

DEFINICIÓN DEL SERVICIO WEB DE COMUNICACIÓN DE PRESCRIPCIONES DEL SISTEMA INFORMÁTICO CENTRAL DE CONTROL DE PRESCRIPCIONES VETERINARIAS



Mayo 2024

Índice

1. Versiones	4
2. Introducción	5
3. Definición del servicio web de comunicación de prescripciones de Presvet.....	6
4. Definición de los métodos del servicio de comunicación de prescripciones de Presvet.....	12
4.1. Obtención del estado de la comunicación.....	12
4.1.1. Esta Activo.....	12
4.1.2. Obtener estado trabajo.....	12
4.1.3. Obtener estado trabajos intervalo	13
4.1.4. Obtener errores por trabajo.....	14
4.2. Comunicación de datos.....	15
4.2.1. Alta prescripción otras formas farmacéuticas	15
4.2.2. Modificación prescripción otras forma farmacéuticas	17
4.2.3. Alta prescripción piensos medicamentosos formulados en base a premezclas que sean antibióticos	19
4.2.4. Modificación prescripción piensos medicamentosos formulados en base a premezclas que sean antibióticos	21
4.2.5. Eliminación de prescripciones.	23
4.3. Datos auxiliares y códigos	24
4.3.1. Obtener códigos verificación definidos.....	24
4.3.2. Obtener códigos incidencia definidos.....	25
4.3.3. Obtener códigos estado proceso.....	27
4.3.4. Obtener especies REGA	28
4.3.5. Obtener relación especies Rega-Nomenclator	28
4.3.6. Obtener prescripciones de un trabajo.....	29
4.3.7. Obtener códigos de unidades de dosificación definidos	30
4.3.8. Obtener códigos clase tratamiento	31
4.3.9. Obtener códigos tipo dispensación antibióticos.....	32
4.3.10. Obtener códigos tipo dispensación premezclas.....	33
5. Recomendaciones de uso	34
6. Definición xml del servicio Web	35
7. Contrato del servicio.	52
8. Definición del contrato de datos del servicio.	53
9. Aclaraciones sobre los estados	61
10. Tabla de especies REGA.....	62

11. Tabla de relación de especies Nomenclátor-Rega.....67

1. Versiones

Versión documento	Modificaciones
Octubre 2019	Versión inicial
Noviembre 2019	Cambio logos
Enero 2020	Corrección de direcciones
Enero 2021	Actualización a especie Rega
Febrero 2022	Actualización de dirección url y cambio MAPAMA por MAPA
Noviembre 2023	Nuevos campos para prescripciones, actualización de los métodos de inserción, actualización y consulta así como creación de nuevos métodos de obtención de maestras, obtenercodigosclasetratamiento, ObtenerCodigosTipoDispensacionAntibioticos y ObtenerCodigosTipoDispensacionPremezclas
Mayo 2024	Actualización de Urls.

2. Introducción

El presente documento describe los distintos métodos que constituyen el Servicio Web que permitirá comunicar al MAPA la información de las prescripciones veterinarias de otras formas farmacéuticas y piensos medicamentosos formulados en base a premezclas que sean antibióticos.

3. Definición del servicio web de comunicación de prescripciones de Presvet.

Mediante este servicio se va a permitir consultar algunas tablas de codificación del sistema útiles, así como, la comunicación de nuevas prescripciones o modificación de prescripciones ya existentes en el sistema. Para acceder al servicio será necesario disponer de un usuario y una clave previamente suministrada y con acceso a REGEUS.

Para el entorno de preproducción las url serán con integra-servicio en vez de servicio.

Ejemplo: <https://integra-servicio.mapa.gob.es/wspressvetcomunicacion/PresVetComunicacionPrescripcion.svc>

La url del acceso al servicio será:

<https://servicio.mapa.gob.es/wspressvetcomunicacion/PresVetComunicacionPrescripcion.svc>

La dirección del fichero de configuración será:

<https://servicio.mapa.gob.es/wspressvetcomunicacion/wSDL/serviciopresvetcomunicacionprescripcion.wSDL>

a. Ejemplo conexión .Net y C#

Y en el cliente .net será necesario establecer la siguiente configuración:

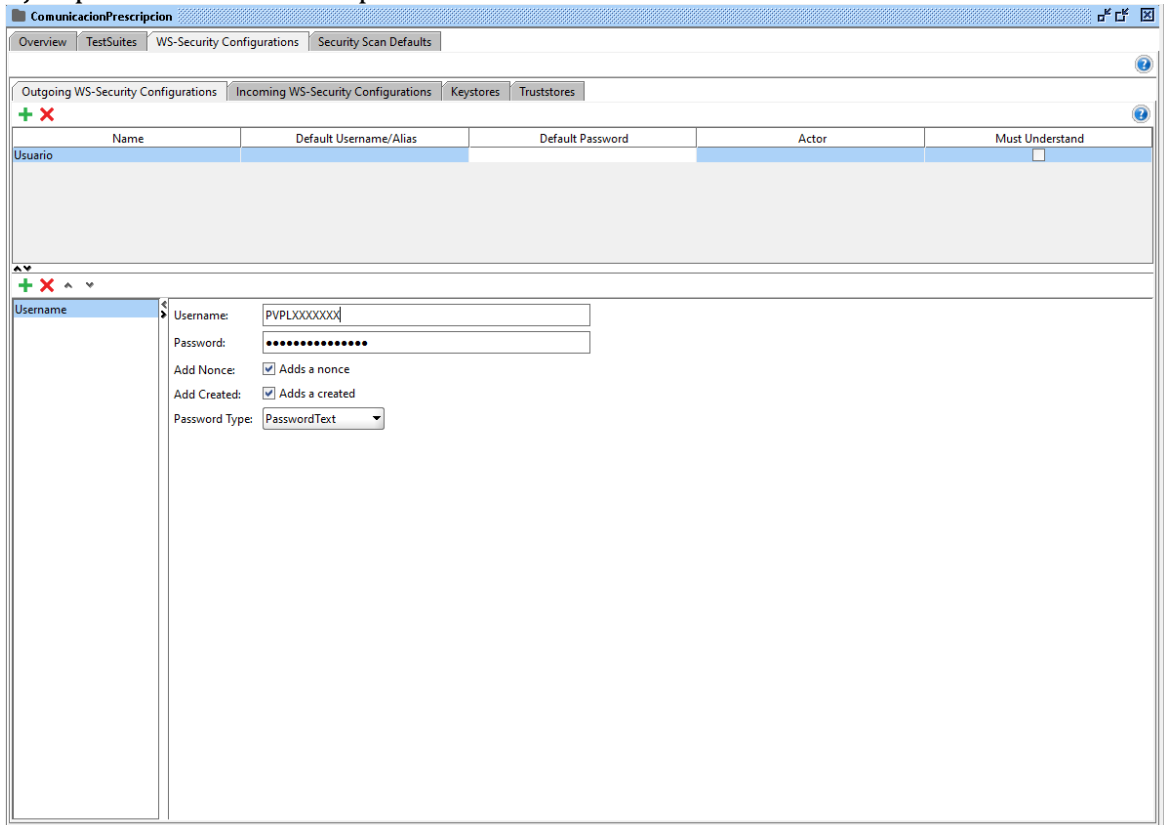
```
<system.serviceModel>
  <bindings>
    <wsHttpBinding>
      <binding name="BasicEndpoint">
        <security mode="TransportWithMessageCredential">
          <transport clientCredentialType="None" />
          <message clientCredentialType="UserName" establishSecurityContext="false" />
        </security>
      </binding>
    </wsHttpBinding>
  </bindings>
  <client>
    <endpoint address="https://servicio.mapa.gob.es/wspressvetcomunicacion/
wspressvetcomunicacion.svc"
      binding="wsHttpBinding" bindingConfiguration="BasicEndpoint"
      contract="PresVetComunicacionPrescripcion.IPresVetComunicacionPrescripcion"
      name="BasicEndpoint" />
  </client>
</system.serviceModel>

static void Main(string[] args)
{
    using (PresVetComunicacionPrescripcionClient comunicacionPrescripcion = new
PresVetComunicacionPrescripcionClient())
    {
        comunicacionPrescripcion.ClientCredentials.UserName.UserName = "usuario";
        comunicacionPrescripcion.ClientCredentials.UserName.Password = "clave";

        System.Net.ServicePointManager.Expect100Continue = false;
    }
}
```

```
string resultado = comunicacionPrescripcion.EstaActivo();  
Console.WriteLine(resultado);  
Console.ReadLine();  
}  
}
```

b. Ejemplo conexión con soap ui



The screenshot shows the SoapUI interface for a project named 'ComunicacionPrescripcion'. The 'Outgoing WS-Security Configurations' tab is active, displaying a table with the following structure:

Name	Default Username/Alias	Default Password	Actor	Must Understand
Usuario				<input type="checkbox"/>

Below the table, the configuration for the 'Usuario' entry is shown:

- Username: PVPLXXXXXXXX
- Password: [Redacted]
- Add Nonce: Adds a nonce
- Add Created: Adds a created
- Password Type: PasswordText

BasicEndpoint

Overview Service Endpoints WSDL Content WS-I Compliance

+ X Assign

Endpoint	Username	Password	Domain	WSS-Type	WSS-Ti...	Outgoin...	Inco...	Mode
https://servicio.mapama.gob.es/wspressvetcomunicacion/PresVetComunicacionPrescripcion.svc	<input type="text"/>					Usuario		COMPLEMENT

SOAP Request 1

https://servicio.mapama.gob.es/wspressvetcomunicacion/PresVetComunicacionPrescripcion.svc

```
1 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:tem="http://tempuri.org/">
2   <soap:Header/>
3   <soap:Body>
4     <tem:EstaActivo/>
5   </soap:Body>
6 </soap:Envelope>
```

Authorization: Basic

Username:

Password:

Domain:

Pre-emptive auth: Use global preference
 Authenticate pre-emptively

Outgoing WSS: UsuarioPro

Incoming WSS:

Auth (Basic) Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)

Request 1

https://servicio.mapama.gob.es/wspresvetcomunicacion/PresVetComunicacionPrescripcion.svc

```
1 <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:tem="http://tempuri.org/">
2   <soap:Header/>
3   <soap:Body>
4     <tem:EstaActivo/>
5   </soap:Body>
6 </soap:Envelope>
```

Enable WS-A addressing:

Must understand: NONE

WS-A Version: 200508

Add default wsa:Action: Add default wsa:Action

Action: http://tempuri.org/PresVetComunicacionPrescripcion/EstaAct

Add default wsa:To: Add default wsa:To

To:

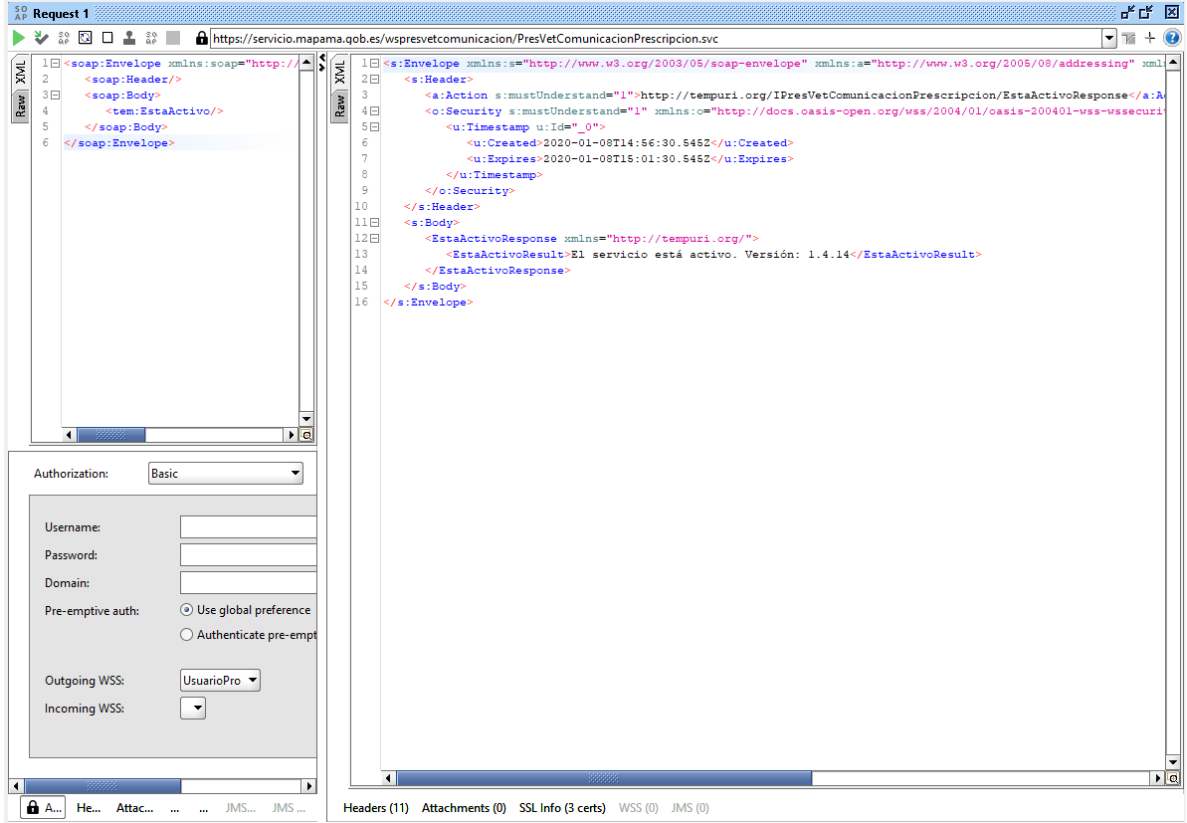
Reply to:

ReplyTo Reference Parameters:

Generate MessageID: Randomly generate MessageID

MessageID:

Auth (Basic) Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Property (0)



The screenshot displays a SOAP client interface with two panels showing XML data. The left panel shows the request XML, and the right panel shows the response XML. Below the XML panels is an authorization configuration section.

Request 1
URL: <https://servicio.mapama.gob.es/wspresvetcomunicacion/PresVetComunicacionPrescripcion.svc>

Raw XML (Request)

```
1 <soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
2   <soap:Header/>
3   <soap:Body>
4     <am:EstaActivo/>
5   </soap:Body>
6 </soap:Envelope>
```

Raw XML (Response)

```
1 <s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope" xmlns:a="http://www.w3.org/2005/08/addressing" xmlns:am="http://tempuri.org/IPresVetComunicacionPrescripcion/EstaActivoResponse">
2   <s:Header>
3     <a:Action s:mustUnderstand="1" http://tempuri.org/IPresVetComunicacionPrescripcion/EstaActivoResponse/><a:Action s:mustUnderstand="1" http://tempuri.org/IPresVetComunicacionPrescripcion/EstaActivoResponse/></a:Action>
4     <o:Security s:mustUnderstand="1" xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-200401-1" xmlns:u="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1">
5       <u:Timestamp u:Id="0">
6         <u:Created>2020-01-08T14:56:30.545Z</u:Created>
7         <u:Expires>2020-01-08T15:01:30.545Z</u:Expires>
8       </u:Timestamp>
9     </o:Security>
10  </s:Header>
11  <s:Body>
12    <EstaActivoResponse xmlns="http://tempuri.org/">
13      <EstaActivoResult>El servicio está activo. Versión: 1.4.14</EstaActivoResult>
14    </EstaActivoResponse>
15  </s:Body>
16 </s:Envelope>
```

Authorization: Basic

Username:
Password:
Domain:
Pre-emptive auth: Use global preference Authenticate pre-emptive
Outgoing WSS:
Incoming WSS:

Headers (11) Attachments (0) SSL Info (3 certs) WSS (0) JMS (0)

4. Definición de los métodos del servicio de comunicación de prescripciones de Presvet

A continuación se detallan los métodos de los que dispone el servicio web para manejo de la información. Al final en los **puntos 5 y 6** se detallan los contratos de datos y operaciones permitidas (operationcontract y datacontract) definidos en el servicio WCF.

4.1. Obtención del estado de la comunicación

4.1.1. Esta Activo.

Este método del servicio se puede utilizar para verificar la conexión al mismo.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetros de entrada:** No tiene parámetros de entrada.
3. **Respuesta:** Devolverá una cadena indicando que el servicio está activo y la versión.

4.1.2. Obtener estado trabajo.

Mediante este método (`ObtenerEstadoTrabajo`) del servicio se obtiene el estado en el que se encuentra un trabajo especificado. Al enviar un conjunto de prescripciones al servicio, se añaden al sistema generando un trabajo, que posteriormente el sistema procesará, este identificador de trabajo generado se recibe al enviar las prescripciones, mediante alguno de los métodos de comunicación de datos de los que dispone el servicio, es ese identificador el que posteriormente podemos usar para consultar en qué punto se encuentra nuestro trabajo.

4. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
5. **Parámetros de entrada:** `IdTrabajo`. Identificador del trabajo a consultar.

Nombre	Tipo	Descripción
<code>IdTrabajo</code>	entero	Identificador del trabajo a consultar.

6. **Respuesta:** Clase con la información solicitada que contendrá el estado del trabajo solicitado. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [EstadoInfo](#).

Nombre	Tipo	Descripción
IdTrabajo	entero	Identificador del trabajo a consultar.
CodEstado	entero	Código del estado del trabajo.
Descripcion	cadena	Descripción del estado.

El servicio dispondrá de un método para obtener dichos estados, se corresponden con:

IdEstado	Descripción
1	Correcto
2	Correcto con incidencias
3	En proceso
4	Rechazado
5	Generado Parcialmente

4.1.3. Obtener estado trabajos intervalo

Mediante este método (`ObtenerEstadoTrabajosIntervalo`) del servicio se obtiene el estado en el que se encuentran los trabajos en el intervalo de fechas establecido. Al enviar un conjunto de prescripciones al servicio, se añaden al sistema generando un trabajo, que posteriormente el sistema procesará. Este identificador de trabajo generado se recibe al enviar las prescripciones mediante el método de comunicación de datos que se haya usado, es ese identificador el que nos devolverá este método con su estado. En este caso todos los que encuentre en el rango de fechas definido.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetro de entrada:** Intervalo de fechas a consultar.

Nombre	Tipo	Descripción
fcInicial	Fecha	Fecha de inicio del intervalo.
fcFinal	Fecha	Fecha de fin del intervalo.

3. **Respuesta:** Un listado de la clase que especifica el trabajo y sus estados. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [TrabajoInfo](#)

Nombre	Tipo	Descripción
IdTrabajo	entero	Identificador del trabajo a consultar.
CodEstado	entero	Código del estado del trabajo.
Descripcion	cadena	Descripción del estado.

El servicio dispondrá de un método para obtener dichos estados, se corresponden con:

IdEstado	Descripcion
1	Correcto
2	Correcto con incidencias
3	En proceso
4	Rechazado
5	Generado Parcialmente

4.1.4. Obtener errores por trabajo

Mediante este método (`ObtenerErroresPorTrabajo`) del servicio se obtienen los errores que ha generado la comunicación, si los tuviera. Con errores no se incorporaran las prescripciones al sistema, será necesario corregirlos y volver a enviar la información.

Requisitos: El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.

4. **Parámetro de entrada:** Intervalo de fechas a consultar.

Nombre	Tipo	Descripción
idTrabajo	entero	Identificador del trabajo del que se quiere obtener su listado de errores.

5. **Respuesta:** Un listado de la clase que especifica de errores. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [ErroresTrabajoInfo](#)

Nombre	Tipo	Descripción
IdTrabajo	entero	Identificador del trabajo a consultar.
Fila	entero	Se corresponde con la fila de trabajo enviado.
numeroReceta	cadena	Número propio del usuario
codigoPresvet	cadena	Código Presvet si lo tuviera.
errores	Listado cadenas	Listado de los errores del número de receta.

4.2. Comunicación de datos.

4.2.1. Alta prescripción otras formas farmacéuticas

Mediante este método ([AltaPrescripcionAntibiotico](#)) se enviarán un conjunto de altas de prescripciones de otras forma farmacéuticas, el sistema generará un trabajo para procesar el conjunto de prescripciones y le asignará un identificador que el método devolverá, para que posteriormente si se necesita consultar el estado de proceso en el que se encuentra dicho trabajo.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetro de entrada:** Un listado de las prescripciones de otras formas farmacéuticas a añadir al sistema. Se especifica la clase de prescripción en el Anejo como [PrescripcionAntobioticoAlta](#).

La estructura sería una lista donde cada elemento se correspondería con el siguiente formato.

Nombre	Tipo	Descripción
Explotación	Cadena(14)	Código de identificación de explotación ganadera REGA (ES + 12 dígitos).
NIF	Cadena(10)	NIF del prescriptor.
Nombre	Cadena(50)	
PrimerApellido	Cadena(55)	
SegundoApellido	Cadena(55)	
NúmeroColegiado	Cadena(7)	PPNNNNN Prov+nºcolegiación
Fecha Prescripción	Fecha	Con formato dd/mm/aaaa
CódigoNacional	Numérico	Código nacional obtenido de la base de datos oficial de la AEMPS. Nomenclátor.
EspecieRega	Numérico	Especie REGA.
NúmeroEnvases	Decimal(12.4)	Será un campo numérico con cuatro decimales
NúmeroReceta	Cadena(50)	
Excepcional	Cadena(1)	Indicará si es excepcional o no. S o N
IdClaseTratamiento	Numérico	Código de clase de tratamiento aplicada.
IdTipoDispensacion	Numérico	Código del tipo de dispensación aplicado.
DuracionTratamiento	Decimal(4,1)	Duración del tratamiento en días.
PorcentajeEnvaseTotal	Decimal(4,1)	Porcentaje total del número de envases.

3. **Respuesta:** Una clase con la información de respuesta, que se corresponderá con el id del trabajo asignado por el sistema a la petición de alta, indicará si se ha producido un error y su descripción. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [ResultadoInfo](#). Los posibles errores que se indiquen en la respuesta, serán referentes a comunicaciones o problemas en la comunicación del listado de prescripciones, no de contenido de datos, que se validarán posteriormente.

Nombre	Tipo	Descripción
IdTrabajo	Entero	Identificador del trabajo a consultar.
Error	Lógico	Campo lógico que indica si se ha producido error en la comunicación.
DescripcionError	Cadena	Descripción del error si se ha producido.

- En caso de enviar los mismos datos en los campos: Explotación REGA, fecha prescripción, NIF, número de receta, código nacional, especie, nº de envases que una prescripción ya comunicada con anterioridad, para evitar duplicados el servicio web, impedirá la grabación en el sistema y lo marcará como **generado parcialmente**, teniendo que consultar los métodos correspondientes (**ObtenerErroresPorTrabajo**, **ObtenerPrescripcionesPorTrabajo**) salvo en el caso de que todas las prescripciones estén duplicadas que se marcará como **rechazado** teniendo que consultar los mismos con el método correspondiente (**ObtenerErroresPorTrabajo**)

4.2.2. Modificación prescripción otras forma farmacéuticas

Mediante este método (**ModificacionPrescripcionAntibiotico**) se enviarán las modificaciones de prescripciones de otras formas farmacéuticas. Es obligatorio que lleven código de Presvet ya que son modificaciones. El sistema generará un trabajo para este conjunto de prescripciones enviadas y le asignará un identificador que el método devolverá para que posteriormente si se necesita consultar el estado de proceso en el que se encuentra dicho trabajo.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetro de entrada:** Un listado de las prescripciones de otras formas farmacéuticas a modificar. Se especifica la clase de prescripción en el Anejo como [PrescripcionAntibioticoModificacion](#)

La estructura sería una lista donde cada elemento se correspondería con el siguiente formato

Nombre	Tipo	Descripción
CodigoPresvet	Cadena(19)	Código de prescripción proporcionado por el sistema de Presvet.
Explotación	Cadena(14)	Código de identificación de explotación ganadera REGA (ES + 12 dígitos).
NIF	Cadena(10)	NIF del prescriptor.
Nombre	Cadena(50)	
PrimerApellido	Cadena(55)	
SegundoApellido	Cadena(55)	
NúmeroColegiado	Cadena(7)	PPNNNNN Prov+nºcolegiación
CódigoNacional	Numérico	Código nacional obtenido de la base de datos oficial de la AEMPS. Nomenclátor.
EspecieRega	Numérico	Especie REGA.
NúmeroEnvases	Decimal(12.4)	Será un campo numérico con cuatro decimales
NúmeroReceta	Cadena(50)	
Excepcional	Cadena(1)	Indicará si es excepcional o no. S o N
IdClaseTratamiento	Numérico	Código de clase de tratamiento aplicada.
IdTipoDispensacion	Numérico	Código del tipo de dispensación aplicado.
DuracionTratamiento	Decimal(4,1)	Duración del tratamiento en días.
PorcentajeEnvaseTotal	Decimal(4,1)	Porcentaje total del número de envases.

3. **Respuesta:** Una clase con la información de respuesta, que se corresponderá con el id del trabajo asignado por el sistema a la petición de alta, indicará si se ha producido un error y su descripción. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [ResultadoInfo](#). Los posibles errores que se indiquen en la respuesta, serán referentes a comunicaciones o problemas en la comunicación del listado de prescripciones, no de contenido de datos, que se validarán posteriormente.

Nombre	Tipo	Descripción
IdTrabajo	Entero	Identificador del trabajo a consultar.
Error	Lógico	Campo lógico que indica si se ha producido error en la comunicación.
DescripcionError	Cadena	Descripción del error si se ha producido.

- En caso de enviar los mismos datos en los campos: Explotación REGA, NIF, número de receta, código nacional, especie, nº de envases que una prescripción ya comunicada con anterioridad, para evitar duplicados el servicio web, impedirá la grabación en el sistema y lo marcará como **generado parcialmente**, teniendo que consultar los métodos correspondientes (**ObtenerErroresPorTrabajo**, **ObtenerPrescripcionesPorTrabajo**) salvo en el caso de que todas las prescripciones estén duplicadas que se marcará como **rechazado** teniendo que consultar los mismos con el método correspondiente (**ObtenerErroresPorTrabajo**)

4.2.3. Alta prescripción piensos medicamentosos formulados en base a premezclas que sean antibióticos

Mediante este método (**AltaPrescripcionPremezclaAntibiotico**) se enviarán las altas de prescripciones de piensos medicamentosos formulados en base a premezclas que sean antibióticos. El sistema generará de forma automática un trabajo para el posterior procesado y le asignará un identificador único, este identificar lo devolverá el servicio para su posterior consulta si fuera necesario.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetro de entrada:** Un listado de las prescripciones de piensos medicamentosos formulados en base a premezclas que sean antibióticos a añadir al sistema. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [PrescripcionPremezclaAlta](#)

La estructura sería una lista donde cada elemento se correspondería con el siguiente formato

Nombre	Tipo	Descripción
Explotación REGA	Cadena(14)	Código de identificación de explotación ganadera REGA (ES + 12 dígitos).
NIF	Cadena(10)	NIF del prescriptor.
Nombre	Cadena(50)	
Primer Apellido	Cadena(55)	
Segundo Apellido	Cadena(55)	
Número Colegiado	Cadena(7)	PPNNNNN Prov+nºcolegiación
Fecha Prescripción	Fecha	Con formato dd/mm/aaaa
Código Nacional	Numérico	Código nacional obtenido de la base de datos oficial de la AEMPS. Nomenclátor.
EspecieRega	Numérico	Especie REGA.
Cantidad pienso	Decimal(12,4)	Será un campo numérico con cuatro decimales que indica la cantidad de pienso medicamentoso en Kg.
Número Receta	Cadena(50)	
Excepcional	Cadena(1)	Indicará si es excepcional o no. S o N
Dosificacion	Decimal(12,4)	Dosificación de la premezcla de pienso medicamentoso
UnidadDosificacion	Numérico	Unidad de dosificación de la premezcla.
IdClaseTratamiento	Numérico	Código de clase de tratamiento aplicada.
IdTipoDispensacion	Numérico	Código del tipo de dispensación aplicado. En premezclas solo puede tipo "Dispensación".
DuracionTratamiento	Decimal(4,1)	Duración del tratamiento en días.

3. **Respuesta:** Una clase con la información de respuesta, que se corresponderá con el id del trabajo asignado por el sistema a la petición de alta, indicará si se ha producido un error y su descripción. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [ResultadoInfo](#). Los posibles errores que se indiquen en la respuesta, serán referentes a comunicaciones o problemas en la comunicación del listado de prescripciones, no de contenido de datos, que se validarán posteriormente.

Nombre	Tipo	Descripción
IdTrabajo	Entero	Identificador del trabajo a consultar.
Error	Lógico	Campo lógico que indica si se ha producido error en la comunicación.
DescripcionError	Cadena	Descripción del error si se ha producido.

- En caso de enviar los mismos datos en los campos: Explotación REGA, fecha prescripción, NIF, número de receta, código nacional, especie, cantidad de pienso medicamentoso que una prescripción ya comunicada con anterioridad, para evitar duplicados el servicio web, impedirá la grabación en el sistema y lo marcará como **generado parcialmente**, teniendo que consultar los métodos correspondientes (**ObtenerErroresPorTrabajo**, **ObtenerPrescripcionesPorTrabajo**) salvo en el caso de que todas las prescripciones estén duplicadas que se marcará como **rechazado** teniendo que consultar los mismos con el método correspondiente (**ObtenerErroresPorTrabajo**)

4.2.4. Modificación prescripción piensos medicamentosos formulados en base a premezclas que sean antibióticos

Mediante este método (**ModificacionPrescripcionPremezclaAntibiotico**) se enviarán las modificaciones de prescripciones de piensos medicamentosos formulados en base a premezclas que sean antibióticos. Es obligatorio que lleven código de Presvet ya que son modificaciones.

Cuando se envíe el conjunto de prescripciones a modificar el sistema generará un trabajo y le asignará un identificador numérico que mediante los métodos de que dispone el servicio se podrá consultar su estado.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetro de entrada:** Un listado de las prescripciones de piensos medicamentosos formulados en base a premezclas que sean antibióticos a modificar. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [PrescripcionPremezclaModificacion](#)

Nombre	Tipo	Descripción
CodigoPresvet	Cadena(19)	Código de prescripción proporcionado por el sistema de Presvet
ExplotaciónREGA	Cadena(14)	Código de identificación de explotación ganadera REGA (ES + 12 dígitos).
NIF	Cadena(10)	NIF del prescriptor.
Nombre	Cadena(50)	
PrimerApellido	Cadena(55)	
SegundoApellido	Cadena(55)	
NúmeroColegiado	Cadena(7)	PPNNNNN Prov+nºcolegiación
CódigoNacional	Numérico	Código nacional obtenido de la base de datos oficial de la AEMPS. Nomenclátor.
EspecieRega	Numérico	Especie REGA.
Cantidadpienso	Decimal(12,4)	Será un campo numérico con cuatro decimales que indica la cantidad de pienso medicamentoso en Kg.
NúmeroReceta	Cadena(50)	
Excepcional	Cadena(1)	Indicará si es excepcional o no. S o N
Dosificacion	Decimal(12,4)	Dosificación de la premezcla de pienso medicamentoso
UnidadDosificacion	Númerico	Unidad de dosificación de la premezcla
IdClaseTratamiento	Numérico	Código de clase de tratamiento aplicada.
IdTipoDispensacion	Numérico	Código del tipo de dispensación aplicado. En premezclas solo puede tipo "Dispensación".
DuracionTratamiento	Decimal(4,1)	Duración del tratamiento en días.

3. **Respuesta:** Una clase con la información de respuesta, que se corresponderá con el id del trabajo asignado por el sistema a la petición de alta, indicará si se ha producido un error y su descripción. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [ResultadoInfo](#). Los posibles errores que se indiquen en la respuesta, serán referentes a comunicaciones o problemas en la comunicación del listado de prescripciones, no de contenido de datos, que se validarán posteriormente.

Nombre	Tipo	Descripción
IdTrabajo	Entero	Identificador del trabajo a consultar.
Error	Lógico	Campo lógico que indica si se ha producido error en la comunicación.
DescripcionError	Cadena	Descripción del error si se ha producido.

- En caso de enviar los mismos datos en los campos: Explotación REGA, NIF, número de receta, código nacional, especie, cantidad de pienso medicamentoso que una prescripción ya comunicada con anterioridad, para evitar duplicados el servicio web, impedirá la grabación en el sistema y lo marcará como **generado parcialmente**, teniendo que consultar los métodos correspondientes (**ObtenerErroresPorTrabajo**, **ObtenerPrescripcionesPorTrabajo**) salvo en el caso de que todas las prescripciones estén duplicadas que se marcará como **rechazado** teniendo que consultar los mismos con el método correspondiente (**ObtenerErroresPorTrabajo**)

4.2.5. Eliminación de prescripciones.

Mediante este método (**EliminarPrescripciones**) se enviarán los códigos de prescripciones de las prescripciones que se quieren eliminar del sistema.

Cuando se envíe el conjunto de prescripciones a eliminar el sistema generará un trabajo y le asignará un identificador numérico que mediante los métodos de que dispone el servicio se podrá consultar su estado.

- Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
- Parámetro de entrada:** Un listado de los códigos Presvet de las prescripciones que se desea eliminar

Nombre	Tipo	Descripción
CodigoPresvet	Cadena(19)	Código de prescripción proporcionado por el sistema de Presvet

- Respuesta:** Un clase con la información de respuesta, que se corresponderá con el id del trabajo asignado por el sistema a la petición de alta, indicará si se ha producido un error y su descripción. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `ResultadoInfo`. Los posibles errores que se indiquen en la respuesta, serán referentes a comunicaciones o problemas en la comunicación del listado de prescripciones, no de contenido de datos, que se validarán posteriormente.

Nombre	Tipo	Descripción
IdTrabajo	Entero	Identificador del trabajo.
Error	Lógico	Campo lógico que indica si se ha producido error en la comunicación.
DescripcionError	Cadena	Descripción del error si se ha producido.

4.3. Datos auxiliares y códigos

4.3.1. Obtener códigos verificación definidos

Mediante este método (`ObtenerCodigosVerificacionDefinidos`) se obtienen los códigos de error definidos en el sistema para la verificación. Estos códigos de verificación están referidos al trabajo de proceso de las prescripciones enviadas. Para obtener el error específico de cada prescripción es necesario acceder a la descripción del fichero donde se indicará en la línea correspondiente el problema detectado, si lo hubiera.

- Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
- Parámetro de entrada:** No hay parámetros de entrada
- Respuesta:** Un listado de códigos de verificación definidos. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `CodigoErrorVerificacionInfo`

Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
CodigoErrorVerificadion	Entero	Identificador del error de verificación.
Descripcion	Cadena	Descripción del error.

Los códigos de resultado de verificación son:

CodigoErrorVerificcion	Descripcion
1	La verificación de tipo de datos contiene errores.
2	La verificación de datos contiene errores.
3	Correcto tipo de datos
4	Correcto
5	Generado Pacialmente

4.3.2. Obtener códigos incidencia definidos

Mediante este método (**ObtenerCodigosIncidenciaDefinidos**) se obtienen los códigos de incidencia definidos en el sistema.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
2. **Parámetro de entrada:** No hay parámetros de entrada.
3. **Respuesta:** Un listado de códigos de incidencia definidos. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [CodigosIncidenciaInfo](#)

Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
CodigoIncidencia	Entero	Identificador del código de incidencia.
Descripcion	Cadena	Descripción de la incidencia.

Los códigos de incidencia definidos actualmente en el sistema son:

CódigoIncidencia	Descripción
2	La explotación REGA esta inactiva o es baja.
3	Excedido el plazo de notificación de la prescripción.
4	La especie escogida no está indicada para el medicamento seleccionado.
6	La especie seleccionada no existe en la explotación REGA indicada.

Código de incidencia 2: La explotación REGA esta inactiva o de baja. Si desde REGA se informa que la explotación no está activa, se permitirá registrar datos, pero se marcarán con esta incidencia.

Código de incidencia 3: Excedido el plazo de notificación de la prescripción. Si las prescripciones enviadas tienen una diferencia de más de 30 días entre la fecha de grabación y la fecha de comunicación, en el sistema se genera una incidencia.

Código de incidencia 4: La especie escogida no está indicada para el medicamento seleccionado. Después de realizar el cruce de la especie rega que se ha indicado en la prescripción con la correspondiente en nomenclator, esta no se encuentra como especie del medicamento.

Código de incidencia 6: La especie seleccionada no existe en la explotación REGA indicada. En este caso se genera una incidencia ya que se ha insertado una especie que no corresponde a ninguna de las que tiene la explotación REGA seleccionada.

4.3.3. Obtener códigos estado proceso

Mediante este método (`ObtenerCodigosEstadoProcesoTrabajo`) se obtienen los códigos de estado de proceso de trabajos definidos en el sistema.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso
2. **Parámetro de entrada:** No hay parámetros de entrada.
3. **Respuesta:** Un listado de códigos de estado de proceso de trabajo. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `EstadoTrabajoInfo`

Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
IdEstado	Entero	Identificador del estado de trabajo.
Descripcion	Cadena	Descripción del estado en el que puede estar un trabajo.

Los códigos de estado en que se puede encontrar un trabajo son:

IdEstado	Descripcion
1	Correcto
2	Correcto con incidencias
3	En proceso
4	Rechazado
5	Generado Parcialmente

4.3.4. Obtener especies REGA

Mediante este método (`ObtenerEspeciesRega`) se obtienen las especies Rega definidas en el sistema.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso
2. **Parámetro de entrada:** No hay parámetros de entrada.
3. **Respuesta:** Un listado de especies. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `EspecieInfo`

Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
IdEspecie	Entero	Identificador de la especie de REGA.
Nombre	Cadena	Nombre de la especie REGA.

4.3.5. Obtener relación especies Rega-Nomenclator

Mediante este método (`ObtenerRelacionEspeciesRegaNomenclator`) se obtienen las especies Rega definidas en el sistema.

4. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso
5. **Parámetro de entrada:** No hay parámetros de entrada.
6. **Respuesta:** Un listado de especies. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `EspecieRegaNomenclatorInfo`

Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
IdEspecieRega	Entero	Identificador de la especie de Nomenclátor.
NombreEspecieRega	Cadena	Nombre de la especie Nomenclátor.
EspeciesNomenclator	Lista <code>EspecieInfo</code>	Listado de especies Rega y sus correspondencias de especies Nomenclátor.

4.3.6. Obtener prescripciones de un trabajo

Mediante este método (`ObtenerPrescripcionesPorTrabajo`) se obtienen las prescripciones que se han grabado en el sistema del trabajo que se haya indicado.

- Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso
- Parámetro de entrada:** El identificador del trabajo del que se quiere obtener las prescripciones grabadas en el sistema.
- Respuesta:** Un listado de prescripciones. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `PrescripcionResultInfo`
Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
CodigoPresvet	Cadena(19)	Código de prescripción proporcionado por el sistema de Presvet.
Explotación	Cadena(14)	Código de identificación de explotación ganadera REGA (ES + 12 dígitos).
NIF	Cadena(10)	NIF del prescriptor.
Nombre	Cadena(50)	
Primer Apellido	Cadena(55)	
Segundo Apellido	Cadena(55)	
Número Colegiado	Cadena(7)	PPNNNNN Prov+nºcolegiación
Código Nacional	Numérico	Código nacional obtenido de la base de datos oficial de la AEMPS. Nomenclátor.
EspecieRega	Numérico	Especie Rega según la fecha de comunicación.
EspecieNomenclator	Numérico	Especie nomenclátor según la fecha de comunicación de la prescripción.
Número Envases	Decimal(12.4)	Será un campo numérico con 4 decimales
Número Receta	Cadena(50)	
Excepcional	Cadena(1)	Indicará si es excepcional o no. S o N
fechaPrescripcion	fecha	Fecha de la prescripción
Incidencias	Listado cadenas	Listado de las posibles incidencias de la prescripción.

CantidadPienso	Decimal(12,4)	Será un campo numérico con cuatro decimales que indica la cantidad de pienso medicamentoso en Kg.
ClaseTratamiento	ClaseTratamientoInfo	Objeto clase tratamiento que tiene el id y el valor de clase tratamiento, si lo tuviera.
TipoDispensacion	TipoDispensacionInfo	Objeto tipo dispensación que tiene el id y el valor de tipo dispensacion, si lo tuviera.
DuracionTratamiento	Decimal(4,1)	Duración del tratamiento en días si lo tuviera.
PorcentajeEnvaseTotal	Decimal(4,1)	Porcentaje total del número de envases si lo tuviera.

4.3.7. Obtener códigos de unidades de dosificación definidos

Mediante este método se obtienen las unidades de dosificación y sus códigos definidos en el sistema.

1. **Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso
2. **Parámetro de entrada:** No hay parámetros de entrada.
3. **Respuesta:** Un listado de las unidades. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [UnidadDosificacionInfo](#)

Cada elemento de la lista tendrá la siguiente estructura:

Nombre	Tipo	Descripción
IdUnidadDosificacion	Entero	Identificador del código de unidad.
Descripcion	Cadena	Descripción de la unidad.

Los códigos de las unidades definidas actualmente son:

IdUnidadDosificacion	Descripcion	
1	Kg/Tn	Kilogramos de premezcla por tonelada de pienso medicamentoso.
2	g/Kg	Gramos de premezcla por Kilogramo de pienso medicamento.
3	ppm	Partes por millón. Partes de premezcla por millón de pienso medicamentoso.

4.3.8. Obtener códigos clase tratamiento

Mediante este método (`ObtenerCodigosClaseTratamiento`) se obtienen las clases de tratamiento que están definidas en el sistema.

- 1. Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
- 2. Parámetro de entrada:** No hay parámetros de entrada.
- 3. Respuesta:** Un listado de clases de tratamiento. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `ClaseTratamientoInfo`

Cada elemento de la lista tendrá la siguiente estructura:

Los valores que actualmente están en el sistema para clases de tratamiento son:

Id	Descripción
1	Terapéutico
2	Metafiláctico
3	Profiláctico

4.3.9. Obtener códigos tipo dispensación antibióticos

Mediante este método (`ObtenerCodigosTipoDispensacionAntibioticos`) se obtienen los tipos de dispensación que están definidas en el sistema, para las prescripciones de tipo otras formas farmacéuticas.

- 1. Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
- 2. Parámetro de entrada:** No hay parámetros de entrada.
- 3. Respuesta:** Un listado de clases de tratamiento. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como [TipoDispensacionInfo](#)

Cada elemento de la lista tendrá la siguiente estructura:

Los valores que actualmente están en el sistema para tipos de dispensación para otras formas farmacéuticas son:

Id	Descripción
1	Dispensación
2	No Dispensación
3	Botiquín

4.3.10. Obtener códigos tipo dispensación premezclas

Mediante este método (`ObtenerCodigosTipoDispensacionAntibioticos`) se obtienen los tipos de dispensación que están definidas en el sistema, para las prescripciones de tipo otras formas farmacéuticas.

- 1. Requisitos:** El usuario debe cumplir los requisitos de seguridad establecidos por el MAPA y disponer de usuario y clave de acceso.
- 2. Parámetro de entrada:** No hay parámetros de entrada.
- 3. Respuesta:** Un listado de clases de tratamiento. Se especifica en el punto 6 (Definición del contrato de datos del servicio) como `TipoDispensacionInfo`

Cada elemento de la lista tendrá la siguiente estructura:

Los valores que actualmente están en el sistema para tipos de dispensación para premezclas son:

Id	Descripción
1	Dispensación

5. Recomendaciones de uso

Recomendaciones de uso del servicio web de comunicación de prescripciones.

Con el fin de evitar que el sistema se ralentice al realizar todos los usuarios la comunicación de prescripciones a la vez, se recomienda que la comunicación de las mismas se vaya realizando periódicamente a lo largo del mes y no se espere al último día de plazo de comunicación de las prescripciones (30 días tras las prescripciones) para enviar la información, dado que eso puede provocar retrasos en el proceso de la misma y puede que algunas prescripciones entren fuera del plazo establecido.

También es recomendable, que, en vez de enviar una prescripción por trabajo, siempre que sea posible según el negocio de cada aplicación cliente, se agrupen varias en el mismo trabajo, sobre todo si se envían con los plazos justos, para evitar lo anteriormente expuesto.

Para información de todos los usuarios, el sistema de proceso de las prescripciones enviadas por servicio web, funciona en dos pasos, en un paso inicial que es el propio servicio web de comunicación, el sistema registra un trabajo guardando los datos enviados sin procesar, devolviendo un identificador del trabajo. Este trabajo se procesará posteriormente en un proceso interno del sistema, generando las nuevas prescripciones, modificando, o borrando según la opción elegida. Hasta que este proceso termine, cuyo tiempo de ejecución es variable en función del volumen de datos de cada trabajo y de la carga del sistema en función de los trabajos simultáneos que se le hayan enviado, no tiene sentido estar haciendo llamadas para obtener la información del estado de cada trabajo, es recomendable esperar un tiempo prudencial antes de realizar la petición de información.

Todas las peticiones que se hagan al servicio web, sean de altas, modificación, eliminación, consulta de trabajos, de prescripciones, etc., acceden a través del sistema de validación del Ministerio (REGEUS), se recomienda el uso racional de las llamadas al sistema.

6. Definición xml del servicio Web

A continuación se muestra la definición xml del servicio web:

```
<wsdl:definitions xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:wsx="http://schemas.xmlsoap.org/ws/2004/09/mex"
xmlns:wsap="http://schemas.xmlsoap.org/ws/2004/08/addressing/policy"
xmlns:msc="http://schemas.microsoft.com/ws/2005/12/wsdl/contract"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns="http://tempuri.org/"
xmlns:wsa10="http://www.w3.org/2005/08/addressing" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" name="PresVetComunicacionPrescripcion"
targetNamespace="http://tempuri.org/">
  <wsp:Policy wsu:Id="BasicEndpoint_policy">
    <wsp:ExactlyOne>
      <wsp:All>
        <sp:TransportBinding xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
          <wsp:Policy>
            <sp:TransportToken>
              <wsp:Policy>
                <sp:HttpsToken RequireClientCertificate="false"/>
              </wsp:Policy>
            </sp:TransportToken>
            <sp:AlgorithmSuite>
              <wsp:Policy>
                <sp:Basic256/>
              </wsp:Policy>
            </sp:AlgorithmSuite>
            <sp:Layout>
              <wsp:Policy>
                <sp:Strict/>
              </wsp:Policy>
            </sp:Layout>
            <sp:IncludeTimestamp/>
          </wsp:Policy>
        </sp:TransportBinding>
        <sp:SignedSupportingTokens xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
          <wsp:Policy>
            <sp:UsernameToken
sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipient">
              <wsp:Policy>
                <sp:WssUsernameToken10/>
              </wsp:Policy>
            </sp:UsernameToken>
          </wsp:Policy>
        </sp:SignedSupportingTokens>
        <sp:Wss11 xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
          <wsp:Policy/>
        </sp:Wss11>
        <sp:Trust10 xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
          <wsp:Policy>
            <sp:MustSupportIssuedTokens/>
            <sp:RequireClientEntropy/>
            <sp:RequireServerEntropy/>
          </wsp:Policy>
        </sp:Trust10>
        <wsaw:UsingAddressing/>
      </wsp:All>
    </wsp:ExactlyOne>
  </wsp:Policy>
</wsdl:types>
```

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
targetNamespace="http://tempuri.org/">
  <xs:import
namespace="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"/>
  <xs:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
  <xs:element name="EstaActivo">
    <xs:complexType>
      <xs:sequence/>
    </xs:complexType>
  </xs:element>
  <xs:element name="EstaActivoResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="EstaActivoResult" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ObtenerEstadoTrabajo">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="idTrabajo" type="xs:int"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ObtenerEstadoTrabajoResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element
xmlns:q1="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerEstadoTrabajoResult" nillable="true" type="q1:EstadoInfo"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="AltaPrescripcionAntibiotico">
    <xs:complexType>
      <xs:sequence>
        <xs:element
xmlns:q2="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="prescripciones" nillable="true" type="q2:ArrayOfPrescripcionAntibioticoAlta"/>
        <xs:element minOccurs="0" name="Excepcional" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="AltaPrescripcionAntibioticoResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element
xmlns:q3="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="AltaPrescripcionAntibioticoResult" nillable="true" type="q3:ResultadoInfo"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ModificacionPrescripcionAntibiotico">
    <xs:complexType>
      <xs:sequence>
        <xs:element
xmlns:q4="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="prescripciones" nillable="true" type="q4:ArrayOfPrescripcionAntibioticoModificacion"/>
        <xs:element minOccurs="0" name="Excepcional" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ModificacionPrescripcionAntibioticoResponse">
    <xs:complexType>
      <xs:sequence>
```

```

                <xs:element
xmlns:q5="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ModificacionPrescripcionAntibioticoResult" nillable="true" type="q5:ResultadoInfo"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="AltaPrescripcionPremezclaAntibiotico">
            <xs:complexType>
                <xs:sequence>
                    <xs:element
xmlns:q6="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="prescripciones" nillable="true" type="q6:ArrayOfPrescripcionPremezclaAlta"/>
                    <xs:element minOccurs="0" name="Excepcional" type="xs:boolean"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="AltaPrescripcionPremezclaAntibioticoResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element
xmlns:q7="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="AltaPrescripcionPremezclaAntibioticoResult" nillable="true" type="q7:ResultadoInfo"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ModificacionPrescripcionPremezclaAntibiotico">
            <xs:complexType>
                <xs:sequence>
                    <xs:element
xmlns:q8="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="prescripciones" nillable="true" type="q8:ArrayOfPrescripcionPremezclaModificacion"/>
                    <xs:element minOccurs="0" name="Excepcional" type="xs:boolean"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ModificacionPrescripcionPremezclaAntibioticoResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element
xmlns:q9="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ModificacionPrescripcionPremezclaAntibioticoResult" nillable="true"
type="q9:ResultadoInfo"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ObtenerCodigosVerificacionDefinidos">
            <xs:complexType>
                <xs:sequence/>
            </xs:complexType>
        </xs:element>
        <xs:element name="ObtenerCodigosVerificacionDefinidosResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element
xmlns:q10="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerCodigosVerificacionDefinidosResult" nillable="true"
type="q10:ArrayOfCodigoErrorVerificacionInfo"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ObtenerCodigosIncidenciaDefinidos">
            <xs:complexType>
                <xs:sequence/>
            </xs:complexType>
        </xs:element>
    </xs:element>

```

```
<xs:element name="ObtenerCodigosIncidenciaDefinidosResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:q11="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerCodigosIncidenciaDefinidosResult" nillable="true"
type="q11:ArrayOfCodigosIncidenciaInfo"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerCodigosEstadoProcesoTrabajo">
  <xs:complexType>
    <xs:sequence/>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerCodigosEstadoProcesoTrabajoResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:q12="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerCodigosEstadoProcesoTrabajoResult" nillable="true"
type="q12:ArrayOfEstadoTrabajoInfo"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerRelacionEspeciesRegaNomenclator">
  <xs:complexType>
    <xs:sequence/>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerRelacionEspeciesRegaNomenclatorResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:q13="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerRelacionEspeciesRegaNomenclatorResult" nillable="true"
type="q13:ArrayOfEspecieRegaNomenclatorInfo"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerEspeciesREGA">
  <xs:complexType>
    <xs:sequence/>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerEspeciesREGAResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:q14="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerEspeciesREGAResult" nillable="true" type="q14:ArrayOfEspecieInfo"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerEstadoTrabajosIntervalo">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="fcInicial" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="fcFinal" type="xs:dateTime"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ObtenerEstadoTrabajosIntervaloResponse">
  <xs:complexType>
    <xs:sequence>
```

```

                <xs:element
xmlns:q15="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerEstadoTrabajosIntervaloResult" nillable="true" type="q15:ArrayOfEstadoInfo"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ObtenerErroresPorTrabajo">
            <xs:complexType>
                <xs:sequence>
                    <xs:element minOccurs="0" name="idTrabajo" type="xs:int"/>
                </xs:sequence>
            </xs:complexType>
        </xs:element>
        <xs:element name="ObtenerErroresPorTrabajoResponse">
            <xs:complexType>
                <xs:sequence>
                    <xs:element
xmlns:q16="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerErroresPorTrabajoResult" nillable="true" type="q16:ArrayOfErroresTrabajoInfo"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="ObtenerPrescripcionesPorTrabajo">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element minOccurs="0" name="idTrabajo" type="xs:int"/>
                    </xs:sequence>
                </xs:complexType>
            </xs:element>
            <xs:element name="ObtenerPrescripcionesPorTrabajoResponse">
                <xs:complexType>
                    <xs:sequence>
                        <xs:element
xmlns:q17="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerPrescripcionesPorTrabajoResult" nillable="true"
type="q17:ArrayOfPrescripcionResultInfo"/>
                        </xs:sequence>
                    </xs:complexType>
                </xs:element>
                <xs:element name="EliminarPrescripciones">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element
xmlns:q18="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
minOccurs="0" name="prescripciones" nillable="true" type="q18:ArrayOfstring"/>
                            </xs:sequence>
                        </xs:complexType>
                    </xs:element>
                    <xs:element name="EliminarPrescripcionesResponse">
                        <xs:complexType>
                            <xs:sequence>
                                <xs:element
xmlns:q19="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="EliminarPrescripcionesResult" nillable="true" type="q19:ResultadoInfo"/>
                                </xs:sequence>
                            </xs:complexType>
                        </xs:element>
                        <xs:element name="ObtenerCodigosUnidadesDosificacionDefinidos">
                            <xs:complexType>
                                <xs:sequence/>
                            </xs:complexType>
                        </xs:element>
                        <xs:element name="ObtenerCodigosUnidadesDosificacionDefinidosResponse">
                            <xs:complexType>
                                <xs:sequence>

```

```

                <xs:element
xmlns:q20="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
minOccurs="0" name="ObtenerCodigosUnidadesDosificacionDefinidosResult" nillable="true"
type="q20:ArrayOfUnidadDosificacionInfo"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
</xs:schema>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://schemas.microsoft.com/2003/10/Serialization/" attributeFormDefault="qualified"
elementFormDefault="qualified" targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/">
    <xs:element name="anyType" nillable="true" type="xs:anyType"/>
    <xs:element name="anyURI" nillable="true" type="xs:anyURI"/>
    <xs:element name="base64Binary" nillable="true" type="xs:base64Binary"/>
    <xs:element name="boolean" nillable="true" type="xs:boolean"/>
    <xs:element name="byte" nillable="true" type="xs:byte"/>
    <xs:element name="dateTime" nillable="true" type="xs:dateTime"/>
    <xs:element name="decimal" nillable="true" type="xs:decimal"/>
    <xs:element name="double" nillable="true" type="xs:double"/>
    <xs:element name="float" nillable="true" type="xs:float"/>
    <xs:element name="int" nillable="true" type="xs:int"/>
    <xs:element name="long" nillable="true" type="xs:long"/>
    <xs:element name="QName" nillable="true" type="xs:QName"/>
    <xs:element name="short" nillable="true" type="xs:short"/>
    <xs:element name="string" nillable="true" type="xs:string"/>
    <xs:element name="unsignedByte" nillable="true" type="xs:unsignedByte"/>
    <xs:element name="unsignedInt" nillable="true" type="xs:unsignedInt"/>
    <xs:element name="unsignedLong" nillable="true" type="xs:unsignedLong"/>
    <xs:element name="unsignedShort" nillable="true" type="xs:unsignedShort"/>
    <xs:element name="char" nillable="true" type="tns:char"/>
    <xs:simpleType name="char">
        <xs:restriction base="xs:int"/>
    </xs:simpleType>
    <xs:element name="duration" nillable="true" type="tns:duration"/>
    <xs:simpleType name="duration">
        <xs:restriction base="xs:duration">
            <xs:pattern value="-?P(\d*D)?(T(\d*H)?(\d*M)?(\d*(\.\d*)?S)?)?" />
            <xs:minInclusive value="-P10675199DT2H48M5.4775808S" />
            <xs:maxInclusive value="P10675199DT2H48M5.4775807S" />
        </xs:restriction>
    </xs:simpleType>
    <xs:element name="guid" nillable="true" type="tns:guid"/>
    <xs:simpleType name="guid">
        <xs:restriction base="xs:string">
            <xs:pattern value="[\da-fA-F]{8}-[\da-fA-F]{4}-[\da-fA-F]{4}-[\da-fA-F]{4}-[\da-fA-
F]{12}"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:attribute name="FactoryType" type="xs:QName"/>
    <xs:attribute name="Id" type="xs:ID"/>
    <xs:attribute name="Ref" type="xs:IDREF"/>
</xs:schema>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion"
elementFormDefault="qualified"
targetNamespace="http://schemas.datacontract.org/2004/07/PresVet.Data.ServicioComunicacionPrescripcion">
    <xs:import namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
    <xs:complexType name="EstadoInfo">
        <xs:sequence>
            <xs:element minOccurs="0" name="CodEstado" type="xs:int"/>
            <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="IdTrabajo" type="xs:int"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="EstadoInfo" nillable="true" type="tns:EstadoInfo"/>

```



```
<xs:complexType name="ArrayOfPrescripcionAntibioticoAlta">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="PrescripcionAntibioticoAlta"
nillable="true" type="tns:PrescripcionAntibioticoAlta"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfPrescripcionAntibioticoAlta" nillable="true"
type="tns:ArrayOfPrescripcionAntibioticoAlta"/>
  <xs:complexType name="PrescripcionAntibioticoAlta">
    <xs:complexContent mixed="false">
      <xs:extension base="tns:PrescripcionBase">
        <xs:sequence>
          <xs:element minOccurs="0" name="FechaPrescripcion" type="xs:dateTime"/>
          <xs:element minOccurs="0" name="NumeroEnvases" type="xs:decimal"/>
          <xs:element minOccurs="0" name="PorcentajeEnvaseTotal" nillable="true"
type="xs:decimal"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="PrescripcionAntibioticoAlta" nillable="true"
type="tns:PrescripcionAntibioticoAlta"/>
  <xs:complexType name="PrescripcionBase">
    <xs:sequence>
      <xs:element minOccurs="0" name="CodigoNacional" nillable="true" type="xs:int"/>
      <xs:element minOccurs="0" name="EspecieRega" type="xs:short"/>
      <xs:element minOccurs="0" name="Excepcional" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Explotacion" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="NIF" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Nombre" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="NumeroColegiado" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="NumeroReceta" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="PrimerApellido" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="SegundoApellido" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="IdClaseTratamiento" nillable="true" type="xs:int"/>
      <xs:element minOccurs="0" name="IdTipoDispensacion" nillable="true" type="xs:int"/>
      <xs:element minOccurs="0" name="DuracionTratamiento" nillable="true" type="xs:decimal"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="PrescripcionBase" nillable="true" type="tns:PrescripcionBase"/>
  <xs:complexType name="ResultadoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="DescripcionError" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Error" type="xs:boolean"/>
      <xs:element minOccurs="0" name="IdTrabajo" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ResultadoInfo" nillable="true" type="tns:ResultadoInfo"/>
  <xs:complexType name="ArrayOfPrescripcionAntibioticoModificacion">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="PrescripcionAntibioticoModificacion"
nillable="true" type="tns:PrescripcionAntibioticoModificacion"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfPrescripcionAntibioticoModificacion" nillable="true"
type="tns:ArrayOfPrescripcionAntibioticoModificacion"/>
  <xs:complexType name="PrescripcionAntibioticoModificacion">
    <xs:complexContent mixed="false">
      <xs:extension base="tns:PrescripcionBase">
        <xs:sequence>
          <xs:element minOccurs="0" name="CodigoPresvet" nillable="true" type="xs:string"/>
          <xs:element minOccurs="0" name="NumeroEnvases" type="xs:decimal"/>
          <xs:element minOccurs="0" name="PorcentajeEnvaseTotal" nillable="true"
type="xs:decimal"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
```

```
        </xs:extension>
    </xs:complexContent>
</xs:complexType>
<xs:element name="PrescripcionAntibioticoModificacion" nillable="true"
type="tns:PrescripcionAntibioticoModificacion"/>
    <xs:complexType name="ArrayOfPrescripcionPremezclaAlta">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="PrescripcionPremezclaAlta"
nillable="true" type="tns:PrescripcionPremezclaAlta"/>
        </xs:sequence>
    </xs:complexType>
<xs:element name="ArrayOfPrescripcionPremezclaAlta" nillable="true"
type="tns:ArrayOfPrescripcionPremezclaAlta"/>
    <xs:complexType name="PrescripcionPremezclaAlta">
        <xs:complexContent mixed="false">
            <xs:extension base="tns:PrescripcionBase">
                <xs:sequence>
                    <xs:element minOccurs="0" name="FechaPrescripcion" type="xs:dateTime"/>
                    <xs:element minOccurs="0" name="CantidadPienso" type="xs:decimal"/>
                    <xs:element minOccurs="0" name="Dosificacion" type="xs:decimal"/>
                    <xs:element minOccurs="0" name="UnidadDosificacion" type="xs:int"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
<xs:element name="PrescripcionPremezclaAlta" nillable="true"
type="tns:PrescripcionPremezclaAlta"/>
    <xs:complexType name="ArrayOfPrescripcionPremezclaModificacion">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="PrescripcionPremezclaModificacion"
nillable="true" type="tns:PrescripcionPremezclaModificacion"/>
        </xs:sequence>
    </xs:complexType>
<xs:element name="ArrayOfPrescripcionPremezclaModificacion" nillable="true"
type="tns:ArrayOfPrescripcionPremezclaModificacion"/>
    <xs:complexType name="PrescripcionPremezclaModificacion">
        <xs:complexContent mixed="false">
            <xs:extension base="tns:PrescripcionBase">
                <xs:sequence>
                    <xs:element minOccurs="0" name="CodigoPresvet" nillable="true" type="xs:string"/>
                    <xs:element minOccurs="0" name="CantidadPienso" type="xs:decimal"/>
                    <xs:element minOccurs="0" name="Dosificacion" type="xs:decimal"/>
                    <xs:element minOccurs="0" name="UnidadDosificacion" type="xs:int"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>
<xs:element name="PrescripcionPremezclaModificacion" nillable="true"
type="tns:PrescripcionPremezclaModificacion"/>
    <xs:complexType name="ArrayOfCodigoErrorVerificacionInfo">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="CodigoErrorVerificacionInfo"
nillable="true" type="tns:CodigoErrorVerificacionInfo"/>
        </xs:sequence>
    </xs:complexType>
<xs:element name="ArrayOfCodigoErrorVerificacionInfo" nillable="true"
type="tns:ArrayOfCodigoErrorVerificacionInfo"/>
    <xs:complexType name="CodigoErrorVerificacionInfo">
        <xs:sequence>
            <xs:element minOccurs="0" name="CodigoErrorVerificacion" type="xs:int"/>
            <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
<xs:element name="CodigoErrorVerificacionInfo" nillable="true"
type="tns:CodigoErrorVerificacionInfo"/>
```

```
<xs:complexType name="ArrayOfCodigosIncidenciaInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="CodigosIncidenciaInfo"
nillable="true" type="tns:CodigosIncidenciaInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfCodigosIncidenciaInfo" nillable="true"
type="tns:ArrayOfCodigosIncidenciaInfo"/>
  <xs:complexType name="CodigosIncidenciaInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="CodigoIncidencia" type="xs:int"/>
      <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="CodigosIncidenciaInfo" nillable="true" type="tns:CodigosIncidenciaInfo"/>
  <xs:complexType name="ArrayOfEstadoTrabajoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="EstadoTrabajoInfo" nillable="true"
type="tns:EstadoTrabajoInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfEstadoTrabajoInfo" nillable="true" type="tns:ArrayOfEstadoTrabajoInfo"/>
  <xs:complexType name="EstadoTrabajoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="IdEstado" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="EstadoTrabajoInfo" nillable="true" type="tns:EstadoTrabajoInfo"/>
  <xs:complexType name="ArrayOfEspecieRegaNomenclatorInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="EspecieRegaNomenclatorInfo"
nillable="true" type="tns:EspecieRegaNomenclatorInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfEspecieRegaNomenclatorInfo" nillable="true"
type="tns:ArrayOfEspecieRegaNomenclatorInfo"/>
  <xs:complexType name="EspecieRegaNomenclatorInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="EspeciesNomenclator" nillable="true"
type="tns:ArrayOfEspecieInfo"/>
      <xs:element minOccurs="0" name="IdEspecieRega" type="xs:int"/>
      <xs:element minOccurs="0" name="NombreEspecieRega" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="EspecieRegaNomenclatorInfo" nillable="true"
type="tns:EspecieRegaNomenclatorInfo"/>
  <xs:complexType name="ArrayOfEspecieInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="EspecieInfo" nillable="true"
type="tns:EspecieInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfEspecieInfo" nillable="true" type="tns:ArrayOfEspecieInfo"/>
  <xs:complexType name="EspecieInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="IdEspecie" type="xs:int"/>
      <xs:element minOccurs="0" name="Nombre" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="EspecieInfo" nillable="true" type="tns:EspecieInfo"/>
  <xs:complexType name="ArrayOfEstadoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="EstadoInfo" nillable="true"
type="tns:EstadoInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfEstadoInfo" nillable="true" type="tns:ArrayOfEstadoInfo"/>
```

```

    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfEstadoInfo" nillable="true" type="tns:ArrayOfEstadoInfo"/>
  <xs:complexType name="ArrayOfErroresTrabajoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="ErroresTrabajoInfo" nillable="true"
type="tns:ErroresTrabajoInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfErroresTrabajoInfo" nillable="true"
type="tns:ArrayOfErroresTrabajoInfo"/>
  <xs:complexType name="ErroresTrabajoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="CodigoPresvet" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Errores" nillable="true" type="q1:ArrayOfstring"/>
      <xs:element minOccurs="0" name="Fila" nillable="true" type="xs:int"/>
      <xs:element minOccurs="0" name="IdTrabajo" type="xs:int"/>
      <xs:element minOccurs="0" name="NumeroReceta" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ErroresTrabajoInfo" nillable="true" type="tns:ErroresTrabajoInfo"/>
  <xs:complexType name="ArrayOfPrescripcionResultInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="PrescripcionResultInfo"
nillable="true" type="tns:PrescripcionResultInfo"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfPrescripcionResultInfo" nillable="true"
type="tns:ArrayOfPrescripcionResultInfo"/>
  <xs:complexType name="PrescripcionResultInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="CantidadPienso" nillable="true" type="xs:decimal"/>
      <xs:element minOccurs="0" name="ClaseTratamiento" nillable="true"
type="tns:ClaseTratamientoInfo"/>
      <xs:element minOccurs="0" name="CodigoNacional" nillable="true" type="xs:int"/>
      <xs:element minOccurs="0" name="CodigoPresvet" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Dosificacion" nillable="true" type="xs:decimal"/>
      <xs:element minOccurs="0" name="DuracionTratamiento" nillable="true" type="xs:decimal"/>
      <xs:element minOccurs="0" name="Excepcional" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Explotacion" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="FechaPrescripcion" type="xs:dateTime"/>
      <xs:element minOccurs="0" name="Incidencias" nillable="true" type="q2:ArrayOfstring"/>
      <xs:element minOccurs="0" name="NIF" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Nombre" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="NumeroColegiado" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="NumeroEnvases" nillable="true" type="xs:decimal"/>
      <xs:element minOccurs="0" name="NumeroReceta" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="PorcentajeEnvaseTotal" nillable="true" type="xs:decimal"/>
      <xs:element minOccurs="0" name="PrimerApellido" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="SegundoApellido" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="TipoDispensacion" nillable="true"
type="tns:TipoDispensacionInfo"/>
      <xs:element minOccurs="0" name="UnidadDosificacion" nillable="true" type="xs:int"/>
      <xs:element minOccurs="0" name="especieNomenclator" nillable="true" type="xs:short"/>
      <xs:element minOccurs="0" name="especieRega" nillable="true" type="xs:short"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="PrescripcionResultInfo" nillable="true" type="tns:PrescripcionResultInfo"/>
  <xs:complexType name="ClaseTratamientoInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Id" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>

```

```
</xs:complexType>
<xs:element name="ClaseTratamientoInfo" nillable="true" type="tns:ClaseTratamientoInfo"/>
<xs:complexType name="TipoDispensacionInfo">
  <xs:sequence>
    <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Id" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="TipoDispensacionInfo" nillable="true" type="tns:TipoDispensacionInfo"/>
<xs:complexType name="ArrayOfUnidadDosificacionInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="UnidadDosificacionInfo"
nillable="true" type="tns:UnidadDosificacionInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfUnidadDosificacionInfo" nillable="true"
type="tns:ArrayOfUnidadDosificacionInfo"/>
  <xs:complexType name="UnidadDosificacionInfo">
    <xs:sequence>
      <xs:element minOccurs="0" name="Descripcion" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="IdUnidadDosificacion" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="UnidadDosificacionInfo" nillable="true" type="tns:UnidadDosificacionInfo"/>
</xs:schema>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://schemas.microsoft.com/2003/10/Serialization/Arrays" elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays">
  <xs:complexType name="ArrayOfstring">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfstring" nillable="true" type="tns:ArrayOfstring"/>
</xs:schema>
</wsdl:types>
<wsdl:message name="IPresVetComunicacionPrescripcion_EstaActivo_InputMessage">
  <wsdl:part name="parameters" element="tns:EstaActivo"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_EstaActivo_OutputMessage">
  <wsdl:part name="parameters" element="tns:EstaActivoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajo_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerEstadoTrabajo"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajo_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerEstadoTrabajoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_AltaPrescripcionAntibiotico_InputMessage">
  <wsdl:part name="parameters" element="tns:AltaPrescripcionAntibiotico"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_AltaPrescripcionAntibiotico_OutputMessage">
  <wsdl:part name="parameters" element="tns:AltaPrescripcionAntibioticoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ModificacionPrescripcionAntibiotico_InputMessage">
  <wsdl:part name="parameters" element="tns:ModificacionPrescripcionAntibiotico"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ModificacionPrescripcionAntibiotico_OutputMessage">
  <wsdl:part name="parameters" element="tns:ModificacionPrescripcionAntibioticoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_AltaPrescripcionPremezclaAntibiotico_InputMessage">
  <wsdl:part name="parameters" element="tns:AltaPrescripcionPremezclaAntibiotico"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_AltaPrescripcionPremezclaAntibiotico_OutputMessage">
```

```
<wsdl:part name="parameters" element="tns:AltaPrescripcionPremezclaAntibioticoResponse"/>
</wsdl:message>
<wsdl:message
name="IPresVetComunicacionPrescripcion_ModificacionPrescripcionPremezclaAntibiotico_InputMessage">
  <wsdl:part name="parameters" element="tns:ModificacionPrescripcionPremezclaAntibiotico"/>
</wsdl:message>
<wsdl:message
name="IPresVetComunicacionPrescripcion_ModificacionPrescripcionPremezclaAntibiotico_OutputMessage">
  <wsdl:part name="parameters" element="tns:ModificacionPrescripcionPremezclaAntibioticoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerCodigosVerificacionDefinidos_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosVerificacionDefinidos"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerCodigosVerificacionDefinidos_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosVerificacionDefinidosResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerCodigosIncidenciaDefinidos_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosIncidenciaDefinidos"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerCodigosIncidenciaDefinidos_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosIncidenciaDefinidosResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerCodigosEstadoProcesoTrabajo_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosEstadoProcesoTrabajo"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerCodigosEstadoProcesoTrabajo_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosEstadoProcesoTrabajoResponse"/>
</wsdl:message>
<wsdl:message
name="IPresVetComunicacionPrescripcion_ObtenerRelacionEspeciesRegaNomenclator_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerRelacionEspeciesRegaNomenclator"/>
</wsdl:message>
<wsdl:message
name="IPresVetComunicacionPrescripcion_ObtenerRelacionEspeciesRegaNomenclator_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerRelacionEspeciesRegaNomenclatorResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerEspeciesREGA_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerEspeciesREGA"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerEspeciesREGA_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerEspeciesREGAResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajosIntervalo_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerEstadoTrabajosIntervalo"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajosIntervalo_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerEstadoTrabajosIntervaloResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerErroresPorTrabajo_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerErroresPorTrabajo"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerErroresPorTrabajo_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerErroresPorTrabajoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerPrescripcionesPorTrabajo_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerPrescripcionesPorTrabajo"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_ObtenerPrescripcionesPorTrabajo_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerPrescripcionesPorTrabajoResponse"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_EliminarPrescripciones_InputMessage">
  <wsdl:part name="parameters" element="tns:EliminarPrescripciones"/>
</wsdl:message>
<wsdl:message name="IPresVetComunicacionPrescripcion_EliminarPrescripciones_OutputMessage">
  <wsdl:part name="parameters" element="tns:EliminarPrescripcionesResponse"/>
</wsdl:message>
```



```
</wsdl:message>
<wsdl:message
name="IPresVetComunicacionPrescripcion_ObtenerCodigosUnidadesDosificacionDefinidos_InputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosUnidadesDosificacionDefinidos"/>
</wsdl:message>
<wsdl:message
name="IPresVetComunicacionPrescripcion_ObtenerCodigosUnidadesDosificacionDefinidos_OutputMessage">
  <wsdl:part name="parameters" element="tns:ObtenerCodigosUnidadesDosificacionDefinidosResponse"/>
</wsdl:message>
<wsdl:portType name="IPresVetComunicacionPrescripcion">
  <wsdl:operation name="EstaActivo">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/EstaActivo"
message="tns:IPresVetComunicacionPrescripcion_EstaActivo_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/EstaActivoResponse"
message="tns:IPresVetComunicacionPrescripcion_EstaActivo_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ObtenerEstadoTrabajo">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEstadoTrabajo"
message="tns:IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajo_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEstadoTrabajoResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajo_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="AltaPrescripcionAntibiotico">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/AltaPrescripcionAntibiotico"
message="tns:IPresVetComunicacionPrescripcion_AltaPrescripcionAntibiotico_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/AltaPrescripcionAntibioticoResponse"
message="tns:IPresVetComunicacionPrescripcion_AltaPrescripcionAntibiotico_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ModificacionPrescripcionAntibiotico">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ModificacionPrescripcionAntibiotico"
message="tns:IPresVetComunicacionPrescripcion_ModificacionPrescripcionAntibiotico_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ModificacionPrescripcionAntibioticoResponse"
message="tns:IPresVetComunicacionPrescripcion_ModificacionPrescripcionAntibiotico_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="AltaPrescripcionPremezclaAntibiotico">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/AltaPrescripcionPremezclaAntibiotico"
message="tns:IPresVetComunicacionPrescripcion_AltaPrescripcionPremezclaAntibiotico_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/AltaPrescripcionPremezclaAntibioticoResponse"
message="tns:IPresVetComunicacionPrescripcion_AltaPrescripcionPremezclaAntibiotico_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ModificacionPrescripcionPremezclaAntibiotico">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ModificacionPrescripcionPremezclaAntibiotico"
message="tns:IPresVetComunicacionPrescripcion_ModificacionPrescripcionPremezclaAntibiotico_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ModificacionPrescripcionPremezclaAntibiotico
Response"
message="tns:IPresVetComunicacionPrescripcion_ModificacionPrescripcionPremezclaAntibiotico_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ObtenerCodigosVerificacionDefinidos">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosVerificacionDefinidos"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosVerificacionDefinidos_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosVerificacionDefinidosResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosVerificacionDefinidos_OutputMessage"/>
  </wsdl:operation>
```

```
<wsdl:operation name="ObtenerCodigosIncidenciaDefinidos">
  <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosIncidenciaDefinidos"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosIncidenciaDefinidos_InputMessage"/>
  <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosIncidenciaDefinidosResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosIncidenciaDefinidos_OutputMessage"/>
  </wsdl:operation>
  <wsdl:operation name="ObtenerCodigosEstadoProcesoTrabajo">
    <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosEstadoProcesoTrabajo"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosEstadoProcesoTrabajo_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosEstadoProcesoTrabajoResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosEstadoProcesoTrabajo_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ObtenerRelacionEspeciesRegaNomenclator">
      <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerRelacionEspeciesRegaNomenclator"
message="tns:IPresVetComunicacionPrescripcion_ObtenerRelacionEspeciesRegaNomenclator_InputMessage"/>
      <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerRelacionEspeciesRegaNomenclatorRespon
se" message="tns:IPresVetComunicacionPrescripcion_ObtenerRelacionEspeciesRegaNomenclator_OutputMessage"/>
      </wsdl:operation>
      <wsdl:operation name="ObtenerEspeciesREGA">
        <wsdl:input wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEspeciesREGA"
message="tns:IPresVetComunicacionPrescripcion_ObtenerEspeciesREGA_InputMessage"/>
        <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEspeciesREGAResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerEspeciesREGA_OutputMessage"/>
        </wsdl:operation>
        <wsdl:operation name="ObtenerEstadoTrabajosIntervalo">
          <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEstadoTrabajosIntervalo"
message="tns:IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajosIntervalo_InputMessage"/>
          <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEstadoTrabajosIntervaloResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerEstadoTrabajosIntervalo_OutputMessage"/>
          </wsdl:operation>
          <wsdl:operation name="ObtenerErroresPorTrabajo">
            <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerErroresPorTrabajo"
message="tns:IPresVetComunicacionPrescripcion_ObtenerErroresPorTrabajo_InputMessage"/>
            <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerErroresPorTrabajoResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerErroresPorTrabajo_OutputMessage"/>
            </wsdl:operation>
            <wsdl:operation name="ObtenerPrescripcionesPorTrabajo">
              <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerPrescripcionesPorTrabajo"
message="tns:IPresVetComunicacionPrescripcion_ObtenerPrescripcionesPorTrabajo_InputMessage"/>
              <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerPrescripcionesPorTrabajoResponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerPrescripcionesPorTrabajo_OutputMessage"/>
              </wsdl:operation>
              <wsdl:operation name="EliminarPrescripciones">
                <wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/EliminarPrescripciones"
message="tns:IPresVetComunicacionPrescripcion_EliminarPrescripciones_InputMessage"/>
                <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/EliminarPrescripcionesResponse"
message="tns:IPresVetComunicacionPrescripcion_EliminarPrescripciones_OutputMessage"/>
                </wsdl:operation>
                <wsdl:operation name="ObtenerCodigosUnidadesDosificacionDefinidos">
```



```
<wsdl:input
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosUnidadesDosificacionDefinidos"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosUnidadesDosificacionDefinidos_InputMessage"/>
  <wsdl:output
wsaw:Action="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosUnidadesDosificacionDefinidosR
esponse"
message="tns:IPresVetComunicacionPrescripcion_ObtenerCodigosUnidadesDosificacionDefinidos_OutputMessage"/>
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="BasicEndpoint" type="tns:IPresVetComunicacionPrescripcion">
  <wsp:PolicyReference URI="#BasicEndpoint_policy"/>
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="EstaActivo">
    <soap12:operation      soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/EstaActivo"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ObtenerEstadoTrabajo">
    <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEstadoTrabajo" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="AltaPrescripcionAntibiotico">
    <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/AltaPrescripcionAntibiotico"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="ModificacionPrescripcionAntibiotico">
    <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ModificacionPrescripcionAntibiotico"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="AltaPrescripcionPremezclaAntibiotico">
    <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/AltaPrescripcionPremezclaAntibiotico"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
```

```
<wsdl:operation name="ModificacionPrescripcionPremezclaAntibiotico">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ModificacionPrescripcionPremezclaAntibiotico"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerCodigosVerificacionDefinidos">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosVerificacionDefinidos"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerCodigosIncidenciaDefinidos">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosIncidenciaDefinidos"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerCodigosEstadoProcesoTrabajo">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosEstadoProcesoTrabajo"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerRelacionEspeciesRegaNomenclator">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerRelacionEspeciesRegaNomenclator"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerEspeciesREGA">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEspeciesREGA" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
```

```
<wsdl:operation name="ObtenerEstadoTrabajosIntervalo">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerEstadoTrabajosIntervalo"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerErroresPorTrabajo">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerErroresPorTrabajo" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerPrescripcionesPorTrabajo">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerPrescripcionesPorTrabajo"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="EliminarPrescripciones">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/EliminarPrescripciones" style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ObtenerCodigosUnidadesDosificacionDefinidos">
  <soap12:operation
soapAction="http://tempuri.org/IPresVetComunicacionPrescripcion/ObtenerCodigosUnidadesDosificacionDefinidos"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="PresVetComunicacionPrescripcion">
  <wsdl:port name="BasicEndpoint" binding="tns:BasicEndpoint">
    <soap12:address
servicio.mapa.gob.es/PresVetComunicacionPrescripcion.svc"/>
    <wsa10:EndpointReference>
      <wsa10:Address>https://integracion.servicio.mapa.gob.es/PresVetComunicacionPrescripcion.svc</wsa10:Address>
    </wsa10:EndpointReference>
  </wsdl:port>
</wsdl:service>
</wsdl:definitions>
```

7. Contrato del servicio.

Operaciones que permite el servicio WCF.

```
[OperationContract]  
string EstaActivo();
```

```
[OperationContract]  
EstadoInfo ObtenerEstadoTrabajo(int idTrabajo); //Acceder al estado del proceso de los envios masivos por id.
```

```
[OperationContract]  
ResultadoInfo AltaPrescripcionAntibiotico(List<PrescripcionAntibioticoAlta> prescripciones, bool  
Excepcional); //Para enviar altas de prescripcion antibióticos.
```

```
[OperationContract]  
ResultadoInfo ModificacionPrescripcionAntibiotico(List<PrescripcionAntibioticoModificacion> prescripciones,  
bool Excepcional); //Para enviar modificaciones de antibióticos.
```

```
[OperationContract]  
ResultadoInfo AltaPrescripcionPremezclaAntibiotico(List<PrescripcionPremezclaAlta> prescripciones, bool  
Excepcional); //Para enviar altas de premezcla de antibióticos
```

```
[OperationContract]  
ResultadoInfo ModificacionPrescripcionPremezclaAntibiotico(List<PrescripcionPremezclaModificacion>  
prescripciones, bool Excepcional); //Para enviar modificaciones de premezcla de antibióticos.
```

```
[OperationContract]  
List<CodigoErrorVerificacionInfo> ObtenerCodigosVerificacionDefinidos(); // Para obtener Los códigos de error  
definidos en el sistema en la verificación
```

```
[OperationContract]  
List<CodigosIncidenciaInfo> ObtenerCodigosIncidenciaDefinidos(); // Para obtener los códigos de incidencia  
definidos en el sistema.
```

```
[OperationContract]  
List<EstadoTrabajoInfo> ObtenerCodigosEstadoProcesoTrabajo();
```

```
[OperationContract]  
List<EspecieInfo> ObtenerEspeciesREGA();
```

```
[OperationContract]  
List<EspecieRegaNomenclatorInfo> ObtenerRelacionEspeciesRegaNomenclator();
```

```
[OperationContract]  
List<EstadoInfo> ObtenerEstadoTrabajosIntervalo(DateTime fcInicial, DateTime fcFinal);
```

```
[OperationContract]  
List<ErroresTrabajoInfo> ObtenerErroresPorTrabajo(int idTrabajo);
```

```
[OperationContract]  
List<PrescripcionResultInfo> ObtenerPrescripcionesPorTrabajo(int idTrabajo);
```

```
[OperationContract]  
ResultadoInfo EliminarPrescripciones(List<string> prescripciones);
```

```
[OperationContract]  
List<UnidadDosificacionInfo> ObtenerCodigosUnidadesDosificacionDefinidos();
```

8. Definición del contrato de datos del servicio.

Definición de los contratos de datos que utiliza el servicio WCF.

```
[DataContract]
public class PrescripcionBase
{
    string explotacion;
    string nif;
    string numeroColegiado;
    string nombre;
    string primerApellido;
    string segundoApellido;
    int codigoNacional;
    short especieRega;
    string numeroReceta;
    string excepcional;
    int? idClaseTratamiento;
    int? idTipoDispensacion;
    decimal? duracionTratamiento;

    [DataMember]
    public string Explotacion { get { return explotacion; } set { explotacion = value; } }
    [DataMember]
    public string NIF { get { return nif; } set { nif = value; } }
    [DataMember]
    public string NumeroColegiado { get { return numeroColegiado; } set { numeroColegiado = value; } }
    [DataMember]
    public string Nombre { get { return nombre; } set { nombre = value; } }
    [DataMember]
    public string PrimerApellido { get { return primerApellido; } set { primerApellido = value; } }
    [DataMember]
    public string SegundoApellido { get { return segundoApellido; } set { segundoApellido = value; } }
    [DataMember]
    public int CodigoNacional { get { return codigoNacional; } set { codigoNacional = value; } }
    [DataMember]
    public short EspecieRega { get { return especieRega; } set { especieRega = value; } }
    [DataMember]
    public string NumeroReceta { get { return numeroReceta; } set { numeroReceta = value; } }
    [DataMember]
    public string Excepcional { get { return excepcional; } set { excepcional = value; } }
    [DataMember]
    public int? IdClaseTratamiento { get { return idClaseTratamiento; } set { idClaseTratamiento =
value; } }
    [DataMember]
    public int? IdTipoDispensacion { get { return idTipoDispensacion; } set { idTipoDispensacion =
value; } }
    [DataMember]
    public decimal? DuracionTratamiento { get { return duracionTratamiento; } set {
duracionTratamiento = value; } }
}

public class PrescripcionInsert : PrescripcionBase
{
    string codigoPresVet;

    public string CodigoPresvet { get { return codigoPresVet; } set { codigoPresVet = value; } }
    DateTime fechaPrescripcion;
    decimal? numeroEnvases;

    decimal? cantidadPienso;
    decimal? dosificacion;
    int? unidaddosificacion;
}
```

```
        public DateTime FechaPrescripcion { get { return fechaPrescripcion; } set { fechaPrescripcion =
value; } }
        public decimal? NumeroEnvases { get { if (numeroEnvases.HasValue) { return numeroEnvases.Value; }
else { return null; } } set { numeroEnvases = value; } }
        public decimal? CantidadPienso { get { if (cantidadPienso.HasValue) { return cantidadPienso.Value; }
else { return null; } } set { cantidadPienso = value; } }

        public decimal? Dosificacion { get { if (dosificacion.HasValue) { return dosificacion.Value; } else {
return null; } } set { dosificacion = value; } }

        public int? UnidadDosificacion { get { if (unidaddosificacion.HasValue) { return
unidaddosificacion.Value; } else { return null; } } set { unidaddosificacion = value; } }
        public int? UnidadDosificacion { get { if (unidaddosificacion.HasValue) { return
unidaddosificacion.Value; } else { return null; } } set { unidaddosificacion = value; } }

        decimal? porcentajeEnvaseTotal;

        public decimal? PorcentajeEnvaseTotal { get { return porcentajeEnvaseTotal; } set {
porcentajeEnvaseTotal = value; } }
    }
    [DataContract]
    public class EstadoInfo
    {
        int idTrabajo;
        int codEstado;
        string descripcion;

        [DataMember]
        public int IdTrabajo { get { return idTrabajo; } set { idTrabajo = value; } }
        [DataMember]
        public int CodEstado { get { return codEstado; } set { codEstado = value; } }
        [DataMember]
        public string Descripcion { get { return descripcion; } set { descripcion = value; } }
    }

    [DataContract]
    public class EstadoTrabajoInfo
    {
        int idEstado;
        string descripcion;

        [DataMember]
        public int IdEstado { get { return idEstado; } set { idEstado = value; } }
        [DataMember]
        public string Descripcion { get { return descripcion; } set { descripcion = value; } }
    }

    [DataContract]
    public class TrabajoInfo
    {
        int id;
        DateTime fechaCreacion;
        string resultadoVerificacion;
        string resultadoProceso;
        string operacion;
        string estadoProceso;

        [DataMember]
        public int Id { get { return id; } set { id = value; } }
        [DataMember]
        public string ResultadoVerificacion { get { return resultadoVerificacion; } set {
resultadoVerificacion = value; } }
        [DataMember]
```

```
public string ResultadoProceso { get { return resultadoProceso; } set { resultadoProceso = value; } }
[DataMember]
public string Operacion { get { return operacion; } set { operacion = value; } }
[DataMember]
public string EstadoProceso { get { return estadoProceso; } set { estadoProceso = value; } }
[DataMember]
public DateTime FechaCreacion { get { return fechaCreacion; } set { fechaCreacion = value; } }
[DataMember]
public DateTime? FechaGrabacion { get; set; }
[DataMember]
public DateTime? FechaVerificacion { get; set; }
}

[DataContract]
public classCodigoErrorVerificacionInfo
{
    int codigoError;
    string descripcion;

    [DataMember]
    public intCodigoErrorVerificacion { get { return codigoError; } set { codigoError = value; } }
    [DataMember]
    public stringDescripcion { get { return descripcion; } set { descripcion = value; } }
}

[DataContract]
public classCodigosIncidenciaInfo
{
    int codigoIncidencia;
    string descripcion;

    [DataMember]
    public intCodigoIncidencia { get { return codigoIncidencia; } set { codigoIncidencia = value; } }
    [DataMember]
    public stringDescripcion { get { return descripcion; } set { descripcion = value; } }
}

[DataContract]
public classPrescripcionAntibioticoAlta : PrescripcionBase
{
    DateTime fechaPrescripcion;
    [DataMember]
    public DateTimeFechaPrescripcion { get { return fechaPrescripcion; } set { fechaPrescripcion =
value; } }
    decimal numeroEnvases;
    [DataMember]
    public decimalNumeroEnvases { get { return numeroEnvases; } set { numeroEnvases = value; } }
    [DataMember]
    public decimal? PorcentajeEnvaseTotal { get { return porcentajeEnvaseTotal; } set {
porcentajeEnvaseTotal = value; } }
}

[DataContract]
public classPrescripcionAntibioticoModificacion : PrescripcionBase
{
    string codigoPresVet;
    decimal numeroEnvases;
    decimal? porcentajeEnvaseTotal;

    [DataMember]
    [StringLength(19, MinimumLength = 19)]
```

```
public string CodigoPresvet { get { return codigoPresVet; } set { codigoPresVet = value; } }

[DataMember]
public decimal NumeroEnvases { get { return numeroEnvases; } set { numeroEnvases = value; } }
[DataMember]
public decimal? PorcentajeEnvaseTotal { get { return porcentajeEnvaseTotal; } set {
porcentajeEnvaseTotal = value; } }

}

[DataContract]
public class PrescripcionPremezclaAlta : PrescripcionBase
{
    DateTime fechaPrescripcion;
    [DataMember]
    public DateTime FechaPrescripcion { get { return fechaPrescripcion; } set { fechaPrescripcion =
value; } }
    decimal cantidadPienso;

    [DataMember]
    public decimal CantidadPienso { get { return cantidadPienso; } set { cantidadPienso = value; } }
    decimal dosificacion;

    [DataMember]
    public decimal Dosificacion { get { return dosificacion; } set { dosificacion = value; } }
    int unidadDosificacion;

    [DataMember]
    public int UnidadDosificacion{ get { return unidadDosificacion; } set { unidadDosificacion = value; } }
}

[DataContract]
public class PrescripcionPremezclaModificacion : PrescripcionBase
{
    string codigoPresVet;

    [DataMember]
    [StringLength(19, MinimumLength = 19)]
    public string CodigoPresvet { get { return codigoPresVet; } set { codigoPresVet = value; } }

    decimal cantidadPienso;

    [DataMember]
    [Required]
    public decimal CantidadPienso { get { return cantidadPienso; } set { cantidadPienso = value; } }

    decimal dosificacion;

    [DataMember]
    [Required]
    public decimal Dosificacion { get { return dosificacion; } set { dosificacion = value; } }
    int unidadDosificacion;

    [DataMember]
    [Required]
    public int UnidadDosificacion {get { return unidadDosificacion; } set { unidadDosificacion = value;}}
}

[DataContract]
public class ResultadoInfo
{
```



```
[DataMember]
public int IdTrabajo { get; set; }
[DataMember]
public bool Error { get; set; }
[DataMember]
public string DescripcionError { get; set; }
}

[DataContract]
public class EspecieInfo
{
    [DataMember]
    public int IdEspecie { get; set; }
    [DataMember]
    public string Nombre { get; set; }
}

[DataContract]
public class EspecieRegaNomencIatorInfo
{
    [DataMember]
    public int IdEspecieRega { get; set; }
    [DataMember]
    public string NombreEspecieRega { get; set; }
    [DataMember]
    public List<EspecieInfo> EspeciesNomencIator { get; set; }
}

[DataContract]
public class SustanciaActivaInfo
{
    [DataMember]
    public int IdSustanciaActiva { get; set; }
    [DataMember]
    public string Nombre { get; set; }
    [DataMember]
    public string Codificacion { get; set; }
}

[DataContract]
public class FormaFarmaceuticaInfo
{
    [DataMember]
    public int IdFormaFarmaceutica { get; set; }
    [DataMember]
    public string Nombre { get; set; }
}

[DataContract]
public class PrescripcionResultInfo
{
    string explotacion;
    [DataMember]
    public string Explotacion { get { return explotacion; } set { explotacion = value; } }

    string nif;
    [DataMember]
    public string NIF { get { return nif; } set { nif = value; } }
    string numeroColegiado;
    [DataMember]
    public string NumeroColegiado { get { return numeroColegiado; } set { numeroColegiado = value; } }
}
```

```
string nombre;
[DataMember]
public string Nombre { get { return nombre; } set { nombre = value; } }
string primerApellido;
[DataMember]
public string PrimerApellido { get { return primerApellido; } set { primerApellido = value; } }
string segundoApellido;
[DataMember]
public string SegundoApellido { get { return segundoApellido; } set { segundoApellido = value; } }
int? codigoNacional;
[DataMember]
public int? CodigoNacional { get { return codigoNacional; } set { codigoNacional = value; } }
[DataMember]
short? especieRega;
public short? EspecieRega { get { return especieRega; } set { especieRega = value; } }
[DataMember]
short? especieNomenclator;
public short? EspecieNomenclator { get { return especieNomenclator; } set { especieNomenclator=
value; } }
string numeroReceta;
[DataMember]
public string NumeroReceta { get { return numeroReceta; } set { numeroReceta = value; } }
string excepcional;
[DataMember]
public string Excepcional { get { return excepcional; } set { excepcional = value; } }

string codigoPresVet;

[DataMember]
[StringLength (19, MinimumLength = 19)]
public string CodigoPresvet { get { return codigoPresVet; } set { codigoPresVet = value; } }

DateTime fechaPrescripcion;
[DataMember]
public DateTime FechaPrescripcion { get { return fechaPrescripcion; } set { fechaPrescripcion =
value; } }

List<string> incidencias;
[DataMember]
public List<string> Incidencias { get { return incidencias; } set { incidencias = value; } }

decimal? cantidadPienso;
[DataMember]
public decimal? CantidadPienso { get { return cantidadPienso; } set { cantidadPienso = value; } }
decimal? numeroEnvases;
[DataMember]
public decimal? NumeroEnvases { get { return numeroEnvases; } set { numeroEnvases = value; } }
decimal? dosificacion;
[DataMember]
public decimal? Dosificacion { get { return dosificacion; } set { dosificacion = value; } }
int? unidadDosificacion;
[DataMember]
public int? UnidadDosificacion { get { return unidadDosificacion; } set { unidadDosificacion = value;
} }

[DataMember]
ClaseTratamientoInfo claseTratamiento;
[DataMember]
public ClaseTratamientoInfo ClaseTratamiento { get { return claseTratamiento; } set {
claseTratamiento = value; } }
TipoDispensacionInfo tipoDispensacion;
[DataMember]
public TipoDispensacionInfo TipoDispensacion { get { return tipoDispensacion; } set {
tipoDispensacion = value; } }
decimal? duracionTratamiento;
[DataMember]
```

```
        public decimal? DuracionTratamiento { get { return duracionTratamiento; } set {
duracionTratamiento = value; } }
        decimal? porcentajeEnvaseTotal;
        [DataMember]
        public decimal? PorcentajeEnvaseTotal { get { return porcentajeEnvaseTotal; } set {
porcentajeEnvaseTotal = value; } }
    }
    [DataContract]
    public class ErroresTrabajoInfo
    {
        int idTrabajo;
        [DataMember]
        public int IdTrabajo { get { return idTrabajo; } set { idTrabajo = value; } }

        string numeroReceta;
        [DataMember]
        public string NumeroReceta { get { return numeroReceta; } set { numeroReceta = value; } }

        string codigoPresvet;
        [DataMember]
        public string CodigoPresvet { get { return codigoPresvet; } set { codigoPresvet = value; } }

        List<string> errores;
        [DataMember]
        public List<string> Errores { get { return errores; } set { errores = value; } }

        int? fila;
        [DataMember]
        public int? Fila { get { return fila; } set { fila= value; } }
    }

    [DataContract]
    public class PrescripcionesEliminadaInfo
    {
        string codigoPresVet;

        [DataMember]
        [StringLength(19, MinimumLength = 19)]
        public string CodigoPresvet { get { return codigoPresVet; } set { codigoPresVet = value; } }

        bool eliminada;
        [DataMember]
        public bool Eliminada { get { return eliminada; } set { eliminada = value; } }

        string motivo;
        [DataMember]
        public string Motivo { get { return motivo; } set { motivo = value; } }
    }

    [DataContract]
    public class IntervaloInfo
    {
        DateTime fcini;
        DateTime fcfi;

        [DataMember]
        public DateTime fcInicial { get { return fcini; } set { fcini = value; } }
        [DataMember]
        public DateTime fcFinal { get { return fcfi; } set { fcfi = value; } }
    }

    [DataContract]
    public class UnidadDosificacionInfo
```

```
{  
    int idUnidadDosificacion;  
    string descripcion;  
  
    [DataMember]  
    public int IdUnidadDosificacion { get { return idUnidadDosificacion; } set { idUnidadDosificacion =  
value; } }  
    [DataMember]  
    public string Descripcion { get { return descripcion; } set { descripcion = value; } }  
} }  
[DataContract]  
public class ClaseTratamientoInfo  
{  
    [DataMember]  
    public int Id { get; set; }  
    [DataMember]  
    public string Descripcion { get; set; }  
  
}  
[DataContract]  
public class TipoDispensacionInfo  
{  
    [DataMember]  
    public int Id { get; set; }  
    [DataMember]  
    public string Descripcion { get; set; }  
  
}
```

9. Aclaraciones sobre los estados

Si el trabajo se encuentra en estado

- 1- **Correcto:** En este estado las prescripciones se han grabado sin ningún problema
- 2- **Correcto con incidencias:** En este estado las prescripciones se han grabado pero alguna o todas tienen asignadas una o varias incidencias del tipo que se indican en el apartado 3.2 de este documento.
- 3- **En proceso:** En este estado el trabajo aún se está procesando y hasta que no termine no se cambiará el estado
- 4- **Rechazado:** Este estado indica que no se ha podido realizar la inserción de ninguna de las prescripciones mandadas en el trabajo y por tanto hay que obtener los errores del trabajo a través del método definido en este documento en el punto 1.2.4 y volver a enviar todas las prescripciones con las correcciones oportunas.
- 5- **Generado Parcialmente:** Este estado indica que se han procesado parte de las prescripciones que contiene el trabajo y algunas no han podido grabarse, por tanto hay que obtener los errores del trabajo a través del método definido en este documento en el punto 1.2.4 y volver a enviar todas las prescripciones con las correcciones oportunas.

10. Tabla de especies REGA

Tabla de especies REGA reconocidas por el sistema.

Identificador	Nombre
1	Bovidos
2	Cerdos
3	Ovino
4	Caprino
5	Équidos
6	Gallinas
7	Pavos
8	Pintadas
9	Patos
10	Ocas
11	Codornices
12	Palomas
13	Faisanes
14	Perdices
15	Ratites
16	Conejos
17	Liebres
18	Abejas
19	Visón
20	Zorro rojo
21	Nutria
22	Chinchilla
23	Ciervos
24	Corzos
25	Gamos
26	Jabalíes
27	Caracoles
28	Ranas
29	Dorada (<i>Spaus aurata</i>)
30	Lubina o Robalo (<i>Dicentrarchus labrax</i>)
31	Rodaballo (<i>Scophtamus maximus</i>)
32	Salmón del Atlántico (<i>Salmo salar</i>)
33	Atunes
34	Anguila (<i>Anguilla anguilla</i>)

35	Tilapias (<i>Oreochromis</i> spp.)
36	Camarón (<i>Palaemon serratus</i>)
37	Langostino tigre (<i>Penaeus japonicus</i>)
38	Cangrejo de las marismas (<i>Procambarus clarkii</i>)
39	Cangrejo de California (<i>Pacifastacus leniusculus</i>)
40	Trucha común (<i>Salmo trutta</i>)
41	Tenca (<i>Tinca tinca</i>)
42	Esturión (<i>Acipenser sturio</i>)
43	Carpa Comun (<i>Cyprinus carpio carpio</i>)
44	Mejillón atlántico (<i>Mytilus edulis</i>)
45	Almeja fina (<i>Ruditapes decussatus</i>)
46	Ostra (<i>Ostrea edulis</i>)
47	Berberecho (<i>Cerastoderma edule</i>)
54	Pez de limón (<i>Seriola dumerili</i>)
55	Zamburiña (<i>Chlamys varia</i>)
56	Baila (<i>Dicentrarchus punctatus</i>)
57	Corvina (<i>Argyrosomus regius</i>)
58	Sargo (<i>Diplodus sargus</i>)
59	Vieira o venera (<i>Pecten maximus</i>)
60	Almeja babosa o chocha (<i>Venerupis pullastra</i>)
61	Almeja japonesa (<i>Ruditapes philippinarum</i>)
62	Atún rojo o de aleta azul (<i>Thunnus thynnus</i>)
63	Peces ornamentales
64	Breca (<i>Pagellus erythrinus</i>)
65	Burro o roncador (<i>Plectorhinchus mediterraneus</i>)
67	Anfibios
68	Primates
69	Roedores
70	Alpacas
71	Llamas
73	Saurios (cocodrilos)
80	Perros
81	Gatos
82	Muflones
83	Lenguado europeo (<i>Solea solea</i> , <i>S. vulgaris</i>)
84	Coquina (<i>Donax trunculus</i>)
85	Chirla (<i>Chamelea gallina</i>)
86	Pulpo (<i>Octopus vulgaris</i>)
87	Aves cautivas y ornamentales

88	Dromedario
89	Mugil (<i>Mugil cephalus</i>)
90	Mejillón del Mediterráneo/Clóchina
91	Abadejo (<i>Pollachius pollachius</i>)
92	Besugo (<i>Pagellus bogaraveo</i>)
94	Camellos
100	Dentón o dentón europeo (<i>Dentex dentex</i>)
101	Especies de estero
102	Langostino mediterráneo (<i>Penaeus kerathurus</i>)
103	Lenguado senegalés (<i>Solea senegalensis</i>)
104	Mujoles (<i>Mugil spp.</i>)
105	Mero (<i>Epinephelus marginatus</i>)
106	Mojarra (<i>Diplodus vulgaris</i>)
107	Morragute (<i>Iiza ramada</i>)
108	Ostión u ostra japonesa (<i>Crassostrea gigas</i>)
109	Pargo (<i>Pagrus pagrus</i>)
110	Salema (<i>Sarpa salpa</i>)
111	Salmón real (<i>Oncorhynchus tshawytscha</i>)
112	Sargo picudo (<i>Diplodus puntazzo</i>)
113	Urta (<i>Pagrus auriga</i>)
114	Escupiña grabada (<i>Venus verrucosa</i>)
115	Navaja (<i>Ensis arcuatus</i>)
116	Pectínidos (<i>Peines spp.</i>)
117	Volandeira (<i>Aequipecten opercularis</i>)
118	Cangrejo autóctono (<i>Austropotamobius pallipes</i>)
119	Crustáceos ornamentales
120	Moluscos ornamentales
121	Camarón (<i>Palaemonetes varians</i>)
122	Cherna (<i>Polyprion americanus</i>)
123	Esturión del Adriático (<i>Acipenser naccarii</i>)
124	Esturión siberiano (<i>Acipenser Baerii</i>)
125	Trucha arco iris
126	Carpa real (<i>Cyprinus carpio royal</i>)
127	Acedía (<i>Dicologlossa cuneata</i>)
128	Medregal limón (<i>Seriola rivoliana</i>)
129	Medregal listado (<i>Seriola fasciata</i>)
130	Medregal de guinea (<i>Seriola carpenteri</i>)
131	Mejillón canario (<i>Perna perna</i>)
132	Salmón del Danubio (<i>Hucho hucho</i>)

133	Langostino blanco (<i>Litopenaeus vannamei</i>)
134	Cangrejo verde o coñeta (<i>Carcinus maenas</i>)
135	Pejerrey mediterráneo (<i>Atherina Boyeri</i>)
136	Fúndulus (<i>Fundulus heteroclitus</i>)
137	Esturión estrellado (<i>Acipenser stellatus</i>)
138	Esturión ruso (<i>Acipenser gueldenstaedti</i>)
139	Beluga (<i>Huso huso</i>)
140	Sardina (<i>Sardina pilchardus</i>)
141	Choco o sepia (<i>Sepia officinalis</i>)
142	Cobia (<i>Rachycentron canadum</i>)
143	Lamprea (<i>Petromyzon marinus</i>)
144	Oreja de mar o Abalón (<i>Haliotis tuberculata</i>)
145	Camarón Báltico (<i>Palaemon adspersus</i>)
146	Camarón o esquila (<i>Palaemon elegans</i>)
147	Abejorros (<i>Bombus terrestris</i>)
148	Garra Rufa (Pez doctor)
150	Erizo de mar (<i>Paracentrotus lividus</i>)
151	Ortiguillas (<i>Anemonia sulcata</i>)
152	Vargula hilgendorffii
153	Centolla (<i>Maja squinado</i>)
154	Nécora (<i>Necora púber</i>)
155	Caballito de mar (<i>Hippocampus guttulatus</i>)
156	Pez espinoso (<i>Gasterosteus aculeatus</i>)
157	Esturión esterlete (<i>Acipenser ruthenus</i>)
158	Lubrigante (<i>Homarus gammarus</i>)
159	Santiaguíño (<i>Scyllarus arctus</i>)
165	Moluscos bivalvos (<i>Politapes rhomboides</i> y otros)
200	Animales de experimentación
232	Hurones (<i>Mustela putoris furo</i>)
233	Otros carnívoros (<i>Carnivora</i>)
243	Otras aves
245	Lagartos y Serpientes (<i>Squamata</i>)
246	Tortugas (<i>Chelonia</i>)
281	Otros peces
285	Mamíferos acuáticos
286	Otros herbívoros
287	Mamíferos voladores
288	Cocodrilianos
289	Rinocéfalos

290	Úrsidos
291	Aves rapaces
292	Artrópodos
293	Aves psitaciformes
294	Otros omnívoros
295	Cebo vivo
300	Varios moluscos de centros de expedición, depuración y similares

11. Tabla de relación de especies Nomenclátor-Rega

Tabla de relación de especies Nomenclátor-REGA reconocidas por el sistema.

Identificador Especie Nomenclátor	Nombre Especie Nomenclátor	Identificador Especie Rega	Nombre Especie Rega
17	Vacas lecheras	1	Bovidos
18	Vacas en lactación	1	Bovidos
19	Vacas en secado	1	Bovidos
20	Bovino de carne	1	Bovidos
21	Terneros prerrumiantes	1	Bovidos
22	Bovino rumiante	1	Bovidos
23	Vacas reproductoras	1	Bovidos
24	Toros reproductores	1	Bovidos
25	Terneros	1	Bovidos
1	Bovino	1	Bovidos
75	Vacas gestantes	1	Bovidos
87	Todas las especies	1	Bovidos
95	MAMIFEROS DE PRODUCCION (Animales de Granja)*	1	Bovidos
97	Rumiantes	1	Bovidos
98	Vacas	1	Bovidos
132	Novillas	1	Bovidos
192	Bovino, incluidos los búfalos	1	Bovidos
193	Bovinos salvajes	1	Bovidos
194	Bucerros	1	Bovidos
195	Bueyes adultos	1	Bovidos
196	Bueyes jóvenes	1	Bovidos
197	Búfalas	1	Bovidos
198	Búfalos	1	Bovidos
199	Búfalos macho	1	Bovidos
190	Bisontes	1	Bovidos
281	Novillas gestantes	1	Bovidos
289	Otros bóvidos	1	Bovidos
346	Rumiantes y porcinos	1	Bovidos
350	Sin especificar	1	Bovidos
352	Terneros lactantes	1	Bovidos
353	Terneros recién nacidos	1	Bovidos
360	Toros	1	Bovidos

371	Todos los rumiantes	1	Bovidos
400	Bovinos jóvenes	1	Bovidos
401	Novillos	1	Bovidos
402	Cerdos jóvenes	2	Cerdos
403	Cerdos adultos	2	Cerdos
404	Cerdas en lactación	2	Cerdos
405	Verracos jóvenes	2	Cerdos
350	Sin especificar	2	Cerdos
346	Rumiantes y porcinos	2	Cerdos
304	Otros porcinos	2	Cerdos
261	Lechones lactantes	2	Cerdos
262	Lechones recién nacidos	2	Cerdos
215	Cerdos de reposición	2	Cerdos
216	Cerdos macho	2	Cerdos
144	Cerdas	2	Cerdos
145	Verracos	2	Cerdos
146	Cerdas jóvenes	2	Cerdos
147	Cerdas nulíparas	2	Cerdos
164	Cerdas adultas	2	Cerdos
115	Lechones destetados	2	Cerdos
103	Cerdas multíparas	2	Cerdos
104	Cerdas primíparas	2	Cerdos
95	MAMIFEROS DE PRODUCCION (Animales de Granja)*	2	Cerdos
87	Todas las especies	2	Cerdos
77	Cerdas gestantes	2	Cerdos
7	Porcino	2	Cerdos
35	Cerdos de engorde	2	Cerdos
36	Lechones	2	Cerdos
37	Cerdas reproductoras	2	Cerdos
38	Cerdos reproductores	2	Cerdos
2	Ovino	3	Ovino
26	Ovejas lecheras	3	Ovino
27	Ovejas en lactación	3	Ovino
28	Ovejas en secado	3	Ovino
29	Ovino de carne	3	Ovino
30	Corderos prerrumiantes	3	Ovino
31	Ovino rumiante	3	Ovino
32	Corderos	3	Ovino

33	OVINO (REPRODUCTORAS)*	3	Ovino
76	Ovejas gestantes	3	Ovino
87	Todas las especies	3	Ovino
95	MAMIFEROS DE PRODUCCION (Animales de Granja)*	3	Ovino
97	Rumiantes	3	Ovino
166	Ovejas	3	Ovino
163	Ovejas gestantes	3	Ovino
227	Corderos lactantes	3	Ovino
209	Carneros	3	Ovino
302	Otros óvidos	3	Ovino
307	Ovejas salvajes	3	Ovino
346	Rumiantes y porcinos	3	Ovino
350	Sin especificar	3	Ovino
371	Todos los rumiantes	3	Ovino
375	Ovinos (reproductores)	3	Ovino
371	Todos los rumiantes	4	Caprino
372	Cabras	4	Caprino
350	Sin especificar	4	Caprino
346	Rumiantes y porcinos	4	Caprino
292	Otros caprinos	4	Caprino
272	Machos cabríos	4	Caprino
203	Cabras	4	Caprino
204	Cabras salvajes	4	Caprino
205	Cabritos lactantes	4	Caprino
97	Rumiantes	4	Caprino
95	MAMIFEROS DE PRODUCCION (Animales de Granja)*	4	Caprino
87	Todas las especies	4	Caprino
82	Cabras gestantes	4	Caprino
3	Caprino	4	Caprino
61	Cabras en lactación	4	Caprino
62	Cabras en secado	4	Caprino
63	Cabritos	4	Caprino
64	Cabritos prerrumiante	4	Caprino
65	Caprino Rumiante	4	Caprino
66	CAPRINO (REPRODUCTORES)*	4	Caprino

67	CAPRINO (REPRODUCTORAS)*	4	Caprino
49	Yeguas	5	Équidos
50	EQUIDOS (CABALLOS DE DEPORTE)*	5	Équidos
51	Burros	5	Équidos
52	Mulas	5	Équidos
53	Ponis	5	Équidos
4	Potros	5	Équidos
5	Caballos	5	Équidos
6	Equino	5	Équidos
87	Todas las especies	5	Équidos
78	Yeguas gestantes	5	Équidos
95	MAMIFEROS DE PRODUCCION (Animales de Granja)*	5	Équidos
91	Caballos no destinados a consumo humano	5	Équidos
88	ASNO*	5	Équidos
201	Caballos castrados	5	Équidos
202	Caballos enteros	5	Équidos
214	Cebras	5	Équidos
154	Équidos	5	Équidos
295	Otros équidos	5	Équidos
232	Équidos hembra	5	Équidos
233	Caballos destinados a consumo humano	5	Équidos
334	Potras	5	Équidos
335	Potras lactantes	5	Équidos
336	Potros lactantes	5	Équidos
337	Potros macho	5	Équidos
338	Potros macho lactantes	5	Équidos
350	Sin especificar	5	Équidos
370	Equinos	5	Équidos
373	Gallina-Pollitas futuras ponedoras	6	Gallinas
374	Gallina-Pollitas futuras reproductoras	6	Gallinas
406	Gallinas reproductoras	6	Gallinas
411	Pollitas de reposición	6	Gallinas
350	Sin especificar	6	Gallinas
333	Pollitas	6	Gallinas

240	Galliformes	6	Gallinas
241	Gallina india	6	Gallinas
87	Todas las especies	6	Gallinas
105	Huevos embrionados de gallina	6	Gallinas
99	Aves de corral	6	Gallinas
142	Aves (broilers ponedoras reproductoras y pavos) *	6	Gallinas
8	Gallinas	6	Gallinas
10	Gallinas ponedoras	6	Gallinas
39	Pollos reproductores	6	Gallinas
40	Pollos de reposición	6	Gallinas
42	Gallos	6	Gallinas
46	Pollitos	6	Gallinas
47	Pollos	6	Gallinas
48	Pollos de engorde	6	Gallinas
9	Pavos	7	Pavos
84	Pavos	7	Pavos
121	Pavos reproductores	7	Pavos
99	Aves de corral	7	Pavos
93	AVES (PAVAS FUTURAS REPRODUCTORAS)*	7	Pavos
87	Todas las especies	7	Pavos
152	Pavos de engorde	7	Pavos
165	Pavas	7	Pavos
313	Pavipollos	7	Pavos
314	Pavos macho	7	Pavos
407	Pavas reproductoras	7	Pavos
408	Pavas ponedoras	7	Pavos
409	Pavos futuros reproductores	7	Pavos
410	Pavas futuras ponedoras	7	Pavos
350	Sin especificar	7	Pavos
350	Sin especificar	8	Pintadas
87	Todas las especies	8	Pintadas
99	Aves de corral	8	Pintadas
125	Pintadas	8	Pintadas
99	Aves de corral	9	Patos
87	Todas las especies	9	Patos
45	Patos	9	Patos
308	Patitos	9	Patos

309	Pato de Berbería	9	Patos
310	Patos de engorde	9	Patos
311	Patos domésticos	9	Patos
312	Patos reproductores	9	Patos
350	Sin especificar	9	Patos
350	Sin especificar	10	Ocas
149	Ocas	10	Ocas
87	Todas las especies	10	Ocas
99	Aves de corral	10	Ocas
99	Aves de corral	11	Codornices
87	Todas las especies	11	Codornices
225	Codornices	11	Codornices
350	Sin especificar	11	Codornices
350	Sin especificar	12	Palomas
87	Todas las especies	12	Palomas
99	Aves de corral	12	Palomas
129	Palomas mensajeras	12	Palomas
100	OTRAS AVES (PALOMAS NO CONSUMO)*	12	Palomas
44	Palomas	12	Palomas
99	Aves de corral	13	Faisanes
112	Faisanes	13	Faisanes
87	Todas las especies	13	Faisanes
350	Sin especificar	13	Faisanes
350	Sin especificar	14	Perdices
87	Todas las especies	14	Perdices
99	Aves de corral	14	Perdices
124	Perdices	14	Perdices
99	Aves de corral	15	Ratites
87	Todas las especies	15	Ratites
185	Avestruces	15	Ratites
231	Emús	15	Ratites
350	Sin especificar	15	Ratites
350	Sin especificar	16	Conejos
412	Conejos de engorde	16	Conejos
151	Conejos no destinados a consumo humano	16	Conejos
226	Conejos enanos	16	Conejos
87	Todas las especies	16	Conejos

95	MAMIFEROS DE PRODUCCION (Animales de Granja)*	16	Conejos
123	LAGOMORFOS*	16	Conejos
16	Conejos	16	Conejos
58	Conejas reproductoras	16	Conejos
59	Gazapos	16	Conejos
81	Conejas gestantes	16	Conejos
137	Pequeños mamíferos	17	Liebres
92	OTROS ANIMALES DOMESTICOS*	17	Liebres
87	Todas las especies	17	Liebres
265	Liebres	17	Liebres
297	Otros lepóridos	17	Liebres
350	Sin especificar	17	Liebres
350	Sin especificar	18	Abejas
87	Todas las especies	18	Abejas
13	Abejas	18	Abejas
87	Todas las especies	19	Visón
109	Visones	19	Visón
137	Pequeños mamíferos	19	Visón
128	Animales para producción de piel	19	Visón
350	Sin especificar	19	Visón
350	Sin especificar	20	Zorro rojo
137	Pequeños mamíferos	20	Zorro rojo
110	Zorros	20	Zorro rojo
87	Todas las especies	20	Zorro rojo
87	Todas las especies	21	Nutria
92	OTROS ANIMALES DOMESTICOS*	21	Nutria
137	Pequeños mamíferos	21	Nutria
350	Sin especificar	21	Nutria
282	Nutrias	21	Nutria
350	Sin especificar	22	Chinchilla
137	Pequeños mamíferos	22	Chinchilla
117	Chinchillas	22	Chinchilla
87	Todas las especies	22	Chinchilla
87	Todas las especies	23	Ciervos
92	OTROS ANIMALES DOMESTICOS*	23	Ciervos

350	Sin especificar	23	Ciervos
218	Cérvidos	23	Ciervos
367	Venados	23	Ciervos
228	Corzos	24	Corzos
350	Sin especificar	24	Corzos
92	OTROS ANIMALES DOMESTICOS*	24	Corzos
87	Todas las especies	24	Corzos
87	Todas las especies	25	Gamos
92	OTROS ANIMALES DOMESTICOS*	25	Gamos
350	Sin especificar	25	Gamos
242	Gamos	25	Gamos
258	Jabalíes	26	Jabalíes
350	Sin especificar	26	Jabalíes
346	Rumiantes y porcinos	26	Jabalíes
92	OTROS ANIMALES DOMESTICOS*	26	Jabalíes
87	Todas las especies	26	Jabalíes
371	Todos los rumiantes	26	Jabalíes
87	Todas las especies	27	Caracoles
350	Sin especificar	27	Caracoles
350	Sin especificar	28	Ranas
341	Ranas	28	Ranas
87	Todas las especies	28	Ranas
87	Todas las especies	29	Dorada (Spaus aurata)
71	Dorada	29	Dorada (Spaus aurata)
350	Sin especificar	29	Dorada (Spaus aurata)
350	Sin especificar	30	Lubina o Robalo (Dicentrarchus labrax)
70	Lubina	30	Lubina o Robalo (Dicentrarchus labrax)
87	Todas las especies	30	Lubina o Robalo (Dicentrarchus labrax)
87	Todas las especies	31	Rodaballo (Scophtamus maximus)
69	Rodaballo	31	Rodaballo (Scophtamus maximus)
350	Sin especificar	31	Rodaballo (Scophtamus maximus)
350	Sin especificar	32	Salmón del Atlántico (Salmo salar)

347	Salmónidos hembra	32	Salmón del Atlántico (Salmo salar)
85	Salmón del Atlántico	32	Salmón del Atlántico (Salmo salar)
87	Todas las especies	32	Salmón del Atlántico (Salmo salar)
87	Todas las especies	33	Atunes
350	Sin especificar	33	Atunes
350	Sin especificar	34	Anguila (Anguilla anguilla)
173	Anguila	34	Anguila (Anguilla anguilla)
87	Todas las especies	34	Anguila (Anguilla anguilla)
87	Todas las especies	35	Tilapias (Oreochromis spp.)
350	Sin especificar	35	Tilapias (Oreochromis spp.)
350	Sin especificar	36	Camarón (Palaemon serratus)
87	Todas las especies	36	Camarón (Palaemon serratus)
87	Todas las especies	37	Langostino tigre (Penaeus japonicus)
350	Sin especificar	37	Langostino tigre (Penaeus japonicus)
350	Sin especificar	38	Cangrejo de las marismas (Procambarus clarkii)
87	Todas las especies	38	Cangrejo de las marismas (Procambarus clarkii)
87	Todas las especies	39	Cangrejo de California (Pacifastacus leniusculus)
350	Sin especificar	39	Cangrejo de California (Pacifastacus leniusculus)
350	Sin especificar	40	Trucha común (Salmo trutta)
87	Todas las especies	40	Trucha común (Salmo trutta)
364	Trucha común	40	Trucha común (Salmo trutta)
87	Todas las especies	41	Tenca (Tinca tinca)
350	Sin especificar	41	Tenca (Tinca tinca)

350	Sin especificar	42	Esturión (<i>Acipenser sturio</i>)
238	Esturión	42	Esturión (<i>Acipenser sturio</i>)
87	Todas las especies	42	Esturión (<i>Acipenser sturio</i>)
87	Todas las especies	43	Carpa Comun (<i>Cyprinus carpio carpio</i>)
211	Carpa común	43	Carpa Comun (<i>Cyprinus carpio carpio</i>)
212	Carpas	43	Carpa Comun (<i>Cyprinus carpio carpio</i>)
350	Sin especificar	43	Carpa Comun (<i>Cyprinus carpio carpio</i>)
350	Sin especificar	44	Mejillón atlántico (<i>Mytilus edulis</i>)
87	Todas las especies	44	Mejillón atlántico (<i>Mytilus edulis</i>)
87	Todas las especies	45	Almeja fina (<i>Ruditapes decussatus</i>)
350	Sin especificar	45	Almeja fina (<i>Ruditapes decussatus</i>)
350	Sin especificar	46	Ostra (<i>Ostrea edulis</i>)
87	Todas las especies	46	Ostra (<i>Ostrea edulis</i>)
87	Todas las especies	47	Berberecho (<i>Cerastoderma edule</i>)
350	Sin especificar	47	Berberecho (<i>Cerastoderma edule</i>)
350	Sin especificar	54	Pez de limón (<i>Seriola dumerili</i>)
87	Todas las especies	54	Pez de limón (<i>Seriola dumerili</i>)
87	Todas las especies	55	Zamburiña (<i>Chlamys varia</i>)
350	Sin especificar	55	Zamburiña (<i>Chlamys varia</i>)
350	Sin especificar	56	Baila (<i>Dicentrarchus punctatus</i>)
87	Todas las especies	56	Baila (<i>Dicentrarchus punctatus</i>)
87	Todas las especies	57	Corvina (<i>Argyrosomus regius</i>)

350	Sin especificar	57	Corvina (<i>Argyrosomus regius</i>)
350	Sin especificar	58	Sargo (<i>Diplodus sargus</i>)
87	Todas las especies	58	Sargo (<i>Diplodus sargus</i>)
87	Todas las especies	59	Vieira o venera (<i>Pecten maximus</i>)
350	Sin especificar	59	Vieira o venera (<i>Pecten maximus</i>)
350	Sin especificar	60	Almeja babosa o chocha (<i>Venerupis pullastra</i>)
87	Todas las especies	60	Almeja babosa o chocha (<i>Venerupis pullastra</i>)
87	Todas las especies	61	Almeja japonesa (<i>Ruditapes philippinarum</i>)
350	Sin especificar	61	Almeja japonesa (<i>Ruditapes philippinarum</i>)
350	Sin especificar	62	Atún rojo o de aleta azul (<i>Thunnus thynnus</i>)
87	Todas las especies	62	Atún rojo o de aleta azul (<i>Thunnus thynnus</i>)
87	Todas las especies	63	Peces ornamentales
126	Peces ornamentales	63	Peces ornamentales
350	Sin especificar	63	Peces ornamentales
350	Sin especificar	64	Breca (<i>Pagellus erythrinus</i>)
87	Todas las especies	64	Breca (<i>Pagellus erythrinus</i>)
87	Todas las especies	65	Burro o roncador (<i>Plectorhinchus mediterraneus</i>)
350	Sin especificar	65	Burro o roncador (<i>Plectorhinchus mediterraneus</i>)
350	Sin especificar	67	Anfibios
172	Anfibios	67	Anfibios
286	Otros anfibios	67	Anfibios
87	Todas las especies	67	Anfibios
87	Todas las especies	68	Primates
92	OTROS ANIMALES DOMESTICOS*	68	Primates

278	Monos	68	Primates
220	Chimpancés	68	Primates
350	Sin especificar	68	Primates
339	Primates	68	Primates
306	Otros roedores	69	Roedores
320	Pequeños roedores	69	Roedores
343	Ratones de campo	69	Roedores
350	Sin especificar	69	Roedores
213	Castores	69	Roedores
268	Lirones	69	Roedores
87	Todas las especies	69	Roedores
120	Jerbos	69	Roedores
118	Hámsteres	69	Roedores
116	Cobayas	69	Roedores
137	Pequeños mamíferos	69	Roedores
106	Roedores	69	Roedores
107	Ratas	69	Roedores
108	Ratones	69	Roedores
92	OTROS ANIMALES DOMESTICOS*	69	Roedores
87	Todas las especies	70	Alpacas
171	Alpacas	70	Alpacas
350	Sin especificar	70	Alpacas
350	Sin especificar	71	Llamas
269	Llamas	71	Llamas
87	Todas las especies	71	Llamas
87	Todas las especies	73	Saurios (cocodrilos)
136	Reptiles	73	Saurios (cocodrilos)
114	Reptiles para producción	73	Saurios (cocodrilos)
350	Sin especificar	73	Saurios (cocodrilos)
350	Sin especificar	80	Perros
323	Perras	80	Perros
324	Perras en lactación	80	Perros
326	Perros grandes	80	Perros
327	Perros macho	80	Perros
328	Perros macho reproductores	80	Perros
329	Perros medianos	80	Perros
330	Perros muy grandes	80	Perros
331	Perros muy pequeños	80	Perros

332	Perros pequeños	80	Perros
137	Pequeños mamíferos	80	Perros
92	OTROS ANIMALES DOMESTICOS*	80	Perros
87	Todas las especies	80	Perros
79	Perras gestantes	80	Perros
14	Perros	80	Perros
54	Cachorros	80	Perros
55	Perras reproductoras	80	Perros
56	Gatitos	81	Gatos
57	Gatas reproductoras	81	Gatos
15	Gatos	81	Gatos
80	Gatas gestantes	81	Gatos
87	Todas las especies	81	Gatos
92	OTROS ANIMALES DOMESTICOS*	81	Gatos
137	Pequeños mamíferos	81	Gatos
350	Sin especificar	81	Gatos
243	Gatas	81	Gatos
244	Gatas en lactación	81	Gatos
246	Gatos grandes	81	Gatos
247	Gatos macho	81	Gatos
248	Gatos medianos	81	Gatos
249	Gatos pequeños	81	Gatos
350	Sin especificar	82	Muflones
92	OTROS ANIMALES DOMESTICOS*	82	Muflones
87	Todas las especies	82	Muflones
87	Todas las especies	83	Lenguado europeo (Solea solea, S. vulgaris)
350	Sin especificar	83	Lenguado europeo (Solea solea, S. vulgaris)
350	Sin especificar	84	Coquina (Donax trunculus)
87	Todas las especies	84	Coquina (Donax trunculus)
87	Todas las especies	85	Chirla (Chamelea gallina)
350	Sin especificar	85	Chirla (Chamelea gallina)

350	Sin especificar	86	Pulpo (<i>Octopus vulgaris</i>)
87	Todas las especies	86	Pulpo (<i>Octopus vulgaris</i>)
74	Cefalópodos	86	Pulpo (<i>Octopus vulgaris</i>)
43	Aves ornamentales	87	Aves cautivas y ornamentales
87	Todas las especies	87	Aves cautivas y ornamentales
135	Pájaros ornamentales	87	Aves cautivas y ornamentales
127	Canarios	87	Aves cautivas y ornamentales
350	Sin especificar	87	Aves cautivas y ornamentales
350	Sin especificar	88	Dromedario
290	Otros camélidos	88	Dromedario
92	OTROS ANIMALES DOMESTICOS*	88	Dromedario
87	Todas las especies	88	Dromedario
87	Todas las especies	89	Mugil (<i>Mugil cephalus</i>)
350	Sin especificar	89	Mugil (<i>Mugil cephalus</i>)
350	Sin especificar	90	Mejillón del Mediterráneo/Clóchina
87	Todas las especies	90	Mejillón del Mediterráneo/Clóchina
87	Todas las especies	91	Abadejo (<i>Pollachius pollachius</i>)
350	Sin especificar	91	Abadejo (<i>Pollachius pollachius</i>)
350	Sin especificar	92	Besugo (<i>Pagellus bogaraveo</i>)
189	Besugo	92	Besugo (<i>Pagellus bogaraveo</i>)
87	Todas las especies	92	Besugo (<i>Pagellus bogaraveo</i>)
87	Todas las especies	94	Camellos
92	OTROS ANIMALES DOMESTICOS*	94	Camellos
208	Camellos	94	Camellos
350	Sin especificar	94	Camellos
350	Sin especificar	100	Dentón o dentón europeo (<i>Dentex dentex</i>)

87	Todas las especies	100	Dentón o dentón europeo (Dentex dentex)
87	Todas las especies	101	Especies de estero
350	Sin especificar	101	Especies de estero
350	Sin especificar	102	Langostino mediterráneo (Penaeus kerathurus)
87	Todas las especies	102	Langostino mediterráneo (Penaeus kerathurus)
87	Todas las especies	103	Lenguado senegalés (Solea senegalensis)
350	Sin especificar	103	Lenguado senegalés (Solea senegalensis)
350	Sin especificar	104	Mujoles (Mugil spp.)
87	Todas las especies	104	Mujoles (Mugil spp.)
87	Todas las especies	105	Mero (Epinephelus marginatus)
350	Sin especificar	105	Mero (Epinephelus marginatus)
350	Sin especificar	106	Mojarra (Diplodus vulgaris)
87	Todas las especies	106	Mojarra (Diplodus vulgaris)
87	Todas las especies	107	Morragute (liza ramada)
350	Sin especificar	107	Morragute (liza ramada)
350	Sin especificar	108	Ostión u ostra japonesa (Crassostrea gigas)
87	Todas las especies	108	Ostión u ostra japonesa (Crassostrea gigas)
87	Todas las especies	109	Pargo (Pagrus pagrus)
350	Sin especificar	109	Pargo (Pagrus pagrus)
350	Sin especificar	110	Salema (Sarpa salpa)
87	Todas las especies	110	Salema (Sarpa salpa)
87	Todas las especies	111	Salmón real (Oncorhynchus tshawytscha)
350	Sin especificar	111	Salmón real (Oncorhynchus tshawytscha)

350	Sin especificar	112	Sargo picudo (<i>Diplodus puntazzo</i>)
87	Todas las especies	112	Sargo picudo (<i>Diplodus puntazzo</i>)
87	Todas las especies	113	Urta (<i>pagrus auriga</i>)
350	Sin especificar	113	Urta (<i>pagrus auriga</i>)
350	Sin especificar	114	Escupiña grabada (<i>Venus verrucosa</i>)
87	Todas las especies	114	Escupiña grabada (<i>Venus verrucosa</i>)
87	Todas las especies	115	Navaja (<i>Ensis arcuatus</i>)
350	Sin especificar	115	Navaja (<i>Ensis arcuatus</i>)
350	Sin especificar	116	Pectínidos (<i>Peines spp.</i>)
87	Todas las especies	116	Pectínidos (<i>Peines spp.</i>)
87	Todas las especies	117	Volandeira (<i>Aequipecten opercularis</i>)
350	Sin especificar	117	Volandeira (<i>Aequipecten opercularis</i>)
350	Sin especificar	118	Cangrejo autóctono (<i>Austropotamobius pallipes</i>)
87	Todas las especies	118	Cangrejo autóctono (<i>Austropotamobius pallipes</i>)
87	Todas las especies	119	Crustáceos ornamentales
350	Sin especificar	119	Crustáceos ornamentales
350	Sin especificar	120	Moluscos ornamentales
87	Todas las especies	120	Moluscos ornamentales
87	Todas las especies	121	Camarón (<i>Palaemonetes varians</i>)
72	Crustáceos	121	Camarón (<i>Palaemonetes varians</i>)
350	Sin especificar	121	Camarón (<i>Palaemonetes varians</i>)
350	Sin especificar	122	Cherna (<i>Polyprion americanus</i>)
87	Todas las especies	122	Cherna (<i>Polyprion americanus</i>)

87	Todas las especies	123	Esturión del Adriático (Acipenser naccarii)
350	Sin especificar	123	Esturión del Adriático (Acipenser naccarii)
350	Sin especificar	124	Esturión siberiano (Acipenser Baerii)
87	Todas las especies	124	Esturión siberiano (Acipenser Baerii)
87	Todas las especies	125	Trucha arco iris
86	Truchas	125	Trucha arco iris
11	Salmónidos	125	Trucha arco iris
139	Trucha arcoiris	125	Trucha arco iris
350	Sin especificar	125	Trucha arco iris
363	Trucha alpina	125	Trucha arco iris
350	Sin especificar	126	Carpa real (Cyprinus carpio royal)
87	Todas las especies	126	Carpa real (Cyprinus carpio royal)
87	Todas las especies	127	Acedía (Dicologlossa cuneata)
350	Sin especificar	127	Acedía (Dicologlossa cuneata)
350	Sin especificar	128	Medregal limón (Seriola rivoliana)
87	Todas las especies	128	Medregal limón (Seriola rivoliana)
87	Todas las especies	129	Medregal listado (Seriola fasciata)
350	Sin especificar	129	Medregal listado (Seriola fasciata)
350	Sin especificar	130	Medregal de guinea (Seriola carpenteri)
87	Todas las especies	130	Medregal de guinea (Seriola carpenteri)
87	Todas las especies	131	Mejillón canario (Perna perna)
350	Sin especificar	131	Mejillón canario (Perna perna)
350	Sin especificar	132	Salmón del Danubio (Hucho hucho)
87	Todas las especies	132	Salmón del Danubio (Hucho hucho)
87	Todas las especies	133	Langostino blanco (Litopenaeus vannamei)

350	Sin especificar	133	Langostino blanco (Litopenaeus vannamei)
350	Sin especificar	134	Cangrejo verde o coñeta (Carcinus maenas)
87	Todas las especies	134	Cangrejo verde o coñeta (Carcinus maenas)
87	Todas las especies	135	Pejerrey mediterráneo (Atherina Boyeri)
350	Sin especificar	135	Pejerrey mediterráneo (Atherina Boyeri)
350	Sin especificar	136	Fúndulus (Fundulus heteroclitus)
87	Todas las especies	136	Fúndulus (Fundulus heteroclitus)
87	Todas las especies	137	Esturión estrellado (Acipenser stellatus)
350	Sin especificar	137	Esturión estrellado (Acipenser stellatus)
350	Sin especificar	138	Esturión ruso (Acipenser gueldenstaedti)
87	Todas las especies	138	Esturión ruso (Acipenser gueldenstaedti)
87	Todas las especies	139	Beluga (Huso huso)
350	Sin especificar	139	Beluga (Huso huso)
350	Sin especificar	140	Sardina (Sardina pilchardus)
87	Todas las especies	140	Sardina (Sardina pilchardus)
87	Todas las especies	141	Choco o sepia (Sepia officinalis)
74	Cefalópodos	141	Choco o sepia (Sepia officinalis)
350	Sin especificar	141	Choco o sepia (Sepia officinalis)
350	Sin especificar	142	Cobia (Rachycentron canadum)
87	Todas las especies	142	Cobia (Rachycentron canadum)
87	Todas las especies	143	Lamprea (Petromyzun marinus)
350	Sin especificar	143	Lamprea (Petromyzun marinus)

350	Sin especificar	144	Oreja de mar o Abalón (Haliotis tuberculata)
87	Todas las especies	144	Oreja de mar o Abalón (Haliotis tuberculata)
87	Todas las especies	145	Camarón Báltico (Palaemon adspersus)
350	Sin especificar	145	Camarón Báltico (Palaemon adspersus)
350	Sin especificar	146	Camarón o esquila (Palaemon elegans)
87	Todas las especies	146	Camarón o esquila (Palaemon elegans)
87	Todas las especies	147	Abejorros (Bombus terrestris)
13	Abejas	147	Abejorros (Bombus terrestris)
350	Sin especificar	147	Abejorros (Bombus terrestris)
350	Sin especificar	148	Garra Rufa (Pez doctor)
87	Todas las especies	148	Garra Rufa (Pez doctor)
87	Todas las especies	150	Erizo de mar (Paracentrotus lividus)
350	Sin especificar	150	Erizo de mar (Paracentrotus lividus)
350	Sin especificar	151	Ortiguillas (Anemonia sulcata)
87	Todas las especies	151	Ortiguillas (Anemonia sulcata)
87	Todas las especies	152	Vargula hilgendorfii
350	Sin especificar	152	Vargula hilgendorfii
350	Sin especificar	153	Centolla (Maja squinado)
87	Todas las especies	153	Centolla (Maja squinado)
87	Todas las especies	154	Nécora (Necora púber)
350	Sin especificar	154	Nécora (Necora púber)
350	Sin especificar	155	Caballito de mar (Hippocampus guttulatus)
87	Todas las especies	155	Caballito de mar (Hippocampus guttulatus)

87	Todas las especies	156	Pez espinoso (Gasterosteus aculeatus)
350	Sin especificar	156	Pez espinoso (Gasterosteus aculeatus)
350	Sin especificar	157	Esturión esterlete (Acipenser ruthenus)
87	Todas las especies	157	Esturión esterlete (Acipenser ruthenus)
87	Todas las especies	158	Lubrigante (Homarus gammarus)
350	Sin especificar	158	Lubrigante (Homarus gammarus)
350	Sin especificar	159	Santiaguíño (Scyllarus arctus)
87	Todas las especies	159	Santiaguíño (Scyllarus arctus)
73	Moluscos	165	Moluscos bivalvos (Politapes rhomboides y otros)
191	Bivalvos	165	Moluscos bivalvos (Politapes rhomboides y otros)
300	Otros moluscos	165	Moluscos bivalvos (Politapes rhomboides y otros)
350	Sin especificar	200	Animales de experimentación
87	Todas las especies	200	Animales de experimentación
133	Animales de laboratorio	200	Animales de experimentación
137	Pequeños mamíferos	200	Animales de experimentación
137	Pequeños mamíferos	232	Hurones (Mustela putoris furo)
119	Hurones	232	Hurones (Mustela putoris furo)
92	OTROS ANIMALES DOMESTICOS*	232	Hurones (Mustela putoris furo)
87	Todas las especies	232	Hurones (Mustela putoris furo)
350	Sin especificar	232	Hurones (Mustela putoris furo)
350	Sin especificar	233	Otros carnívoros (Carnivora)

340	Pumas	233	Otros carnívoros (Carnivora)
325	Perro mapache	233	Otros carnívoros (Carnivora)
301	Otros mustélidos	233	Otros carnívoros (Carnivora)
296	Otros félicos	233	Otros carnívoros (Carnivora)
291	Otros cánidos	233	Otros carnívoros (Carnivora)
280	Mustélidos	233	Otros carnívoros (Carnivora)
275	Martas	233	Otros carnívoros (Carnivora)
245	Gato leopardo de Tsushima	233	Otros carnívoros (Carnivora)
234	Erizos	233	Otros carnívoros (Carnivora)
270	Lobos	233	Otros carnívoros (Carnivora)
259	Jaguares	233	Otros carnívoros (Carnivora)
266	Linces	233	Otros carnívoros (Carnivora)
267	Linces rojos	233	Otros carnívoros (Carnivora)
263	Leones	233	Otros carnívoros (Carnivora)
264	Leopardos	233	Otros carnívoros (Carnivora)
210	Carnívoros	233	Otros carnívoros (Carnivora)
219	Chacales	233	Otros carnívoros (Carnivora)
87	Todas las especies	233	Otros carnívoros (Carnivora)
92	OTROS ANIMALES DOMESTICOS*	233	Otros carnívoros (Carnivora)
365	Turones	233	Otros carnívoros (Carnivora)
354	Tigres	233	Otros carnívoros (Carnivora)
351	Tejones	233	Otros carnívoros (Carnivora)
361	Tórtolas	243	Otras aves
134	Pinzones	243	Otras aves

90	Otras aves	243	Otras aves
87	Todas las especies	243	Otras aves
229	Cuervos	243	Otras aves
223	Cisnes	243	Otras aves
183	Aves de caza	243	Otras aves
167	Agachadiza común	243	Otras aves
148	Aves	243	Otras aves
186	Avutardas	243	Otras aves
251	Gorriones	243	Otras aves
237	Estorninos	243	Otras aves
350	Sin especificar	243	Otras aves
350	Sin especificar	245	Lagartos y Serpientes (Squamata)
348	Serpientes	245	Lagartos y Serpientes (Squamata)
260	Lagartos	245	Lagartos y Serpientes (Squamata)
257	Iguanas	245	Lagartos y Serpientes (Squamata)
305	Otros reptiles	245	Lagartos y Serpientes (Squamata)
176	Animales de terrario	245	Lagartos y Serpientes (Squamata)
87	Todas las especies	245	Lagartos y Serpientes (Squamata)
87	Todas las especies	246	Tortugas (Chelonia)
131	Tortugas	246	Tortugas (Chelonia)
350	Sin especificar	246	Tortugas (Chelonia)
362	Tortugas terrestres	246	Tortugas (Chelonia)
350	Sin especificar	281	Otros peces
349	Serránidos	281	Otros peces
285	Osteíctios	281	Otros peces
315	Peces de agua dulce	281	Otros peces
316	Peces hembra	281	Otros peces
317	Peces marinos	281	Otros peces
318	Peces reproductores	281	Otros peces
174	Animales de acuicultura	281	Otros peces
187	Bacalao	281	Otros peces
303	Otros peces	281	Otros peces
255	Huevas	281	Otros peces
239	Fletán	281	Otros peces

222	Ciprínidos	281	Otros peces
94	Huevos embrionados de trucha	281	Otros peces
87	Todas las especies	281	Otros peces
68	Peces productores de carne	281	Otros peces
12	Peces	281	Otros peces
87	Todas las especies	285	Mamíferos acuáticos
92	OTROS ANIMALES DOMESTICOS*	285	Mamíferos acuáticos
299	Otros mamíferos marinos	285	Mamíferos acuáticos
188	Ballenas	285	Mamíferos acuáticos
230	Delfines	285	Mamíferos acuáticos
350	Sin especificar	285	Mamíferos acuáticos
350	Sin especificar	286	Otros herbívoros
346	Rumiantes y porcinos	286	Otros herbívoros
344	Rebecos	286	Otros herbívoros
345	Renos	286	Otros herbívoros
221	Ciervos de cola blanca	286	Otros herbívoros
218	Cérvidos	286	Otros herbívoros
182	Artiodáctilos	286	Otros herbívoros
180	Antílopes	286	Otros herbívoros
169	Alces	286	Otros herbívoros
293	Otros cérvidos	286	Otros herbívoros
294	Otros cérvidos	286	Otros herbívoros
274	Marsupiales	286	Otros herbívoros
92	OTROS ANIMALES DOMESTICOS*	286	Otros herbívoros
96	Mamíferos no destinados a la producción	286	Otros herbívoros
97	Rumiantes	286	Otros herbívoros
130	Cérvidos	286	Otros herbívoros
87	Todas las especies	286	Otros herbívoros
366	Uapitíes	286	Otros herbívoros
371	Todos los rumiantes	286	Otros herbívoros
87	Todas las especies	287	Mamíferos voladores
92	OTROS ANIMALES DOMESTICOS*	287	Mamíferos voladores
279	Murciélagos	287	Mamíferos voladores
350	Sin especificar	287	Mamíferos voladores
350	Sin especificar	288	Cocodrilianos
224	Cocodrilos	288	Cocodrilianos

114	Reptiles para producción	288	Cocodrilianos
87	Todas las especies	288	Cocodrilianos
87	Todas las especies	289	Rinocéfalos
92	OTROS ANIMALES DOMESTICOS*	289	Rinocéfalos
350	Sin especificar	289	Rinocéfalos
350	Sin especificar	290	Úrsidos
283	Osos	290	Úrsidos
284	Osos panda	290	Úrsidos
92	OTROS ANIMALES DOMESTICOS*	290	Úrsidos
87	Todas las especies	290	Úrsidos
87	Todas las especies	291	Aves rapaces
276	Milano negro	291	Aves rapaces
277	Milanos	291	Aves rapaces
250	Gavilán tachiro	291	Aves rapaces
254	Halcón peregrino	291	Aves rapaces
217	Cernícalos	291	Aves rapaces
168	Águilas	291	Aves rapaces
200	Búhos	291	Aves rapaces
350	Sin especificar	291	Aves rapaces
342	Ratoneros	291	Aves rapaces
350	Sin especificar	292	Artrópodos
181	Arañas	292	Artrópodos
288	Otros artrópodos	292	Artrópodos
87	Todas las especies	292	Artrópodos
87	Todas las especies	293	Aves psitaciformes
253	Guacamayos	293	Aves psitaciformes
271	Loros	293	Aves psitaciformes
206	Cacatúa ninfa	293	Aves psitaciformes
207	Cacatúas	293	Aves psitaciformes
350	Sin especificar	293	Aves psitaciformes
321	Periquitos	293	Aves psitaciformes
350	Sin especificar	294	Otros omnívoros
87	Todas las especies	294	Otros omnívoros
92	OTROS ANIMALES DOMESTICOS*	294	Otros omnívoros
87	Todas las especies	295	Cebo vivo
350	Sin especificar	295	Cebo vivo

350	Sin especificar	300	Varios moluscos de centros de expedición, depuración y similares
87	Todas las especies	300	Varios moluscos de centros de expedición, depuración y similares