peticiones con error. Deben incluir los filtros por criterios que se consideren más importantes, como la fecha o rangos de fechas para estas peticiones.

También se recomienda verificar los registros de eventos con el propósito de monitorear si el servicio está funcionando correctamente.

■

[13] https://redis.io/
[14] https://varnish-cache.org/

Se recomienda siempre aplicar algún mecanismo de monitoreo automático para los servicios de interoperabilidad.

Para facilitar el monitoreo automático del estado del servicio se recomienda publicar una operación especial que devuelva una respuesta que permita verificar si el servicio se encuentra disponible. Esta operación debería tener solamente la restricción de autenticación.

Por ejemplo en REST:

```
GET /estado
HTTP/1,1 200 OK
        {
                "estado": "El servicio se encuentra disponible"
        }
```

En SOAP se recomienda que esta operación se llame "verificarEstado" y que retorne la siguiente estructura XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<respuesta>
        <estado>El servicio se encuentra disponible</estado>
</respuesta>
```

Adicionalmente, se recomienda publicar una página del estado del servicio con un detalle de los errores que se hubieran registrado y el estado general de salud del mismo.

10 ACCESIBILIDAD

Si una entidad publica un servicio de interoperabilidad, es recomendable que disponga de una serie de mecanismos para simplificar la forma de acceder al mismo, especialmente para el primer acceso.

Los siguientes mecanismos simplificarán el uso del servicio de interoperabilidad para las entidades consumidoras y en ningún caso son limitantes (pueden existir otros mecanismos) para la publicación del servicio:

10.1 Entorno de pruebas

Es recomendable contar con un entorno de pruebas (aparte del entorno de producción). El objetivo de este entorno es simular el ambiente de producción (el entorno de pruebas y producción deben ser idénticos en términos de sintaxis, condiciones de conexión y seguridad, siendo diferentes solamente por los datos provistos, por su nivel de disponibilidad y por los recursos que disponen) con el fin de poner a prueba el recurso desarrollado y así asegurar su estabilidad en base a pruebas constantes.

10.2 Software de consumo

La entidad publicadora del servicio de interoperabilidad hace disponibles una serie de herramientas de consumo en forma de librerías, módulos o paquetes en los lenguajes de programación más populares o que sean de mayor requerimiento por las entidades. El software libre de consumo facilita a las entidades la integración de sus aplicaciones con los servicios de interoperabilidad. Este software de consumo viene acompañado de su documentación en forma de manuales de uso o manuales técnicos.

Se recomienda que en la implementación de este software participe la comunidad publicándose luego en el repositorio de software libre, tomando en cuenta que no se publiquen parámetros u otra información que pueda ser un riesgo de seguridad para el servicio de interoperabilidad.

10.3 Documentación

Es recomendable que el servicio de interoperabilidad permita generar su documentación, que posteriormente será publicada para los consumidores. Esta do-

cumentación se escribe al momento de la implementación del servicio y dependiendo del tipo de servicio de interoperabilidad y las herramientas que provee el lenguaje de programación.

La documentación tendrá un detalle de cada operación, con una descripción de cada uno de los parámetros de entrada y salida (ver punto 7.1 Definición de parámetros de entrada/salida), ejemplos de consumo y manejo de errores (ver punto 7.5 Manejo de errores).

10.4 Manuales

Se recomienda que la entidad publicadora del servicio de interoperabilidad elabore un manual de uso y un manual técnico, con las siguientes características:

- El manual de uso: contempla la forma de uso del servicio de una manera más detallada que la documentación publicada. El manual de uso describe cómo realizar el consumo de las diferentes operaciones, explicando detalladamente aspectos como la autenticación y autorización, integridad y confidencialidad en los mensajes. También considera aspectos como el manejo de errores o manejo de catálogos de errores. Este manual puede contener ejemplos concretos de consumos haciendo uso del entorno de pruebas con instrucciones paso a paso.

  Este manual está orientado al desarrollador del cliente del servicio de interoperabilidad.

- El manual técnico: contempla los aspectos técnicos que permitan realizar la comunicación entre las partes para acceder al servicio, considerando principalmente la seguridad en la capa de transporte, de la forma más detallada posible. También considera aspectos de resolución de problemas de comunicación más comunes. Este manual puede contener gráficos que ayuden a comprender la topología de la comunicación.

  Este manual esta destinado al personal de infraestructura de la entidad que desea consumir el servicio de interoperabilidad.

10.5 Portal Web

Con el fin de facilitar la verificación del consumo del servicio con ejemplos concretos, haciendo uso de los parámetros de entrada y verificando los parámetros de salida tal como se describe en la documentación, se recomienda que la entidad publicadora del servicio de interoperabilidad cuente con un portal web, en el cual se puedan realizar consultas desde una interfaz gráfica, simulando una petición del cliente.

10.6 Recomendaciones generales

Se recomienda abrir un canal de comunicación con las entidades consumidoras para recibir retroalimentación (críticas, observaciones, recomendaciones y otros) que permita mejorar la calidad del servicio.

Es muy recomendable el servicio de soporte en línea, mediante números de teléfono o chats con personas de contacto que atiendan las consultas, para que la experiencia de la puesta en marcha para el consumidor sea lo más simple posible.

Para mejorar la accesibilidad al servicio de interoperabilidad, es recomendable que todos los mecanismos sean implementados.

En general, cualquier otro mecanismo que facilite el acceso al servicio para el consumidor es recomendado.

11 POLÍTICAS

Es de vital importancia contar con políticas que definan explícitamente la responsabilidad, accesos y otros que fueran necesarios. La entidad publicadora es la que establece, de forma clara, dichas políticas con relación al servicio de interoperabilidad.

Los principales aspectos a contemplarse al desarrollar las políticas son:

11.1 Aspectos legales

Cada entidad de manera unilateral genera la normativa jurídica para la producción de sus datos y las modalidades de su intercambio.

La entidad que proporcione un servicio de interoperabilidad debe tener la competencia legal para dicho propósito.

El requerimiento o consulta de datos sólo será restringido si existe alguna normativa legal vigente que impida su publicación y/o atente contra la seguridad nacional.

11.2 Datos a intercambiar

Es necesario que se definan de manera clara los datos que se van a intercambiar, además del uso que se le dará a los mismos.

Definir también qué datos son considerados sensibles.

Definir el tratamiento que se le dará a los datos obtenidos a partir del consumo del servicio, precautelando la seguridad de los datos.

11.3 Obligaciones

Definir de manera clara las entidades que participan en el intercambio de datos, así como las obligaciones que tengan.

Las entidades publicadoras son responsables sobre la calidad de sus datos, así como también de su actualización.

11.4 Responsabilidades

Se deberán detallar todas las responsabilidades tanto de la entidad publicadora como de la entidad consumidora.

Las sanciones, en consecuencia a las faltas incurridas por el incumplimiento de las obligaciones y clasificadas de acuerdo al grado de severidad de la falta, pueden generar responsabilidades, que entre otras podría incluir la revocatoria de accesos al servicio.

11.5 Vigencia

La fecha límite de acceso al servicio de interoperabilidad estará determinada en el documento de manera explícita.

12 CATÁLOGO DE SERVICIOS

El catálogo de servicios es una lista organizada de los servicios de interoperabilidad de las entidades públicas del Estado Plurinacional de Bolivia.

Para cada servicio de interoperabilidad existirá una ficha de descripción que contendrá toda la información necesaria para conocer, evaluar y solicitar acceso al mismo (si bien un servicio de interoperabilidad se encuentra publicado en el catálogo, esto no implica de ninguna manera que ya se pueda consumir el servicio. El dar acceso a los servicios es competencia netamente de la entidad publicadora en el marco de las políticas de acceso).

12.1 Ficha de un servicio de interoperabilidad

Para cada servicio de interoperabilidad existirá una ficha que detallará un conjunto de metadatos que permita identificarlos, conocer qué información proveen, verificar qué servicios relacionados existen y otros. En el anexo 15.6 (Ejemplo de ficha de metadatos de servicios de interoperabilidad) se encuentra un ejemplo de ficha de metadatos de servicio de interoperabilidad.

12.2 Perfil de metadatos de servicios de interoperabilidad

El perfil de metadatos nace a partir de la necesidad de uniformar y centralizar la información de los servicios de interoperabilidad. El perfil de metadatos se obtuvo de un proceso de recopilación de experiencias de otros países, estándares internacionales y del trabajo de la mesa de Interoperabilidad del CTIC-EPB.

El perfil de metadatos de servicios define los siguientes datos principales que permitirán la identificación de un servicio de interoperabilidad, divididos en la siguientes secciones:

12.2.1 Datos generales

Los datos generales permiten identificar las características más comunes de los servicios de interoperabilidad.

Datos Generales		
Identificador	Identificador único del servicio	Requerido
Nombre	Nombre descriptivo del servicio	Requerido
Descripción	Descripción comprensiva del propósito y alcance del servicio	Requerido
Estado	Describe el estado del servicio (Activo, Inactivo, Mantenimiento, etc)	Requerido
Fecha de Inicio de disponibilidad	Fecha desde la cual está disponible el servicio	Requerido
Fecha de Fin de disponibilidad	Fecha en la que el servicio deja de estar disponible	
Publicador	Entidad responsable del servicio	Requerido
Fecha de registro	Fecha en la que se registra la información del servicio en el catálogo	Requerido
Fecha de última actualización	Última fecha en la que se actualizó la información del servicio	
Versión	Versión del servicio	Requerido
Documentación	Links con documentos en línea que proveen información adicional relacionada al servicio	
Servicios Relacionados	Enlaces a otras fichas relacionadas en el catálogo de servicios	
Palabras Clave	Palabras que son relevantes al servicio	Requerido
Tipo de Acceso a la información	Indica si la información que provee el servicio es pública o es confidencial	

12.2.2 Restricciones de uso

Esta sección contiene las políticas que se deben cumplir en el consumo del servicio de interoperabilidad.

Restricciones de uso		
Base Legal	Marco normativo por el cual se hace uso del servicio de intercambio de información	Requerido
Prerrequisitos	Detalle de las condiciones que se deben cumplir (normativamente) para poder realizar el consumo del servicio	Requerido
Restricciones adicionales	Otras restricciones que se presenten a la hora de consumir el servicio	

12.2.3 Contacto institucional

En esta sección se encuentra información de contacto para consultas a la entidad publicadora.

Contacto Institucional		
Unidad Responsable	Unidad o dependencia que se encuentra a cargo del servicio	Requerido
Correo Electrónico	Correo electrónico genérico de la unidad o dependencia propietaria del servicio	Requerido
Teléfono	Teléfono de contacto de la unidad o dependencia encargada del servicio	Requerido

12.2.4 Información técnica

Esta sección contiene todos los aspectos relacionados al tipo de servicio, entornos y parámetros de entrada/salida.

Información Técnica		
Dirección Física del servicio	URL de consumo del servicio	
Tipo	Tecnología utilizada para la obtención de la información del servicio (SOAP, REST u otro)	Requerido
Entorno	Entorno de desarrollo, pruebas, producción u otro	Requerido
Datos de Entrada	Detalle de los parámetros o modificadores de entrada del servicio	Requerido
Datos de Salida	Detalle de la estructura de respuesta del servicio	Requerido
Tipo de Conexión	Medio por el cual se transmiten los datos del proveedor al cliente	Requerido
Software Relacionado	Vínculos de software relacionado al servicio	

12.2.5 Seguridad

Esta sección nos permite conocer las características de seguridad que tiene el servicio de interoperabilidad.

Seguridad		
Políticas de Seguridad	Políticas de seguridad implementadas para consumir el servicio	Requerido
Tipo de autenticación	Describe qué tipo de autenticación se utiliza para consumir el servicio	
Firma Digital	Afirmación o negación del uso de Firma Digital	
Transporte	Detalle del tipo de seguridad que se aplica en el transporte de los datos (HTTP, HTTPS, VPN u otros)	Requerido

12.3 Actores

Los actores identificados en el catálogo de servicios de interoperabilidad son:

- Usuario publicador: Servidor público de la entidad que tiene la responsabilidad de registrar y mantener los servicios en el catálogo.
- Usuario operador: Servidor público que tiene la responsabilidad de validar la información registrada en el catálogo.
- Usuario visitante: Cualquier persona que desea ver información de acuerdo al perfil de metadatos de servicios.

12.4 Uso del catálogo de servicios de interoperabilidad

Las operaciones permitidas en el catálogo son:

- Solicitud de acceso al catálogo: El usuario publicador debe registrar su solicitud para acceder al catálogo en un formulario de registro que será aprobado posteriormente por el usuario operador.
- Solicitud de publicación o modificación de servicios: El usuario publicador debe registrar su solicitud en el formulario de registro de servicios conforme a lo establecido en el perfil de metadatos de servicios para su posterior aprobación por el operador del catálogo.
- Búsquedas en el catálogo de servicios: Los usuarios podrán realizar búsquedas sobre los servicios por los medios más simples posibles.

Todas las fichas del catálogo pueden exportarse en los formatos (JSON, XML, PDF y CSV) haciendo posible su procesamiento automático en caso de ser necesario.

Adicionalmente, se publicarán estadísticas de la disponibilidad del servicio del resultado de un proceso de monitoreo de cada servicio (ver punto 9.5 Monitoreo).

12.5 Recomendaciones generales

La entidad publicadora debe llenar la ficha correspondiente al servicio de interoperabilidad, especificando cada recomendación aplicada durante la implementación del servicio en las siguientes secciones: datos generales, restricciones de uso, contacto institucional, información técnica y seguridad.

13 RESUMEN

A lo largo del documento se establecieron una serie de recomendaciones generales que, en función de las necesidades técnicas de cada entidad pública, deberán ser cumplidos para lograr un estándar en los servicios de interoperabilidad en las entidades del sector público.

Para su implementación, se sugiere que los lineamientos sean evaluados secuencialmente, es decir, siguiendo el orden establecido. En el anexo 15.7 (Lista de verificación para la implementación de servicio de interoperabilidad) se encuentra una lista de verificación para este proceso.

14 TÉRMINOS Y DEFINICIONES

Algoritmo. Conjuntos de reglas definidas para encontrar la solución a un problema.

API. Es un conjunto de reglas y especificaciones que los programas de software pueden seguir para comunicarse entre ellos.

Administrador de APIs. Api Gateway, servicio administrable que permite la publicación, mantenimiento y monitoreo de servicios web. Es un punto único de entrada para todos los clientes que permite el redireccionamiento de las solicitudes a uno o varios servicios internos.

Autenticación. Característica que permite identificar y validar la identidad de un usuario, servicio o proceso.

Autorización. Es el proceso de dar permisos a un usuario para realizar alguna acción.

Confidencialidad. Propiedad que determina que la información no esté disponible ni sea revelada a individuos, entidades o procesos no autorizados.

Datos. Caracteres, números o símbolos recogidos para su tratamiento informático, análisis estadístico o referencia[15].

Dato Sensible. Se entiende por datos sensibles aquellos que contienen información vinculada a la privacidad, intimidad, honra, honor, propia imagen, dignidad, información de sectores económicos estratégicos e información catalogada como

[15] https://www.ctic.gob.bo/wp-content/uploads/bsk-pdf-manager/DATOS-ABIERTOS-v1.0_114.pdf

secreta, reservada o confidencial, cuya divulgación, de alguna manera, afecte a su titular o al Estado Boliviano.

Disponibilidad. Propiedad de acceso y uso de información a entidades autorizadas cuando estas lo requieran.

Entidad publicadora. Entidad responsable de la publicación del servicio de interoperabilidad.

Entidad consumidora. Entidad que obtiene datos del servicio de interoperabilidad.

Ficha del servicio de interoperabilidad. Descripción de las características principales de un servicio de interoperabilidad.

Función de Hash o de resumen. Función matemática que resume un dato largo a un dato de longitud determinada y asegura con alta probabilidad un resultado distinto para dos datos diferentes.

Idempotencia. Es la propiedad que indica que en caso de realizarse la misma solicitud varias veces a un servicio de interoperabilidad el resultado será el mismo en cuanto al estado del sistema.

Integridad. Propiedad que salvaguarda la exactitud y completitud de la información.

Interoperabilidad. Es la capacidad de intercambiar y compartir datos entre sistemas o componentes informáticos.

IETF (Internet Engineering Task Force). Es una comunidad internacional abierta que desarrolla y emite documentos de alta calidad y relevancia para el diseño, uso y gestión del Internet.

Metadatos. Son los datos que describen la calidad, contenido, condición y otras características de otros datos.

No repudio. Garantía de que un mensaje electrónico de datos o un documento digital ambos firmados digitalmente, no puedan ser negados en su autoría y contenido.

Portal Web. Es una fuente de información que provee el acceso a una serie de recursos y servicios relacionados a un mismo tema.

Punto único de fallo. Es una parte del sistema que en caso de fallar detendría el funcionamiento de todo el sistema. Puede ser cualquier dispositivo de red, servidor, software u otro.

Semántica. Significado de una unidad lingüística. En interoperabilidad es el significado o interpretación de los datos.

Servicio de interoperabilidad. Es cualquier servicio ofrecido a través de la red, diseñado para soportar interacción máquina a máquina para el intercambio de datos.

Trazabilidad. Capacidad de llevar un registro de las acciones y eventos de un sistema, servicio y/o proceso.

URI: Uniform Resource Identifier o identificador uniforme de recursos, sirve para identificar recursos en una red (internet).

15 ANEXOS

15.1 Tipos de formatos de representación de datos

Los principales tipos de formatos de datos de representación de la información en un servicio de interoperabilidad son JSON y XML, y se detallan a continuación:

JSON (JavaScript Object Notation o Notación de Objetos JavaScript)

Es un formato de representación de datos liviano, simple de ser leído y entendido tanto por humanos como por máquinas. Es el principal tipo de formato de dato para el intercambio de información en un servicio de interoperabilidad REST y, en su forma más simple, se caracteriza por el uso de una colección de pares nombre/valor que podrían ser anidados. A continuación se observa su estructura:

```
{
    "nombre1": "valor1",          // par nombre valor
    "nombre2": "valor2",
    "nombre3": {
        "nombre4": "valor4",
        "nombre5": ["val1", "val2"]          // par nombre valor, donde el valor es una colección
(array)
    }
}
```

Como ejemplo se describe el objeto "persona" en formato JSON:

```
{
    "ci": 3483646,
    "nombres": "Juan",
    "apellidos": "Perez Gomez",
    "fechaNacimiento": "22/05/2017"
}
```

XML (eXtensible Markup Language o Lenguaje de Marcado Extensible)

Es un formato de representación de datos flexible que define un conjunto de reglas para formar documentos que puedan ser entendibles por humanos y por máquinas. Este formato de tipo de dato se utiliza generalmente con servicios de interoperabilidad SOAP, aunque también es posible utilizarlo con un servicio de interoperabilidad REST.

La estructura del documento XML tiene que contener la declaración <xml> que especifique la versión (version) y la codificación (encoding) del documento, esto con el fin de mejorar la interoperabilidad de los datos (al conocer la versión y la codificación se evitan errores de interpretación).

Como ejemplo se detalla el objeto "persona" en formato XML:

```
<?xml version="1.0" encoding="UTF-8"?>
<persona>
   <ci>3483646</ci>
   <nombres>Juan</nombres>
   <apellidos>Perez Gomez</apellidos>
   <fechaNacimiento>22/05/2017</fechaNacimiento>
</persona>
```

15.2 Tipos de servicios web

Al momento de la redacción del documento los tipos de servicios web más utilizados al implementar un proceso de interoperabilidad son REST y SOAP. A continuación se exponen las definiciones y características principales de cada uno:

15.2.1 REST

REST es un estilo de arquitectura para sistemas distribuidos y, si bien no depende de ningún protocolo, en su gran mayoría se utiliza sobre HTTP (Hypertext Transfer Protocol).

Este estilo de arquitectura define un conjunto de principios para su implementación:

- Interfaz uniforme. Todos los recursos son identificados por una URI (Uniform Resource Identifier).

- Interacciones sin almacenamiento del estado (stateless). Cada mensaje tiene información suficiente para ser procesado sin necesidad de guardar un estado en el servidor de anteriores mensajes.
- Permite el uso de caché. Las respuestas pueden ser cacheadas o no por el cliente para optimizar las consultas.
- Es una arquitectura cliente-servidor. El servidor desconoce los clientes que se conectarán al mismo. Ambos lados pueden desarrollarse independientemente.
- Arquitectura por capas. El cliente desconoce si está conectado directamente al servicio, a una caché o a cualquier otra capa intermedia.

El servicio basado en estos principios se denomina RESTful.

Si bien técnicamente un servicio REST puede transferir la información en cualquier formato, se recomienda utilizar JSON para el intercambio de datos. Véase el anexo 15.1 (Tipos de formatos de representación de datos).

15.2.1.1 Buenas prácticas

Para mantener la consistencia en el desarrollo de los servicios REST, se sugiere implementar las siguientes buenas prácticas:

- Contar con manejo de respuestas de tipo JSON por defecto, y solo en caso de necesidad, otro tipo de contenido (XML, CSV u otro)
- Los URI (Uniform Resource Identifier) son nombres o sustantivos en plural y en minúscula para todos los recursos; esto porque el protocolo HTTP ya maneja los verbos que corresponden a las acciones de CRUD (POST, PUT, PATCH, GET, u otros).
- Usar la notación "camello" (camelCase), que es la práctica de escribir frases compuestas de manera que cada palabra en medio de la frase comience con una letra mayúscula (por ejemplo, fraseEnCamelCase). Utilizar esta notación para nombrar recursos, atributos y parámetros.

- Aprovechar la naturaleza jerárquica de la URL siempre que un objeto tenga relación con otro.

```
GET  /tramites/1234/instanciasTramite/4321
```

  Se recomienda no abusar de esta característica y utilizar preferentemente solo un nivel de recurso en el URL.

- Usar el símbolo "?" para filtrar los recursos y "&" para añadir más filtros en el URL.

```
GET  /tramites?idEntidad=2345&estado=pendiente
```

- Usar la palabra clave "buscar" en el URL para ejecutar una búsqueda avanzada (se hace una excepción en este caso para utilizar un verbo en lugar de un nombre debido a que se trata de un procesamiento y no de un proceso de lectura como tal).

```
GET  /buscar?q=nacimiento
```

- Para las paginaciones (en caso de listas con una cantidad alta de datos) se sugiere contar con los parámetros "limite" (cantidad de datos devueltos por el recurso) e "indice" (índice del primer elemento devuelto por el recurso) en el URL.

```
GET  /tramites?limite=10&indice=51
```

Además, se sugiere enviar opcionalmente en la respuesta el header link con información referente a la paginación ("first" la primera página, "prev" la página anterior, "next" la siguiente página y "last" la última página).

```
Link:  <https://servicios.gob.bo/tramites?limite=20&intervalo=0>; rel="first",
<https://servicios.gob.bo/tramites?limite=20&intervalo=40>; rel="prev",
<https://servicios.gob.bo/tramites?limite=20&intervalo=80>; rel="next",
<https://servicios.gob.bo/tramites?limite=20&intervalo=180>; rel="last"
```

- Para ordenar los recursos se utiliza el parámetro "orden"; por defecto los recursos están ordenados ascendentemente. Para ordenar los recursos de forma descendente, se usa el carácter "-".

```
GET  /tramites?orden=estado,-descripcion
```

- En el caso de manejar un límite de solicitudes por cliente (rate-limiting), es necesario enviar en la respuesta una cabecera (header) HTTP extra, con información de la cantidad de solicitudes permitidas, la cantidad de solicitudes restantes y el tiempo (timestamp) en el cual estos límites volverán a su estado inicial

```
X-LimiteSolicitudes-Limite: 100
X-LimiteSolicitudes-Restante: 14
X-LimiteSolicitudes-Reset: 1499875025
```

- Utilizar los códigos HTTP para responder a las consultas realizadas por el cliente. Véase el Anexo 15.3 (Códigos de respuesta HTTP).

- Considerar el cifrado de los parámetros enviados en el URI si estos fueran de alguna manera sensibles. Véase el punto 8.2.3 (Confidencialidad).

15.2.1.2 Validación de los mensajes

Utilizar JSON Schema para validación de mensajes en REST de acuerdo a lo descrito por la IETF[16].

JSON Schema está descrita también en formato JSON; sin embargo, se utiliza para describir la estructura de otros datos. En su núcleo, JSON Schema define los si-

[16] http://json-schema.org/latest/json-schema-core.html

guientes tipos básicos de datos:

- **string**: que se utiliza para cadenas de texto.
- **number o integer**: se utilizan para definir valores numéricos, siendo la principal diferencia que "number" acepta tanto números enteros como números decimales, mientras que "integer" se usa solamente para validar números enteros.
- **object**: se utiliza para validar estructuras JSON anidadas (una estructura dentro de otra).
- **array**: se usa para definir un conjunto de elementos.
- **boolean**: este tipo de dato se utiliza para verificar dos valores especiales, true (valor verdadero) y false (valor falso).
- **null**: este valor especial se utiliza generalmente para representar la ausencia de un valor.

Se recomienda que cada objeto identificado en la semántica tenga asociado y publicado su JSON Schema respectivo.

Por ejemplo, para validar la estructura JSON del Anexo 15.1 utilizar:

```
{
    "title": "Persona",
    "type": "object",
    "properties": {
        "ci": {
            "type": "integer"
        },
        "nombres": {
            "type": "string"
        },
        "apellidos": {
            "type": "string"
        },
        "fechaNacimiento": {
            "type": "string"
        }
    }
}
```

15.2.1.3 Versionamiento

El versionamiento se aplica en la URL. Se podrán tener hasta dos versiones al mismo tiempo.

```
GET  /v1/tramites
GET  /v2/tramites
```

15.2.1.4 Manejo de errores

Todos los problemas que presenten los servicios de interoperabilidad deben ser identificados, de modo que la entidad consumidora de los mismos comprenda los motivos de las fallas.

Para el manejo de errores, se recomienda utilizar la siguiente estructura mínima JSON:

```
{
    "codigo": "Código del error",
    "error": "Descripción detallada del error"
}
```

En caso de que existan muchos errores se puede utilizar la siguiente estructura JSON:

```
{
    "codigo": "Código del error",
    "error": "Descripción detallada del error",
    "errores": [
        {
            "codigo": "identificador del contexto del error",
            "error": "Descripción detallada del error"
        },
        {
            "codigo": "identificador del contexto del error",
            "error": "Descripción detallada del error"
        },
        ...
    ]
}
```

Por ejemplo, en un servicio de inserción de trámites que se hace de forma masiva y que haya presentado un error, la respuesta esperada es:

```
{
   "codigo": "0003",
   "error": "Error en el procesamiento de la inserción masiva de trámites",
   "errores": [
      {
        "codigo": 1568548,
        "error": "El trámite no tiene el atributo monto"
      },
      {
        "codigo": 4894564,
        "error": "El monto del trámite no puede ser negativo"
      },
 {
        "codigo": 8598568,
        "error": "La cuenta no corresponde a la entidad"
      },
     ]
}
```

15.2.1.5 Codificación

La codificación en el encabezado "content-type" del protocolo HTTP se establecerá de la siguiente manera:

```
content-type: application/json; charset=utf-8
```

15.2.2 SOAP

SOAP es un protocolo creado para realizar el intercambio estructurado de datos en un entorno descentralizado (no todos los participantes de la comunicación están en el mismo lugar). Se trata de un protocolo de acceso a servicios basado en estándares que define un conjunto de reglas para estructurar los mensajes en XML. Es independiente del protocolo de transporte, del lenguaje de implementación, de la plataforma y del sistema operativo.

Se recomienda utilizar la versión 1.2 de SOAP[17] siempre que sea posible.

15.2.2.1 Buenas prácticas

Para mantener la consistencia en el desarrollo de los servicios SOAP, se sugiere implementar las siguientes buenas prácticas:

- Las operaciones y mensajes se definen de acuerdo a lo detallado en el archivo WSDL (Web Service Definition Language o lenguaje de definición de servicio web), que tiene la siguiente estructura:

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
    <wsdl:types>
        <!-- definiciones de los tipos de datos (XSD) -->
    </wsdl:types>
    <wsdl:message>
        <!-- definiciones de los datos que se intercambian -->
    </wsdl:message>
    <wsdl:portType>
        <!-- conjunto de operaciones soportadas -->
    </wsdl:portType>
    <wsdl:binding>
        <!-- especificación del protocolo y formato de dato para un portType -->
    </wsdl:binding>
    <wsdl:service>
        <!-- colección de recursos finales (endpoints) -->
    </wsdl:service>
</wsdl:definitions>
```

Es importante hacer notar que usualmente este WSDL es generado de manera automática a partir del código en el lenguaje de programación que se desee utilizar.

El archivo WSDL se tiene que publicar y debe ser accesible por la entidad consumidora.

■

[17] https://www.w3.org/TR/soap12-part1/

- Existen diferentes formas de traducir el WSDL al mensaje SOAP, esto se define en los estilos. Es recomendable utilizar el estilo documento con uso literal (document/literal), ya que todo lo que se encuentra en el cuerpo del mensaje SOAP se define en el esquema lo que permite validarlo. Este estilo cumple con la especificación WS-I para interoperabilidad.

Los estilos de documentos se definen en el elemento <binding> del archivo WSDL como se puede observar a continuación:

```
<wsdl:binding name="TramiteBinding" type="ns:TramitePortType">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
   <wsdl:operation name="obtenerTramite">
      <soap:operation soapAction="urn:obtenerTramite" style="document" />
      <wsdl:input>
         <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
         <soap:body use="literal" />
      </wsdl:output>
   </wsdl:operation>
 </wsdl:binding>
```

- Se recomienda usar notación "camello" (camelCase) para nombrar las operaciones del servicio web.

  Los nombres de las operaciones son verbos descriptivos que denotan acciones a realizar.

- Se recomienda que en el archivo WSDL se tenga operaciones relacionadas a un solo objeto.

- En la implementación de un servicio de interoperabilidad SOAP se recomienda no mantener un estado entre peticiones al servicio (stateless), lo que permite que sea reusable por distintos consumidores.

Para ver una definición WSDL completa junto a un ejemplo de su consumo, revisar los anexos 15.4 (Ejemplo de definición de un WSDL) y 15.5 (Ejemplo de mensaje SOAP de consumo).

15.2.2.2 Validación de los mensajes

La validación de la estructura y las restricciones de contenido de XML se la realiza mediante XSD[18] (XML Schema Definition Language), que es una recomendación de la W3C. La versión actual de esta recomendación es la XML Schema 1.1, que consta de dos partes: XML Schema 1.1 Part 1 Structures y XML Schema 1.1 Part 2 Datatypes.

XSD es el lenguaje de definición de esquemas representado mediante XML. Entre sus principales componentes usados se tiene: definiciones de tipo simple, definiciones de tipo complejo, declaración de atributos y declaración de elementos.

Para las restricciones sobre el contenido de los elementos XML se tienen los siguientes tipos de datos más utilizados:

- string: representa cadenas de caracteres.
- boolean: representa valores lógicos.
- dateTime: representa una instancia de tiempo.
- decimal: representa un conjunto de números reales.
- base64binary: representa datos binarios codificados en base 64.

Para validar el XML del Anexo 15.1 (Tipos de formatos de representación de datos) se puede utilizar el siguiente XSD:

■

[18] https://www.w3.org/XML/Schema

```
<xs:element name="persona">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ci" type="decimal"/>
      <xs:element name="nombres" type="string"/>
      <xs:element name="apellidos" type="string"/>
      <xs:element name="fechaNacimiento" type="dateTime"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Es recomendable que se separe la definición de los tipos de datos del archivo WSDL, permitiendo hacer más legible el archivo WSDL; trabajar el archivo XSD (XML Schema Definition) separadamente y reutilizar esquemas y nombres de espacio (namespaces).

15.2.2.3 Versionamiento

Para el versionamiento en SOAP existen tres opciones que son las más utilizadas:

- Versionamiento de la operación: cuando un servicio que tiene muchas operaciones desea modificar solamente una de ellas; para este caso se recomienda aplicar el versionamiento en el nombre de la operación, lo que previene cambios drásticos en el WSDL.

```
<wsdl:operation name="operacionV1"></wsdl:operation>
<wsdl:operation name="operacionV2"></wsdl:operation>
```

- Versionamiento del servicio: cuando una entidad cambia muchas de las operaciones en un servicio es preferible crear una nueva definición; en este caso se definirá un nuevo WSDL.

  La definición de un nuevo WSDL implica que existirá un nuevo recurso final (endpoint).

- Agregando al XML la propiedad "version", de tal manera que el consumidor

pueda elegir la versión que requiera de acuerdo al valor enviado en el XML.

Se recomienda mantener el tipo de versionamiento que ya se esté utilizando, sin embargo para los nuevos servicios de interoperabilidad se sugiere realizar el versionamiento agregando al XML la propiedad "version".

15.2.2.4 Manejo de errores

Todos los problemas que presenten los servicios de interoperabilidad deben ser identificados, de modo que la entidad consumidora del servicio comprenda el motivo de la falla. Para el tratamiento de errores se recomienda la siguiente estructura XML:

```
<respuesta>
        <codigo>Código del error</codigo>
        <error>Descripción detallada del error</error>
</respuesta>
```

En caso de que existan muchos errores se puede utilizar la siguiente estructura XML:

```
<respuesta>
        <codigo>Código del error</codigo>
        <error>Descripción detallada del error</error>
        <errores>
                <error>
                    <codigo>Identificador del contexto del error</codigo>
                    <descripcion>Descripción detallada del error</descripcion>
                </error>
                <error>
                    <codigo>Identificador del contexto del error</codigo>
                    <descripcion>Descripción detallada del error</descripcion>
                </error>
        ...
        </errores>
</respuesta>
```

A continuación se puede ver un ejemplo del manejo de errores en SOAP:

```
<respuesta>
    <codigo>0003</codigo>
    <error>Error en el procesamiento de la inserción masiva de trámites</error>
    <errores>
        <error>
            <codigo>15561561</codigo>
            <descripcion>El trámite no tiene el atributo monto</descripcion>
        </error>
        <error>
            <codigo>4455621</codigo>
            <descripcion>El monto del trámite no puede ser negativo</descripcion>
        </error>
        <error>
            <codigo>15267548</codigo>
            <descripcion>La cuenta no corresponde a la entidad</descripcion>
        </error>
    </errores>
</respuesta>
```

15.2.2.5 Codificación

Es necesario establecer la codificación en el encabezado <xml> del documento XML de la siguiente manera:

```
<?xml version="1.0" encoding="UTF-8"?>
```

15.2.3 Otras tecnologías

Existen muchas otras tecnologías que pueden utilizarse para interoperabilidad, las principales se mencionarán a continuación:

15.2.3.1 MQTT

Es un protocolo de mensajería liviano, diseñado específicamente para ambientes

donde se necesita que el código fuente no sea demasiado grande y existan problemas de ancho de banda. Es principalmente utilizado para comunicaciones de máquina a máquina y para la Internet de las cosas (IOT - Internet of Things).

Se diferencia de las demás tecnologías por su modelo de publicar/suscribirse, este principio es simple, se "publica" mensajes y todos los que se encuentran "suscritos" lo reciben. El proceso de publicar y suscribirse se realiza a través de un broker[19].

Mosquitto[20] es una implementación de código abierto que implementa este protocolo.

15.2.3.2 AMQP

AMQP es un estándar abierto para el intercambio de mensajes tanto de manera síncrona como asíncrona de acuerdo a la necesidad. AMQP garantiza que los mensajes se enviarán introduciéndolos en una cola y sacándolos solamente cuando la comunicación haya sido exitosa.

AMQP es implementado sobre distintos protocolos (HTTPS o MQTT, por ejemplo).

Se sugiere su uso cuando existen tareas costosas en tiempo de procesamiento para evitar tener una conexión en espera de respuesta en el servicio web. De esta manera la tarea costosa puede realizarse con diferentes recursos que no afecten el proceso principal.

RabbitMQ[21] es la implementación más conocida de AMQP.

15.3 Códigos de respuesta HTTP

Si bien existe una gran cantidad de códigos HTTP, los más utilizados son:

[19] Los brokers son elementos en la red que permiten a las aplicaciones comunicarse intercambiando mensajes definidos formalmente.
[20] https://mosquitto.org/
[21] https://github.com/rabbitmq/rabbitmq-server

Grupo 200 - Éxito

Código	Detalle	Definición
200	OK	Código básico de éxito. Funciona para los casos generales. Usado especialmente en la respuesta exitosa de GET o el contenido actualizado.
201	Created	Indica que el recurso fue creado. Típicamente es la respuesta a solicitudes PUT de creación o POST.
202	Accepted	Indica que la solicitud ha sido aceptada para procesamiento. Típicamente es la respuesta para una llamada a procesamiento asíncrono.
204	No Content	La solicitud ha tenido éxito, pero no hay nada que mostrar. Frecuentemente enviado luego de DELETE exitoso.
206	Partial Content	El recurso devuelto está incompleto. Típicamente usado con recursos paginados.

Grupo 300 - Redirecciones

Código	Detalle	Definición
301	Moved Permanently	El URI solicitado ha sido redireccionado permanentemente a otro recurso. El consumidor debe direccionar estas solicitudes a otro URI.
302	Found	El URI solicitado ha sido redireccionado temporalmente, el consumidor debe seguir pidiendo este recurso en el futuro.

Grupo 400 - Errores

Código	Detalle	Definición
400	Bad Request	Error general para una solicitud que no puede ser procesada (el consumidor no debe repetir la solicitud sin modificarla).
401	Unauthorized	Indica que el consumidor no tiene una identidad definida en el servicio.
403	Forbidden	Indica que el consumidor tiene una identidad definida en el servicio, pero no tiene los permisos para la solicitud que ha realizado.
404	Not Found	El recurso solicitado no existe.

405	Method Not Allowed	O el método no está soportado o lo relacionado a este recurso no tiene el permiso.
406	Not Acceptable	No existe el recurso en el formato solicitado. Por ejemplo, se solicita un recurso en XML pero sólo está disponible en JSON.

Grupo 500 - Error del servidor

Código	Detalle	Definición
500	Internal Server Error	La solicitud parece correcta, pero un problema ha ocurrido en el servidor. El cliente no puede hacer nada al respecto.

15.4 Ejemplo de definición de un WSDL

Definición de un servicio en el archivo WSDL:

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://tramites.gob.bo/" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://tramites.gob.bo/" name="WsTramites">
    <types>
        <xsd:schema>
            <xsd:import namespace="http://tramites.gob.bo/" schemaLocation="https://localhost:8181/WsTramites/WsTramites?xsd=1"/>
        </xsd:schema>
    </types>
    <message name="obtenerTramite">
        <part name="parameters" element="tns:obtenerTramite"/>
    </message>

<message name="obtenerTramiteResponse">
        <part name="parameters" element="tns:obtenerTramiteResponse"/>
    </message>
    <message name="obtenerTramites">
        <part name="parameters" element="tns:obtenerTramites"/>
    </message>
    <message name="obtenerTramitesResponse">
        <part name="parameters" element="tns:obtenerTramitesResponse"/>
    </message>
```

```
<message name="pagarTramite">
  <part name="parameters" element="tns:pagarTramite"/>
</message>
<message name="pagarTramiteResponse">
  <part name="parameters" element="tns:pagarTramiteResponse"/>
</message>
<portType name="WsTramites">
  <operation name="obtenerTramite">
    <input wsam:Action="http://tramites.gob.bo/WsTramites/obtenerTramiteRequest"
message="tns:obtenerTramite"/>

    <output wsam:Action="http://tramites.gob.bo/WsTramites/obtenerTramiteRespon-
se" message="tns:obtenerTramiteResponse"/>
  </operation>
  <operation name="obtenerTramites">
    <input wsam:Action="http://tramites.gob.bo/WsTramites/obtenerTramitesRequest"
message="tns:obtenerTramites"/>
    <output wsam:Action="http://tramites.gob.bo/WsTramites/obtenerTramitesResponse"
message="tns:obtenerTramitesResponse"/>
  </operation>
  <operation name="pagarTramite">
    <input wsam:Action="http://tramites.gob.bo/WsTramites/pagarTramiteRequest" mes-
sage="tns:pagarTramite"/>
```

```
            <output wsam:Action="http://tramites.gob.bo/WsTramites/pagarTramiteResponse"
message="tns:pagarTramiteResponse"/>
        </operation>
    </portType>
    <binding name="WsTramitesPortBinding" type="tns:WsTramites">
        <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
        <operation name="obtenerTramite">
            <soap:operation soapAction=""/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
        <operation name="obtenerTramites">
            <soap:operation soapAction=""/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
        <operation name="pagarTramite">
            <soap:operation soapAction=""/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>

            </output>
```

```
        </operation>
    </binding>
    <service name="WsTramites">
        <port name="WsTramitesPort" binding="tns:WsTramitesPortBinding">
            <soap:address location="https://localhost:8181/WsTramites/WsTramites"/>
        </port>
    </service>
</definitions>
```

Esquema que corresponde a la definición WSDL anterior:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:tns="http://tramites.gob.bo/" xmlns:xs="http://www.w3.org/2001/XMLS-
chema" version="1.0" targetNamespace="http://tramites.gob.bo/">
    <xs:element name="obtenerTramite" type="tns:obtenerTramite"/>
    <xs:element name="obtenerTramiteResponse" type="tns:obtenerTramiteResponse"/>
    <xs:element name="obtenerTramites" type="tns:obtenerTramites"/>
    <xs:element name="obtenerTramitesResponse" type="tns:obtenerTramitesResponse"/>
    <xs:element name="pagarTramite" type="tns:pagarTramite"/>
    <xs:element name="pagarTramiteResponse" type="tns:pagarTramiteResponse"/>
    <xs:complexType name="obtenerTramites">
        <xs:sequence>
            <xs:element name="estado" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="obtenerTramitesResponse">
        <xs:sequence>
            <xs:element name="return" type="tns:tramites" minOccurs="0" maxOccurs="unboun-
ded"/>
        </xs:sequence>
```

```
    </xs:complexType>
    <xs:complexType name="tramites">
        <xs:sequence>
            <xs:element name="codigo" type="xs:string" minOccurs="0"/>
            <xs:element name="codigoEntidad" type="xs:string" minOccurs="0"/>
            <xs:element name="descripcion" type="xs:string" minOccurs="0"/>
            <xs:element name="estado" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="obtenerTramite">
        <xs:sequence>
            <xs:element name="codigo" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="obtenerTramiteResponse">
        <xs:sequence>
            <xs:element name="return" type="tns:tramites" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="pagarTramite">
        <xs:sequence>
            <xs:element name="codigo" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
    <xs:complexType name="pagarTramiteResponse">
        <xs:sequence>
            <xs:element name="return" type="xs:string" minOccurs="0"/>
        </xs:sequence>
    </xs:complexType>
</xs:schema>
```

15.5 Ejemplo de mensaje SOAP de consumo

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ws="ht-
tp://ws.fundempresa.org.bo/">
  <soap:Body>
    <ws:SrvActividades>
      <idContrato>idcontrato</idContrato>
      <keyContrato>245237g34g237sfdgsdf4334</keyContrato>
      <IdMatricula>12345</IdMatricula>
    </ws:SrvActividades>
  </soap:Body>
</soap:Envelope>
```

15.6 Ejemplo de ficha de metadatos de servicio de interoperabilidad

Datos Generales	
Identificador	Serv-10323
Nombre	GastoInversiónWS.registrarGastoInversión
Descripción	Permite realizar la carga de un registro de gasto de inversión en el momento de la formulación y reformulación del Presupuesto General del Estado.
Estado	ACTIVO
Fecha de Inicio de disponibilidad	01/09/15
Fecha de Fin de disponibilidad	
Publicador	Ministerio de Economía y Finanzas Públicas
Fecha de registro	01/05/17
Fecha de última actualización	
Versión	1.0
Documentación	http://economiayfinanzas.gob.bo/servicios/ginv.html
Servicios Relacionados	Serv-9723, Serv-9584
Palabras Clave	Gasto Inversión, Economía
Tipo de Acceso a la información	Confidencial

Restricciones de uso	
Base Legal	Se tiene como base legal las previsiones establecidas en la Constitución Política del Estado referidas al acceso de información. http://www.economiayfinanzas.gob.bo/index.php?opcion=com_contenido&ver=contenido&id=3751&id_item=255
Prerequisitos	El servicio es accesible solamente por convenio con el Viceministerio de Planificación del Desarrollo
Restricciones adicionales	
Contacto Institucional	
Unidad Responsable	Dirección General de Sistemas de Información Financiera
Correo Electrónico	dgsgif@economiayfinanzas.gob.bo
Teléfono	2203434 (Int. 327)
Información Técnica	
Dirección Física del servicio	http://economiayfinanzas.gob.bo/servicios/ginv.wsdl
Tipo	SOAP
Entorno	Producción
Datos de Entrada	Gestión, entidad, dirección administrativa, unidad ejecutora, partida, fuente, organismo financiador, entidad de transferencia e importe
Datos de Salida	Código de éxito o error de inserción
Tipo de Conexión	Http sobre internet
Software Relacionado	Ninguno
Seguridad	
Políticas de Seguridad	El ingreso es restringido por usuario y contraseña, debiendo registrarse previamente el usuario en el SIGEP.
Tipo de autenticación	Basic
Firma Digital	NO
Transporte	HTTPS

15.7 Lista de verificación para la implementación de servicios de interoperabilidad

Los campos marcados con * son obligatorios

1. Sensibilidad de los datos*

☐ Datos sensibles

2. Naturaleza del servicio de interoperabilidad (puede escoger uno o varios)*

a. ☐ Tipo de servicio de lectura/consulta ☐ Tipo de servicio transaccional

b. ☐ Tipo de servicio individual ☐ Tipo de servicio masivo

c. ☐ Tipo de servicio estadístico agregado ☐ Tipo de servicio desagregado

3. Semántica*

☐ Objetos y códigos definidos

4. tipos de servicio web y formatos de datos*

○ REST con JSON ○ SOAP con XML ○ Otras tecnologías

5. Seguridad*

5.1. Seguridad en la capa de transporte (mínimamente utilizar TLS)*

☐ TLS ☐ VPN

5.2. Seguridad de los datos (Autenticación y autorización)*

☐ Basic ☐ Certificados digitales ☐ JWT

☐ OAuth ☐ OpenId

5.3. Seguridad de los datos (Integridad)

☐ Registro centralizado de hashes ☐ Firma Electrónica

☐ Firma Digital ☐ Blockchain

(Si se marcó la opción 1 – datos sensibles, se debe implementar firma electrónica o blockchain)

5.4. Confidencialidad

☐ Cifrado Simétrico ☐ Cifrado Asimétrico

Si se marcó la opción 1 – datos sensibles, se debe implementar un tipo de cifrado)

5.5. Auditoría*

○ Implementa registro de eventos (logs)

6. Disponibilidad

☐ Redundancia

☐ Balanceo de carga (si se tiene una cantidad alta de consumidores)

☐ Cacheado (cuando el tipo del servicio es solo de lectura)

☐ Monitoreo*

7. Accesibilidad*

☐ Entorno de pruebas ☐ Software de consumo

☐ Documentación ☐ Manuales ☐ Portal web

8. Políticas*

☐ Tiene documento de políticas de acceso y uso del servicio

9. Catálogo de servicios*

☐ Se ha llenado la ficha del servicio

16 REFERENCIAS

Consejo para las Tecnologías de Información y Comunicación del Estado Plurinacional de Bolivia. (2016). Punto 8.1. Codificación estándar de caracteres. En Lineamientos para la adecuación y publicación de datos abiertos. (pp. 12-13). Recuperado de **https://www.ctic.gob.bo/wp-content/uploads/bsk-pdf-manager/DATOS-ABIERTOS-v1.0_114.pdf**

Decreto Supremo nº 1793. Artículo 3, Definiciones. Recuperado de **http://www.redipd.org/legislacion/common/legislacion/Bolivia/DS_1793_Telecomunicaciones.pdf**

Fielding, Roy. (2000). CHAPTER 5: Representational State Transfer (REST). En Architectural Styles and the Design of Network-based Software Architectures. Recuperado de **http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm**

Internet Engineering Task Force (IETF). (2015). RFC 7519 - JSON Web Token (JWT). Recuperado de **https://tools.ietf.org/html/rfc7519**

Internet Engineering Task Force (IETF). (2015). RFC 7515 - JSON Web Signature (JWS). Recuperado de **https://tools.ietf.org/html/rfc7515**

Internet Engineering Task Force (IETF). (1999). RFC 2616 - Hypertext Transfer Protocol – HTTP/1.1. Recuperado de **https://tools.ietf.org/html/rfc2616**

Internet Engineering Task Force (IETF). (2012). RFC 6749 - The OAuth 2.0 Authorization Framework. Recuperado de **https://tools.ietf.org/html/rfc6749#section-3.1%29**

Manso, Miguel; Wachowicz, Mónica; Bernabe, Miguel; Sánchez, Almudena; y Rodriguez, A. (2008). Modelo de Interoperabilidad Basado en Metadatos. En V Jornadas Técnicas de la IDE de España JIDEE 2008 IDE, Aplicaciones al Planeamiento y la Gestión del Territorio". Jornadas llevadas a cabo en Tenerife, España.

OpenID Foundation. (2014). OpenID Connect Core 1.0 incorporating errata set. Recuperado de **http://openid.net/specs/openid-connect-core-1_0.html**

World Wide Web Consortium (W3C). (s.f.). SOAP Specifications. Recuperado de **https://www.w3.org/TR/soap/**

World Wide Web Consortium (W3C). (2016). SOAP Security Extensions: Digital Signature. Recuperado de **https://www.w3.org/2008/xmlsec**

World Wide Web Consortium (W3C). (2009). XML Protocol Working Group. Recuperado de **https://www.w3.org/2000/xp/Group/**


AGETIC
agencia de gobierno electrónico y
tecnologías de información y comunicación