

Api de los Web's Services del Document Warehouse

Consideraciones generales

Los servicios Web's del Document Warehouse, siempre reciben parámetros en forma de nodos de XML, dentro de los cuales se encuentran los campos y valores esperados por el servicio.

Dichos servicios siempre retornarán nodos de XML dentro de los cuales se encontrarán en el nodo raíz un campo que indicará el estatus de la ejecución del servicio y un mensaje de retorno.

Las únicas excepciones a esta regla se encuentran en aquellos servicios que por su naturaleza no poseen parámetros de entrada o de salida.

Como llamar al servicio

El servicio Web del Document Warehouse, se debe llamar por medio de la dirección URL indicando <http://NombreServidor/DVService/DVServDW01.aspx>, por ejemplo la llamada para un servidor que se llame server2 sería la siguiente:

<http://server2/DVService/DVServDW01.aspx>

Estructura básica del XML de entrada

Todos los nodos de XML que recibe cualquiera de los servicios Web's del Document Warehouse, deben indicar los siguientes parámetros:

Sistema: indica el nombre del sistema que se está utilizando, generalmente es la palabra "DocumentWarehouse"

Usuario: indica el nombre del usuario que realiza la conexión con el Document Warehouse

Password: indica la clave del usuario del Document Warehouse

Token: indica el "User Session ID" de Microsoft Internet Information Server, el identifica a cada usuario conectado y cambia cada vez que un usuario se conecte con el servidor, este valor es retornado por el servicio "Login" y debe ser pasado a todos los demás para lograr la autenticación por parte del mismo.

Por ejemplo la cadena de caracteres correspondiente al servicio GetUserProperties:

```
<GetUserProperties>
  <sistema>DocumentWarehouse</sistema>"
  <usuario>jaraujo</usuario>"
  <password>jose01</password>"
  <user>jaraujo</user>"
  <token>2jthq155wgunjtyhx5sha345</token>"
</GetUserProperties>"
```

Estructura básica del XML de Salida

Todos los nodos de XML que se reciben de los servicios Web's del Document Warehouse indicarán una estructura mínima necesaria con el estado de la ejecución del servicio. Especificando los siguientes parámetros de retorno:

Status: Indica si ha ocurrido o no algún error en la ejecución del servicio, si es mayor que cero indicará algún grado de error, si es cero la ejecución correcta del servicio y en el caso de ser negativo indicará que la ejecución ha sido correcta pero sin conseguir valores para retornar al cliente.

Token: Indica el "User Session ID" de Microsoft Internet Information Server, el cual identifica a cada usuario conectado y cambia cada vez que el mismo se conecte con el servidor; este valor es retornado por el servicio "login" y debe ser pasado a todos los demás para lograr la autenticación por parte del mismo.

Mensaje: Indica el mensaje de error que retorna el servidor de los servicios cuando el status sea diferente de cero, si no su valor es igual a "" (null)

Por ejemplo la cadena de caracteres correspondiente al retorno del servicio **Login:**

```
<loginstatus xmlns="">
  <status>0</status>
  <token>2jthq155wgunjtyhx5sha345</token>
  <mensaje></mensaje>
</loginstatus>
```

Base de datos del Document Warehouse

El Document Warehouse para mantener su funcionalidad utiliza dos bases de datos, las cuales son:

DocWareHouse: posee las siguientes tablas que sirven para la configuración de la herramienta:

- Categoría: Contiene la información de las categorías de Documentos, es decir, la forma en que se agrupan de manera lógica los Documentos. Esta información es utilizada para obtener subconjuntos de tipos de Documentos.
- DocXTipDoc: Contiene la información de los estados de los Documentos ingresados en la herramienta y el tipo de Documento del mismo.
- SegXTipDoc: Contiene la información de los grupos a los cuales debe cambiar la permisología para que sea aprobado un Documento. Los accesos son:
 1. idmDSAccessNone = 0
 2. idmDSAccessViewer = 1
 3. idmDSAccessAuthor = 2

4. idmDSAccessOwner = 3
5. idmDSAccessAdmin = 4

- TipoDocumento: Contiene la información de los tipos de Documentos disponibles en Filenet Content Management, sirve de relación entre las clases de Filenet Content Management y las categorías de Documentos e indica la descripción de las clases de Filenet Content Manager.
- TempMakerChecker: Contiene toda la información que ha sido modificada de los Documentos, la cual será revisada posteriormente para que pueda ser aprobada o rechazada por la persona correspondiente. En esta tabla se guarda el XML que contiene dicha información, la fecha en la que se modificó, el nombre del módulo del Document Warehouse que fue utilizado y el estado en el que se encuentra la acción realizada; es decir, en espera, aprobada o rechazada.

DVSeg: posee las tablas de seguridad de la herramienta, las cuales se indican a continuación:

- Aplicacion: Contiene la descripción de la aplicación que se está ejecutando.
- Opcion: Contiene la información que permite la construcción del menú de usuario de la interfaz del cliente Windows y Web
- OpcionUsuario: Contiene la información que sirve de relación entre los usuarios de Filenet Content Management y las opciones a las cuales tienen acceso los usuarios de la herramienta.
- Usuario: Contiene la información de los usuarios que no son almacenados por Filenet Content Manager
- Audittrail: Contiene la información del archivo de auditoria para el control del Document Warehouse

- RegSisExternos: Contiene la información del registro de los sistemas externos que pueden ser relacionados con el Document Warehouse

Servicios

GetNewUserProperties

Obtiene las propiedades de un usuario nuevo en Filenet Content Manager y retorna los campos que deben ser solicitados por parte de la interfaz para crear un nuevo usuario con el servicio **NewUserProperties**.

XML de entrada:

```
<UpdatePassword>
  <sistema>DocumentWarehouse</sistema>"
  <token>dki5ks55kkmffzy2eixdbau</token>
  <usuario>admin</usuario>
  <password></password>
  <newpassword>nuevopassword01</newpassword>
</UpdatePassword>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <newpassword> permite asignar o cambiar la clave al usuario de la conexión actual.

XML de salida:

```
<GetUserProperties xmlns="">
  <status>0</status>
  <mensaje />
  <Campo>
    <nombre>User Name</nombre>
    <EsModificable>False</EsModificable>
```

```

    <Data>jaraujo</Data>
    <TieneListaValores>False</TieneListaValores>
    <Longitud>32</Longitud>
    <PropertyNum>1793</PropertyNum>
    <IsRequired>True</IsRequired>
    <Valores></Valores>
  </Campo>
  <Campo>
    <nombre>Default Group</nombre>
    <EsModificable>True</EsModificable>
    <Data>General Users</Data>
    <TieneListaValores>True</TieneListaValores>
    <Longitud>32</Longitud>
    <PropertyNum>1797</PropertyNum>
    <IsRequired>True</IsRequired>
    <Valores>
      <valor>Administrators</valor>
      <valor>General Users</valor>
    </Valores>
  </Campo>
</GetUserProperties>
  
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Campo>: Agrupa las diferentes propiedades o campos para que un usuario nuevo ingrese los datos correspondientes al sistema; se puede repetir tantas veces como propiedades posea Filenet Content Manager y posee los siguientes tags:

<nombre>: Es el nombre que se le da a la propiedad o campo que se va a visualizar

<EsModificable>: Indica si la propiedad puede ser modificada o no

<Data>: Es la información que se va a mostrar dentro del campo

<TieneListaValores>: Indica si el campo tiene una lista de valores o no

<Longitud>: Indica la longitud del campo

<PropertyNum>: Indica el número de la propiedad

<IsRequired>: Indica si el campo es requerido o no

<Valores>: Indica el inicio de la lista de valores que tiene el campo y está compuesto por múltiples <valor>

<valor>: Indicará el valor a mostrar en la lista y será repetido tantas veces como valores tenga la lista

</Valores>

</Campo>

NewUserProperties

Crea las propiedades de un usuario en Document y retorna el status

XML de entrada:

<GetNewUserProperties>

<sistema>**DocumentWarehouse**</sistema>

<usuario>**admin**</usuario>

<token>**afpweebk0aqzoi55duwsds55**</token>

<password />

<Campo>

```
<nombre>User Name</nombre>
<EsModificable>True</EsModificable>
<Data>joanna</Data>
<TieneListaValores>False</TieneListaValores>
<Longitud>32</Longitud>
<PropertyNum>1793</PropertyNum>
<Valores />
</Campo>
<Campo>
  <nombre>Default Group</nombre>
  <EsModificable>False</EsModificable>
  <Data>General Users</Data>
  <TieneListaValores>True</TieneListaValores>
  <Longitud>32</Longitud>
  <PropertyNum>1797</PropertyNum>
  <Valores />
</Campo>
<GetNewUserProperties>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <campo> es repetido tantas veces como campos contenga el módulo de "Usuario Nuevo" y contiene las propiedades con los valores ingresados por el usuario, dichos tags están definidos en **GetNewUserProperties**; cada uno de estos valores están dentro de un Tag el cual representa una propiedad diferente.

XML de salida:

```
<NewUserProperties xmlns="">
<status>0</status>
<mensaje></mensaje>
</NewUserProperties>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetUsersList

Retorna una lista de los usuarios creados en IDM

XML de entrada:

```
<GetUserList>
  <token>hgouomyhfmdbw550pvut32l</token>
  <usuario>admin</usuario>
  <password></password>
  <sistema>DocumentWarehouse</sistema>
</GetUserList>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetUsersList xmlns="">
  <status>0</status>
  <mensaje />
  <Campos>
    <UserName>abenaim</UserName>
    <DateAdded>5/12/2004 3:45:29 PM</DateAdded>
    <Comment />
    <DefaultGroup>Administrators</DefaultGroup>
    <Enabled>Yes</Enabled>
  </Campos>
  <Campos>
    <UserName>Admin</UserName>
    <DateAdded>12/9/2003 4:17:12 PM</DateAdded>
```

```
<Comment>Initial User</Comment>  
<DefaultGroup>Administrators</DefaultGroup>  
<Enabled>Yes</Enabled>  
</Campos>  
</GetUsersList>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Campos>: Agrupa la información de todos los usuarios que se han ingresado al sistema los cuales serán mostrados en el módulo de "Administración de Usuarios", se repetirá por la cantidad de usuarios que existan y posee los siguientes tags:

```
<UserName>: Es el nombre del usuario  
<DateAdded>: Indica la fecha en que se ingresó dicho usuario  
<Comment />: Indica los comentarios señalados en el  
momento de ingresar al usuario  
<DefaultGroup>: Indica el grupo al que pertenece el usuario  
<Enabled>: Indica si el usuario puede ser eliminado del  
sistema o si sus propiedades pueden ser modificadas  
</Campos>
```

GetUserProperties

Obtiene de un usuario existente en Document sus propiedades y retorna el status

XML de entrada:

```
<GetUserList>
<sistema>DocumentWarehouse</sistema>
<usuario>admin</usuario>
<token>afpweebk0aqzoi55duwsds55</token>
<password></password>
<user>joanna</user>
</GetUserList>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <user> indica el nombre del usuario a quien se le están consultando las propiedades

XML de salida:

```
<GetUserProperties>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>afpweebk0aqzoi55duwsds55</token>
  <password />
  <Campo>
    <nombre>User Name</nombre>
    <EsModificable>True</EsModificable>
    <Data>joanna</Data>
    <TieneListaValores>False</TieneListaValores>
    <Longitud>32</Longitud>
    <PropertyNum>1793</PropertyNum>
    <Valores />
  </Campo>
  <Campo>
    <nombre>Default Group</nombre>
    <EsModificable>False</EsModificable>
```

```
<Data>General Users</Data>
<TieneListaValores>True</TieneListaValores>
<Longitud>32</Longitud>
<PropertyNum>1797</PropertyNum>
<Valores />
</Campo>
<GetUserProperties>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's y los demás están definidos en el servicio **GetNewUserProperties**.

UpdateUserProperties

Actualiza las propiedades de un usuario en Document y retorna el status

XML de entrada:

```
<GetUserProperties>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>afpweebk0aqzoi55duwsds55</token>
  <password />
  <Campo>
    <nombre>User Name</nombre>
    <EsModificable>True</EsModificable>
    <Data>joanna</Data>
    <TieneListaValores>False</TieneListaValores>
    <Longitud>32</Longitud>
    <PropertyNum>1793</PropertyNum>
    <Valores />
  </Campo>
  <Campo>
```

```
<nombre>Default Group</nombre>
<EsModificable>False</EsModificable>
<Data>General Users</Data>
<TieneListaValores>True</TieneListaValores>
<Longitud>32</Longitud>
<PropertyNum>1797</PropertyNum>
<Valores />
</Campo>
<GetUserProperties>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's y los demás están definidos en el servicio **GetNewUserProperties**.

XML de salida:

```
<UpdateUserProperties xmlns="">
<status>0</status>
<mensaje></mensaje>
</UpdateUserProperties>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

DeleteUser

Elimina un usuario en Document y retorna el status

XML de entrada:

```
<DeleteUser>
  <token>hgouomyhfmdbw550pvut32l</token>
  <usuario>admin</usuario>
  <password></password>
```

```
<sistema>DocumentWarehouse</sistema>
```

```
</DeleteUser>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
< DeleteUser xmlns="">
```

```
<status>0</status>
```

```
<mensaje></mensaje>
```

```
</ DeleteUser>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

UpdatePassword

Ejecuta el cambio de la clave del usuario conectado en Filenet Content Management y retorna el estatus de la acción.

XML de entrada:

```
<UpdatePassword>
```

```
  <sistema>DocumentWarehouse</sistema>"
```

```
  <token>dki5ks55kkmffzy2eiaxdbau</token>
```

```
  <usuario>admin</usuario>
```

```
  <password></password>
```

```
  <newpassword>nuevopassword01</newpassword>
```

```
</UpdatePassword>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <newpassword> permite asignar o cambiar la clave al usuario de la conexión actual.

XML de salida:

```
<UpdatePassword xmlns="">
  <status>0</status>
  <mensaje></mensaje>
</UpdatePassword>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetMenuUsuario

Retorna el menú de opciones del usuario que está ingresando al sistema

XML de entrada:

```
<menu>
  <sistema>DocumentWarehouse</sistema>
  <token>dki5ks55kkmffzy2eiaxdbau</token>
  <usuario>admin</usuario>
  <password></password>
  <newpassword>nuevopassword01</newpassword>
</menu>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <newpassword> permite asignar o cambiar la clave al usuario de la conexión actual.

XML de salida:

```
<menu xmlns="">
  <submenu>
    <Nombre>Expediente</Nombre>
    <Orden>0</Orden>
```

```
<URL_Web></URL_Web>
<Nivel>0</Nivel>
<Padre></Padre>
<CodOpcion>1</CodOpcion>
<item>
<Nombre>Digitalizar</Nombre>
<Orden>0</Orden>
<Archivo>DVGDigit.dll</Archivo>
<Clase>DV.Digit.Digitalizacion</Clase>
<URL_Web></URL_Web>
<Nivel>1</Nivel>
<Padre>1</Padre>
<CodOpcion>2</CodOpcion>
</item>
<item>
<Nombre>Aprobar</Nombre>
<Orden>1</Orden>
<Archivo>DVAprob.dll</Archivo>
<Clase>DV.Aprob.Aprobador</Clase>
<URL_Web></URL_Web>
<Nivel>1</Nivel>
<Padre>1</Padre>
<CodOpcion>7</CodOpcion>
</item>
</submenu>
<status>0</status>
<mensaje></mensaje>
</menu>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<submenu>: Agrupa la información de las diferentes opciones que se pueden mostrar en el menú, puede repetirse tantas veces como menús principales (padres) existan y posee los siguientes tags:

<Nombre>: Es el nombre que se mostrará en el menú

<Orden>: Indica el orden en el que deben aparecer las opciones

<URL_Web>: Indica la dirección URL que será llamada al seleccionar la opción que indica el nombre anterior. Este tag solo tendrá valores cuando se esté cargando el menú para el menú Web

<Nivel>: Indica si la opción es padre (con valor "0", principal) o hijo (submenú)

<Padre>: Si la opción es un submenú (hijo), este tag indicará a que opción del menú principal pertenece, de lo contrario no tendrá valor

<CodOpcion>: Es un valor único que identifica el código de la opción

<Item>: Agrupa los valores indicados anteriormente pero de los submenús

```
<Nombre>Digitalizar</Nombre>
<Orden>0</Orden>
<Archivo>DVDigit.dll</Archivo>
<Clase>DV.Digit.Digitalizacion</Clase>
<URL_Web></URL_Web>
<Nivel>1</Nivel>
<Padre>1</Padre>
<CodOpcion>2</CodOpcion>
```

</Item>

</submenu>

GetIndexFind

Retorna las características de la búsqueda básica de la aplicación

XML de entrada:

No recibe parámetros de entrada

XML de salida:

```
<GetIndexFind xmlns="">
  <status>0</status>
  <mensaje>
  </mensaje>
  <IndicesBusqueda>
    <indice>
      <label>Cedula/Rif</label>
      <name>idmDocMvCustom1</name>
      <format>999.999.999</format>
    </indice>
  </IndicesBusqueda>
</GetIndexFind>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<IndicesBusqueda>: Agrupa las propiedades que le pertenecen al índice Cédula o Rif. Identificándose por los siguientes tags:

<indice>

<label>: Indica la descripción que se debe mostrar del índice.

<name>: Nombre físico del índice en Filenet Content Management.

<format>: Es el formato que acepta el índice

</indice>

</IndicesBusqueda>

GetListDocStat

Busca un Documento de la librería con un status cualquiera

XML de entrada:

<ObtenerExpediente>

<token>**hgouomyhfmdbw550pvut32l**</token>

<usuario>**admin**</usuario>

<sistema>**DocumentWarehouse**</sistema>

<status>**P**</status>

</ObtenerExpediente>

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <status> permite realizar un filtro por cualquiera de los campos que se encuentran en la tabla "TipoDocumento", el cual debe ser estructurado como la cláusula "where" de una búsqueda en SQL, por ejemplo:

XML de salida:

<GetListDocStat xmlns="">

<Documentos>

<TipoDocumento>**18**</TipoDocumento>

<ErrorFuente>**N**</ErrorFuente>

<Descripcion>**Referencia bancaria, comercial, personal o pago de servicios (luz, teléfono, gas, agua)**</Descripcion>

<Campo>

```
<name>IdmId</name>
<label>Identificador Idm</label>
<value>003670446</value>
</Campo>
<Campo>
  <name>idmVerId</name>
  <label>Versiones</label>
  <value>1</value>
</Campo>
<Campo>
  <name>idmDocMvCustom1</name>
  <label>Cedula/Rif</label>
  <value>V-82.133.341</value>
</Campo>
</Documentos>
<Documentos>
  <TipoDocumento>96</TipoDocumento>
  <ErrorFuente>N</ErrorFuente>
  <Descripcion>Novado</Descripcion>
  <Campo>
    <name>IdmId</name>
    <label>Identificador Idm</label>
    <value>003670451</value>
  </Campo>
</Documentos>
<status>0</status>
<mensaje>
</mensaje>
</GetListDocStat>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Documentos>: Agrupa los diferentes documentos que un usuario tiene digitalizado y guardado en Filenet Content Services; se puede repetir tantas veces como documentos contenga y posee los siguientes tags:

<TipoDocumento>: Es el código que identifica al documento

<ErrorFuente>: Indica si el documento presenta algún error

<Descripcion>: Es la descripción o nombre que se le ha dado al documento

<Campo>: Agrupa los campos y propiedades que contiene el documento, se puede repetir tantas veces como campos existan y posee los siguientes tags:

<name>: Indica el nombre original que tiene campo en el Document

<label>: Es el nombre del campo que se le va a mostrar al usuario

<value>: Es un valor único que identifica al campo

</Campo>

</Documentos>

PutListDocStat

Actualiza un Documento de la librería dependiendo del status que se le pase , es decir, si el status tiene como valor "A" quiere decir que es Aceptado y se puede actualizar, en caso contrario tendrá como valor "R" de Rechazado

XML de entrada:

<ObtenerDocs>

<sistema>**DocumentWarehouse**</sistema>

<usuario>**admin**</usuario>

<token>**u5bnjz45wopndn55eyolgqbo**</token>

<password />

```
<Documento>
  <IdmId>003670530</IdmId>
  <TipoDocumento>9</TipoDocumento>
  <status>A</status>
  <Observaciones />
</Documento>
</ObtenerDocs>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <Documento> contiene las propiedades que ya fueron descritas para el servicio **GetListDocStat** y son utilizadas para pasar como parámetros de entrada el IdmId del documento que se va a modificar, su código, un status cualquiera y las observaciones que se necesiten realizar

XML de salida:

```
< PutListDocStat xmlns="">
<status>0</status>
<mensaje></mensaje>
</ PutListDocStat >
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de la salida de los servicios Web's. Los valores de los nodos del XML de salida pueden variar según si aceptada o rechaza la acción del usuario; por lo tanto, si es aceptada el valor que se retornará será "0" y si es rechazada tendrá un valor mayor que "0", un ejemplo del último caso puede ser el siguiente:

```
<PutListDocStat xmlns="">
  <status>-2147157429</status>
  <mensaje>No Storage Managers are available.
    (BorrarDocumento)</mensaje>
</PutListDocStat>
```

GetDerechoImpresion

Obtiene si un usuario tiene derechos de impresión

XML de entrada:

```
<Login>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>u5bnjz45wopndn55eyolgqbo</token>
  <password />
</Login>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetDerechoImpresion xmlns="">
  <mensaje />
  <Impresion>N</Impresion>
  <status>0</status>
</GetDerechoImpresion>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's. <Impresion> indica si el usuario tiene derecho de impresión, en este caso "N" significa que no lo tiene.

IsDatExt

Retorna si la aplicación posee datos externos, si no posee el valor será "N" de lo contrario retornará "S"

GetIndexFindMedium

Retorna las características de la búsqueda media de la aplicación

XML de entrada:

No recibe parámetros de entrada

XML de salida:

```
<GetIndexFindMedium>
  <status>0</status>
  <mensaje />
  <IndicesBusquedaMedia>
    <indice>
      <label>Cedula/Rif</label>
      <name>idmDocMvCustom1</name>
      <format>999.999.999</format>
    </indice>
    <indice>
      <label>Apellido/Denominacion Juridica</label>
      <name>idmDocMvCustom2</name>
      <format />
    </indice>
    <indice>
      <label>Nombre</label>
      <name>idmDocCustom13</name>
      <format />
    </indice>
  </IndicesBusquedaMedia>
</GetIndexFindMedium>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos en el servicio **GetIndexFind**.

GetIndexFindAdvanced

Retorna las características de la búsqueda Avanzada de la aplicación

XML de entrada:

No recibe parámetros de entrada

XML de salida:

```
<GetIndexFindAdvanced>
  <status>0</status>
  <mensaje />
  <IndicesBusquedaAvanzada>
    <indice>
      <label>Cedula/Rif</label>
      <name>idmDocMvCustom1</name>
      <format>999.999.999</format>
    </indice>
    <indice>
      <label>Apellido/Denominacion Juridica</label>
      <name>idmDocMvCustom2</name>
      <format />
    </indice>
    <indice>
      <label>Nombre</label>
      <name>idmDocCustom13</name>
      <format />
    </indice>
    <indice>
      <label>Ubicacion Fisica</label>
      <name>idmDocCustom11</name>
      <format>999.999.999</format>
```

```
</indice>
<indice>
  <label>Digitalizado por</label>
  <name>idmDocOwner</name>
  <format>999.999.999</format>
</indice>
</IndicesBusquedaAvanzada>
</GetIndexFindAdvanced>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos en el servicio **GetIndexFind**.

GetDatosReporte

Retorna los datos del reporte de la aplicación

XML de entrada:

```
<ObtenerExpediente>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>wnf2if45dhu4h345luyvin45</token>
  <password />
  <Codigo>1</Codigo>
  <FechaDesde />
  <FechaHasta />
  <CedulaRIF1 />
  <CedulaRIF2 />
</ObtenerExpediente>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. Los demás tag's permiten realizar un filtro por

cualquiera de los campos que se encuentran en la tabla que tiene almacenado el Documento:

<Codigo>: Es el código que identifica al documento como valor único

<FechaDesde> y <FechaHasta>: Indican un rango de fechas desde-hasta para realizar las búsquedas

<CedulaRIF1> y <CedulaRIF2>: Indican un rango de números de cédulas o Rif. para realizar las búsquedas

XML de salida:

```
<GetDatosReporte xmlns="">
  <status>0</status>
  <mensaje />
  <Reporte>Error Fuente Externa Aprobados</Reporte>
  <Documentos>
    <TipoDocumento>83</TipoDocumento>
    <Descripcion>Nombramiento vigente de la Junta Directiva, debidamente inscrito en la Inspectoría del Trabajo</Descripcion>
    <ErrorFuente>S</ErrorFuente>
    <Campo>
      <name>IdmId</name>
      <label>Identificador Idm</label>
      <value>003670304</value>
    </Campo>
    <Estatus>A</Estatus>
    <Fecha>5/18/2004 12:22:12 PM</Fecha>
    <Observaciones />
  </Documentos>
</GetDatosReporte>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Reporte>: Indica la descripción del reporte que se va a generar

<Estatus>: Indica si el documento fue aprobado "A" o no "R"

<Fecha>: Indica la fecha cuando fue creado el documento

<Observaciones>: Indica si el documento tiene alguna observación de cuando fue creado

Los demás tag's están definidos en el servicio **GetListDocStat**

GetDatExt

Obtiene los datos de una base de datos y retorna el xml

XML de entrada:

<FindDocumentos>

<sistema>**DocumentWarehouse**</sistema>

<usuario>**admin**</usuario>

<token>**wnf2if45dhu4h345luyvin45**</token>

<password />

<Propiedad>

<Etiqueta>**idmDocMvCustom1**</Etiqueta>

<Valor>**v-06.339.695**</Valor>

</Propiedad>

</FindDocumentos>

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <Propiedad> contiene las propiedades del documento que se está consultando para obtener sus datos.

XML de salida:

```
<GetDatExt xmlns="">
  <status>-3</status>
  <mensaje />
</GetDatExt>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetTipoReporte

Retorna las características de los reportes de la aplicación

XML de entrada:

```
<login>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <password />
  <token>wnf2if45dhu4h345luyvin45</token>
</login>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetTipoReporte xmlns="">
  <status>0</status>
  <mensaje />
  <Reporte>
    <Codigo>1</Codigo>
    <Descripcion>Error Fuente Externa
      Aprobados</Descripcion>
    <Condicion>
```

```
<Campo>ErrorFuente</Campo>
<valor>S</valor>
</Condicion>
<Condicion>
  <Campo>Estatus</Campo>
  <valor>A</valor>
</Condicion>
</Reporte>
</GetTipoReporte>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Reporte>: Agrupa todas las característica que posee cada reporte; se puede repetir tantas veces como reportes existan y posee los siguientes tags:

```
<Codigo>: Es el código que identifica al reporte
<Descripcion>: Es el nombre que tiene el reporte
<Condicion>: Permite realizar un filtro para chequear si el
reporte tiene error en la fuente
  <Campo>: Indica el nombre del error
  <valor>: Es el valor que se está buscando en la condición
</Condicion>
```

GetIndexs

Retorna las características de la búsqueda básica de la aplicación

XML de entrada:

```
<login>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <password />
```

```
<token>wnf2if45dhu4h345luyvin45</token>  
<name>idmDocMvCustom1</name>  
</login>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <name> permite especificar el IdmId del documento que se está consultando

XML de salida:

```
<GetIndex xmlns="">  
  <status>0</status>  
  <mensaje />  
  <Campo>  
    <Name>idmDocMvCustom1</Name>  
    <Label>Cedula de Identidad</Label>  
    <Formato>V-XX.XXX.XXX</Formato>  
    <FormatoLabel>A-##.###.###</FormatoLabel>  
  </Campo>  
  <Campo>  
    <Name>idmDocMvCustom1</Name>  
    <Label>R.I.F.</Label>  
    <Formato>J-XX.XXX.XXX-X</Formato>  
    <FormatoLabel>J-##.###.###-#</FormatoLabel>  
  </Campo>  
</GetIndex>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Campo>: Agrupa las propiedades de los campos que se mostrarán en la interfaz

<Name>: Nombre físico del índice en Filenet Content Management.

<Label>: Indica la descripción que se debe mostrar del índice.

<Formato>: Es el formato que acepta el índice

<FormatoLabel>: Es el formato que debe tener la etiqueta que se va a mostrar en la interfaz

</Campo>

GetCategoria

Busca las categorías de Documentos

XML de entrada:

```
<GetCategorias>
  <token>hgouomyhfmdbw550pvut32l</token>
  <usuario>admin</usuario>
  <password></password>
  <sistema>DocumentWarehouse</sistema>
</GetCategorias>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetCategoria xmlns="">
  <status>0</status>
  <mensaje />
  <Categoria>
    <Codigo>1</Codigo>
    <Descripcion>Persona Natural</Descripcion>
  </Categoria>
  <Categoria>
```

```
<Codigo>2</Codigo>  
<Descripcion>Persona Juridica</Descripcion>  
</Categoria>  
</GetCategoria>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Categoria>: Agrupa la información de las diferentes categorías, se puede repetir tantas veces como tipos existan y posee los siguientes tags:

<Codigo>: Indica el código de la categoría.

<Descripcion>: Es la descripción de la categoría
</Categoria>

PutCategoria

Actualiza las categorías de los documentos

XML de entrada:

```
<PutCategorias>  
  <sistema>DocumentWarehouse</sistema>  
  <usuario>admin</usuario>  
  <token>wnf2if45dhu4h345luyvin45</token>  
  <password />  
  <Categoria>  
    <Codigo />  
    <Descripcion>Personales</Descripcion>  
  </Categoria>  
</PutCategorias>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <Categoria> agrupa el código y la descripción de la categoría que se esté consultando

XML de salida:

```
< GetCategoria xmlns="">
    <status>0</status>
    <mensaje></mensaje>
</GetCategoria >
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetCategoria_chk

Busca las categorías que se actualizarán luego de la aprobación de la persona correspondiente (Checker)

XML de entrada:

```
<GetCategorias>
    <token>hgouomyhfmdbwbw550pvut32l</token>
    <usuario>admin</usuario>
    <password></password>
    <sistema>DocumentWarehouse</sistema>
</GetCategorias>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetCategoria xmlns="">
    <status>0</status>
```

```
<mensaje />
<Categoria>
  <NumCaso>50</NumCaso>
  <Fecha>10/11/2004 11:54:40 AM</Fecha>
  <Codigo />
  <Descripcion> Personales </Descripcion>
</Categoria>
<Categoria>
  <NumCaso>51</NumCaso>
  <Fecha>10/11/2004 11:57:39 AM</Fecha>
  <Codigo />
  <Descripcion>Personal</Descripcion>
</Categoria>
</GetCategoria>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Categoria>: Agrupa información generada en el momento de realizar modificaciones a las categorías existentes para que puedan ser aceptadas o rechazadas con posterioridad, se repite tantas veces como casos hayan en espera; contiene los siguientes tag's:

<NumCaso>: Es un valor que identifica cada caso generado

<Fecha>: Indica la fecha y la hora en que se hicieron los cambios

<Codigo>: Indica el código de la categoría.

<Descripcion>: Es la descripción de la categoría

</Categoria>

PutCategoria_chk

Actualiza las categorías de los documentos (Cheker)

XML de entrada:

```
<PutCategorias>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>wnf2if45dhu4h345luyvin45</token>
  <password />
  <Categoria>
    <NumCaso>50</NumCaso>
    <Fecha>10/11/2004 11:54:40 AM</Fecha>
    <Codigo />
    <Descripcion>jyjhgjk</Descripcion>
    <Checker>A</Checker>
  </Categoria>
</PutCategorias>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

<Checker>: Indica si la solicitud ha sido aprobada "A" o no "R"

Los demás tag's están definidos en el servicio **GetCategoria_chk**.

XML de salida:

```
< GetCategoria xmlns="">
  <status>0</status>
  <mensaje></mensaje>
</GetCategoria >
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

FindDocumentos

Busca los Documentos que posee el usuario

XML de entrada:

```
<FindDocumentos>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>wnf2if45dhu4h345luyvin45</token>
  <password />
  <Propiedad>
    <Etiqueta>idmDocMvCustom1</Etiqueta>
    <Valor>v-06.339.695</Valor>
  </Propiedad>
</FindDocumentos>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <Propiedad> agrupa el IdmId y el valor del documento que se esté consultando

XML de salida:

```
<BuscaDocumentos xmlns="">
  <status>0</status>
  <mensaje />
  <Documentos>
    <TipoDocumento>1</TipoDocumento>
    <ErrorFuente>N</ErrorFuente>
    <Descripcion>No definido</Descripcion>
    <Campo>
```

```

    <name>IdmId</name>
    <label>Identificador Idm</label>
    <value>003670375</value>
  </Campo>
</Documentos>
<Documentos>
  <TipoDocumento>10</TipoDocumento>
  <ErrorFuente>N</ErrorFuente>
  <Descripcion>Verificacion de Datos</Descripcion>
  <Campo>
    <name>IdmId</name>
    <label>Identificador Idm</label>
    <value>003670528</value>
  </Campo>
</Documentos>
</BuscaDocumentos>

```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos en el servicio **GetListDocStat**.

GetImgDocumento_chk

Busca los accesos que se deberán asignar luego de la aprobación del 'aprobador' (Checker)

XML de entrada:

```

<GetImgDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>zddp1h45itun4xvewldzer45</token>
  <password />

```

</GetImgDocumento>

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetImgDocumento xmlns="">
  <status>0</status>
  <mensaje />
  <PutImgDocumento>
    <NumCaso>139</NumCaso>
    <Fecha>11/24/2004 10:35:54 AM</Fecha>
    <TipoDocumento>10</TipoDocumento>
    <Propiedad>
      <Etiqueta>idmDocMvCustom1</Etiqueta>
      <Valor>v-06.339.695</Valor>
    </Propiedad>
    <Propiedad>
      <Etiqueta>idmDocMvCustom2</Etiqueta>
      <Valor>Eduardo</Valor>
    </Propiedad>
    <IdmId>003670528</IdmId>
    <ErrorFuente>N</ErrorFuente>
    <FileName />
    <UpdateDoc>S</UpdateDoc>
    <Imagen />
  </PutImgDocumento>
  <PutImgDocumento>
    <NumCaso>140</NumCaso>
    <Fecha>11/24/2004 10:43:34 AM</Fecha>
    <TipoDocumento>10</TipoDocumento>
    <Propiedad>
```

```
<Etiqueta>idmDocMvCustom1</Etiqueta>  
<Valor>v-06.339.695</Valor>  
</Propiedad>  
<IdmId>003670528</IdmId>  
<ErrorFuente>N</ErrorFuente>  
<FileName />  
<UpdateDoc>S</UpdateDoc>  
<Imagen />  
</PutImgDocumento>  
</GetImgDocumento>
```

<PutImgDocumento>: Agrupa información generada en el momento de realizar modificaciones a los documentos existentes para que puedan ser aceptadas o rechazadas con posterioridad, se repite tantas veces como casos hayan en espera; contiene los siguientes tag's:

<NumCaso>: Es un valor que identifica cada caso generado

<Fecha>: Indica la fecha y la hora en que se hicieron los cambios

<TipoDocumento>: Indica el código que identifica al documento

<Propiedad>: agrupa el IdmId con su valor correspondiente

<IdmId>: Representa el IdmId del documento que se requiere la aprobación

<ErrorFuente>: Indica si el documento presenta algún error

<FileName>: Indica la ruta del archivo en el que se encuentra la imagen digitalizada

<UpdateDoc>: Indica que el documento será modificado

<Imagen>: Indica el nombre de la imagen digitalizada

</PutImgDocumento>

PutImgDocumento

Guarda la imagen de un Documento determinado

XML de entrada:

```
<PutImgDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>qnp1yeab4avet145tt4dzk45</token>
  <password />
  <TipoDocumento>10</TipoDocumento>
  <Propiedad>
    <Etiqueta>idmDocMvCustom1</Etiqueta>
    <Valor>v-06.339.695</Valor>
  </Propiedad>
  <Propiedad>
    <Etiqueta>idmDocMvCustom2</Etiqueta>
    <Valor>Eduardo</Valor>
  </Propiedad>
  <IdmId>003670528</IdmId>
  <ErrorFuente>N</ErrorFuente>
  <FileName />
  <UpdateDoc>S</UpdateDoc>
  <Imagen />
</PutImgDocumento>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada y los demás se encuentran definidos en el servicio **GetImgDocumento_chk**.

XML de salida:

```
<PutImgDocumento xmlns="">
  <status>0</status>
  <mensaje />
  <IdmId />
</PutImgDocumento>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, <IdmId> indica el IdmId del documento que se está guardando

GetSisRegExterno

Busca los sistemas externos con acceso al Document Warehouse

XML de entrada:

```
<GetRegSisExterno>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>zddp1h45itun4xvewldzer45</token>
  <password />
</GetRegSisExterno>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada

XML de salida:

```
<GetSisRegExterno xmlns="">
```

```
<status>0</status>
<mensaje />
<SisRegExterno>
  <Codigo>9</Codigo>
  <Descripcion>Prueba 2</Descripcion>
  <Estado>I</Estado>
</SisRegExterno>
</GetSisRegExterno>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<SisRegExterno>: Agrupa la información de las diferentes sistemas externos con acceso al Document Warehouse y posee los siguientes tags:

<Codigo>: Indica el código que representa al sistema

<Descripcion>: Es la descripción del sistema

<Estado>: Indica si el sistema se encuentra inactivo "I" o activo "A"

</SisRegExterno>

PutSisRegExterno

Actualiza los sistemas externos con derecho a acceder al Document Warehouse

XML de entrada:

```
<PutSisRegExterno>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>zddp1h45itun4xvewldzer45</token>
```

```
<password />
<SisRegExterno>
  <Codigo />
  <Descripcion>Prueba</Descripcion>
  <Estado>I</Estado>
</SisRegExterno>
</PutSisRegExterno>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada los demás están definidos en el XML de salida del servicio **GetSisRegExterno**.

XML de salida:

```
<PutSisRegExterno xmlns="">
  <status>0</status>
  <mensaje />
</PutSisRegExterno>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetSisRegExterno_chk

Busca los tipos de documentos que se actualizarán una vez que estén aprobados (Checker)

XML de entrada:

```
<GetSisRegExterno>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>njblzu45fkwrmo3whuvj0oq4</token>
  <password />
```

</GetSisRegExterno>

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

XML de salida:

```
<GetSisRegExterno xmlns="">
  <status>0</status>
  <mensaje />
  <SisRegExterno>
    <NumCaso>141</NumCaso>
    <Fecha>11/24/2004 1:58:10 PM</Fecha>
    <Codigo />
    <Descripcion>Prueba</Descripcion>
    <Estado>I</Estado>
  </SisRegExterno>
</GetSisRegExterno>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<SisRegExterno>: Agrupa información generada en el momento de realizar actualizaciones con respecto a los sistemas externos para que puedan ser aceptadas o rechazadas con posterioridad, se repite tantas veces como casos hayan en espera; contiene los siguientes tag's:

<NumCaso>: Es un valor que identifica cada caso generado

<Fecha>: Indica la fecha y la hora en que se hicieron los cambios

<Codigo>: Indica el código del sistema externo

<Descripcion>: Es la descripción del sistema externo

<Estado>: Indica si el sistema externo se encuentra inactivo "I"
o activo "A"

</SisRegExterno>

PutSisRegExterno_chk

Actualiza los sistemas externos con derecho a acceder al Document Warehouse (Checker)

XML de entrada:

```
<PutSisRegExterno>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>njblzu45fkwrmo3whuvj0oq4</token>
  <password />
  <SisRegExterno>
    <NumCaso>141</NumCaso>
    <Fecha>11/24/2004 1:58:10 PM</Fecha>
    <Codigo />
    <Descripcion>Prueba</Descripcion>
    <Estado>I</Estado>
    <Checker>A</Checker>
  </SisRegExterno>
</PutSisRegExterno>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada.

<SisRegExterno>: Agrupa información generada en el momento de realizar modificaciones a las categorías existentes para que puedan ser aceptadas o

rechazadas con posterioridad, se repite tantas veces como casos hayan en espera; contiene los siguientes tag's:

<NumCaso>: Es un valor que identifica cada caso generado

<Fecha>: Indica la fecha y la hora en que se hicieron los cambios

<Codigo>: Indica el código de la categoría.

<Descripcion>: Es la descripción de la categoría

</SisRegExterno>

XML de salida:

```
<PutSisRegExterno xmlns="">
```

```
  <status>0</status>
```

```
  <mensaje />
```

```
</PutSisRegExterno>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetGrupos

Obtiene todos los grupos a los que pueden pertenecer los usuarios del Document Warehouse

XML de entrada:

```
<GetSegXTipDoc>
```

```
  <sistema>DocumentWarehouse</sistema>
```

```
  <usuario>admin</usuario>
```

```
  <token>zivtvo45nj1umjqybmbm34aa</token>
```

```
<password />  
<SinIndices>True</SinIndices>  
</GetSegXTipDoc>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El tag </SinIndices> permite indicar si el XML de salida retornará o no los índices del documento consultado

XML de salida:

```
<GetGrupos xmlns="">  
  <status>0</status>  
  <mensaje />  
  <Grupos>  
    <Grupo>Administrators</Grupo>  
    <Grupo>Auditoria</Grupo>  
    <Grupo>Consulta1</Grupo>  
    <Grupo>Domain Admins</Grupo>  
  </Grupos>  
</GetGrupos>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<Grupos>: Agrupa la descripción de todos los grupos en el tag <Grupo> el cual puede repetirse tantas veces como sea necesario

GetSegXTipDoc

Busca los accesos que se deberán asignar luego que la persona correspondiente los apruebe

XML de entrada:

```
<GetSegXTipDoc>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>zivtvo45nj1umjqybmbm34aa</token>
  <password />
  <SinIndices>True</SinIndices>
</GetSegXTipDoc>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El tag </SinIndices> permite indicar si el XML de salida retornará o no los índices del documento consultado

XML de salida:

```
<GetSegXTipDoc xmlns="">
  <status>0</status>
  <mensaje />
  <SegXTipDoc>
    <TipoDocumento>2</TipoDocumento>
    <Grupo>Administrators</Grupo>
    <Acceso>1</Acceso>
  </SegXTipDoc>
  <SegXTipDoc>
    <TipoDocumento>2</TipoDocumento>
    <Grupo>Auditoria</Grupo>
    <Acceso>1</Acceso>
  </SegXTipDoc>
</GetSegXTipDoc>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<SegXTipDoc>: Agrupa la descripción de todos los grupos en el tag <Grupo> el cual puede repetirse tantas veces como sea necesario

PutSegXTipDoc

Actualiza los accesos que serán asignados una vez que se apruebe un documento por la persona correspondiente

XML de entrada:

```
<PutTipoDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>su05ja55matp5nfrgrie5d45</token>
  <password />
  <SegXTipDoc>
    <TipoDocumento>20</TipoDocumento>
    <Acceso>0</Acceso>
    <Grupo>OperadoresCS</Grupo>
    <Delete>False</Delete>
    <Insert>True</Insert>
  </SegXTipDoc>
</PutTipoDocumento>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada, los demás están definidos en el servicio **GetSegXTipDoc** excepto <Delete> e <Insert> los cuales permiten saber qué acción va a realizar; es decir:

<Delete>**False**</Delete> e <Insert>**True**</Insert>: la acción que se ejecutará es "agregar o insertar"

<Delete>**True**</Delete> e <Insert>**False**</Insert>: la acción que se ejecutará es "eliminar"

<Delete>**False**</Delete> e <Insert> **False** </Insert>>: la acción que se ejecutará es “modificar”

XML de salida:

```
<PutSegXTipDoc xmlns="">
  <status>0</status>
  <mensaje />
</PutSegXTipDoc>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's

PutSegXTipDoc_chk

Actualiza los accesos que serán asignados luego de ser aprobado el documento (Checker)

XML de entrada:

Actualiza los accesos que serán asignados luego que se apruebe un documento (Checker)

```
<PutTipoDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>su05ja55matp5nfrgrie5d45</token>
  <password />
  <SegXTipDoc>
    <NumCaso>2</NumCaso>
    <Fecha>10/4/2004 12:02:25 PM</Fecha>
    <TipoDocumento>12</TipoDocumento>
    <Acceso>2</Acceso>
    <Grupo>Consulta1</Grupo>
```

```
<Delete>False</Delete>
<Insert>True</Insert>
<Checker>A</Checker>
</SegXTipDoc>
</PutTipoDocumento>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada, los demás están definidos en el servicio **GetSegXTipDoc**. <Checker> indica si la solicitud ha sido aprobada "A" o rechazada "R"

XML de salida:

```
<PutSegXTipDoc xmlns="">
  <status>0</status>
  <mensaje />
</PutSegXTipDoc>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's

GetSegXTipDoc_Chk

Busca los accesos que se deberán asignar luego de la aprobación (Checker)

XML de entrada:

```
<GetSegXTipDoc_Chk>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>su05ja55matp5nfrgrie5d45</token>
  <password />
  <SinIndices>True</SinIndices>
</GetSegXTipDoc_Chk>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El tag `</SinIndices>` permite indicar si el XML de salida retornará o no los índices del documento consultado

XML de salida:

```
<GetSegXTipDoc xmlns="">
  <status>0</status>
  <mensaje />
  <SegXTipDoc>
    <NumCaso>2</NumCaso>
    <Fecha>10/4/2004 12:02:25 PM</Fecha>
    <TipoDocumento>12</TipoDocumento>
    <Acceso>2</Acceso>
    <Grupo>Consulta1</Grupo>
    <Delete>False</Delete>
    <Insert>True</Insert>
  </SegXTipDoc>
  <SegXTipDoc>
    <NumCaso>3</NumCaso>
    <Fecha>10/4/2004 1:29:41 PM</Fecha>
    <TipoDocumento>10</TipoDocumento>
    <Acceso>1</Acceso>
    <Grupo>Consulta2</Grupo>
    <Delete>False</Delete>
    <Insert>True</Insert>
  </SegXTipDoc>
</GetSegXTipDoc>
```

Los tag's `<status>` y `<mensaje>` fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<SegXTipDoc>: Agrupa la información generada en el momento de realizar las peticiones relacionadas a los accesos de los documentos para que puedan ser aceptadas o rechazadas con posterioridad, se repite tantas veces como casos hayan en espera; contiene los siguientes tag's:

<NumCaso>: Es un valor que identifica cada caso generado

<Fecha>: Indica la fecha y la hora en que se hicieron los cambios

<TipoDocumento>: Indica el código del documento que se le va a asignar algún acceso.

</SegXTipDoc>

Los demás tag's están definidos en el servicio **PutSegXTipDoc**.

GetTipoDocumento

Busca los Tipos de Documentos o categorías de las clases definidas en el Document Warehouse

XML de entrada:

```
<GetCategoria>
  <token>hgouomyhfmdbw550pvut32l</token>
  <usuario>admin</usuario>
  <password></password>
  <sistema>DocumentWarehouse</sistema>
  <condicion></condicion>
</GetCategoria>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <condicion> permite realizar un filtro por cualquiera de los campos que se encuentran en la tabla

“TipoDocumento”, el cual debe ser estructurado como la cláusula “where” de una búsqueda en SQL, por ejemplo:

```
<GetCategoria>
  <token>hgouomyhfmdbw550pvut32l</token>
  <usuario>admin</usuario>
  <password></password>
  <sistema>DocumentWarehouse</sistema>
  <condicion>CodCategoria=1</condicion>
</GetCategoria>
```

Este ejemplo solo retornará aquellos tipos de Documentos que pertenezcan a la categoría 1 la cual se encuentra definida en la tabla Categorías de la base de datos

XML de salida:

```
<GetTipoDocumento xmlns="">
  <status>0</status>
  <mensaje />
  <TipoDocumento>
    <CodTipDocumento>2</CodTipDocumento>
    <CodCategoria>1</CodCategoria>
    <Descripcion>Cedula de Identidad o Pasaporte</Descripcion>
    <ClaseDoc>Identificaciones</ClaseDoc>
    <Indices>
      <indice>
        <Label>Cedula/Rif</Label>
        <Name>idmDocMvCustom1</Name>
        <Size>16</Size>
        <DataType>String</DataType>
        <Req>si</Req>
        <HasDefault>False</HasDefault>
```

```
<DefaultValue />
<HasCVL>False</HasCVL>
<HasMultiValue>True</HasMultiValue>
<ReadOnly>False</ReadOnly>
</indice>
<indice>
  <Label>Ubicacion Fisica</Label>
  <Name>idmDocCustom11</Name>
  <Size>64</Size>
  <DataType>String</DataType>
  <Req>si</Req>
  <HasDefault>False</HasDefault>
  <DefaultValue />
  <HasCVL>False</HasCVL>
  <HasMultiValue>False</HasMultiValue>
  <ReadOnly>False</ReadOnly>
</indice>
</Indices>
</TipoDocumento>
</GetTipoDocumento>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos de la siguiente forma:

<TipoDocumento>: Agrupa la información de los diferentes tipos de Documentos, se puede repetir tantas veces como tipos existan y posee los siguientes tags:

<CodTipDocumento>: Indica el código del tipo de Documento que se encuentra en la tabla TipoDocumento de la base de datos.

<CodCategoria>: Código de la categoría a la que pertenece el Documento

<Descripcion>: descripción del tipo de Documento.

<ClaseDoc>: Clase a la que pertenece el tipo de Documento en IDM.

<Indices>: Indica el inicio de los índices del tipo de Documento y está compuesto por múltiples <indice>.

<indice>: indica el inicio de las características de un índice específico del tipo de Documento.

<Label>: Indica la descripción que se debe mostrar del índice.

<Name>: Nombre físico del índice en Filenet Content Management.

<Size>: Longitud del índice expresada en caracteres.

<DataType>: Tipo de dato.

<Req>: Indica si es obligatorio. Valores “si” o “no”

<HasDefault>: Indica si tiene un valor por defecto.

<DefaultValue>: Valor por defecto del índice.

<HasCVL>: Indica si es un “Controlled Vocabulary List”; los posibles valores son, “true” o “false”

<Choices>: agrupa los posibles valores que puede tomar un campo CVL y posee un tag <data> con el valor de cada ítem en un CVL.

<HasMultiValue>: Indica si es un índice multivaluado; valores posibles: “true” o “false”

<ReadOnly>: Indica si es un índice de solo lectura; valores posibles: "true" o "false"

</indice>

PutTipoDocumento

Actualiza los tipos de documentos

XML de entrada:

```
<PutTipoDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>su05ja55matp5nfrgrie5d45</token>
  <password />
  <TipoDocumento>
    <Descripcion>Novado</Descripcion>
    <ClaseDoc>Documentos de Garantias</ClaseDoc>
    <CodCategoria>15</CodCategoria>
    <CodTipDocumento>96</CodTipDocumento>
  </TipoDocumento>
</PutTipoDocumento>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada, los demás tag's están definidos en el XML de salida del servicio **GetTipoDocumento**.

XML de salida:

```
<PutTipoDocumento xmlns="">
  <status>0</status>
  <mensaje></mensaje>
</PutTipoDocumento>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

PutTipoDocumento_chk

Actualiza los tipos de documentos (Checker)

XML de entrada:

```
<PutTipoDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>su05ja55matp5nfrgrie5d45</token>
  <password />
  <TipoDocumento>
    <NumCaso>112</NumCaso>
    <Fecha>11/1/2004 5:00:52 PM</Fecha>
    <Descripcion>Entrevista de Nuevos Clientes</Descripcion>
    <ClaseDoc>Datos Basicos y Autorizaciones</ClaseDoc>
    <CodCategoria>1</CodCategoria>
    <CodTipDocumento>12</CodTipDocumento>
    <Checker>A</Checker>
  </TipoDocumento>
</PutTipoDocumento>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada, los demás están definidos de la siguiente forma:

<PutTipoDocumento>: Agrupa la información generada en el momento de realizar las peticiones relacionadas a las modificaciones de los tipos de documentos para que puedan ser aceptadas o rechazadas con posterioridad, se repite tantas veces como casos hayan en espera; contiene los siguientes tag's:

<NumCaso>: Es un valor que identifica cada caso generado

<Fecha>: Indica la fecha y la hora en que se hicieron los cambios

<TipoDocumento>: Indica el código del documento que se le va a asignar algún acceso.

</SegXTipDoc>

<Checker>: Indica si la solicitud ha sido aprobada "A" o no "R"

Los demás tag's están definidos en el servicio **GetTipoDocumento**.

XML de salida:

```
<PutTipoDocumento xmlns="">
  <status>0</status>
  <mensaje />
</PutTipoDocumento>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's.

GetTipoDocumento_chk

Busca los tipos de documentos que se actualizarán luego que se realice la aprobación (Checker)

XML de entrada:

```
<GetTipoDocumento>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <token>su05ja55matp5nfrgrie5d45</token>
  <password />
```

```
<SinIndices>True</SinIndices>  
</GetTipoDocumento>
```

Los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El tag </SinIndices> permite indicar si el XML de salida retornará o no los índices del documento consultado

XML de salida:

```
<GetTipoDocumento xmlns="">  
  <status>0</status>  
  <mensaje />  
  <TipoDocumento>  
    <NumCaso>112</NumCaso>  
    <Fecha>11/1/2004 5:00:52 PM</Fecha>  
    <Descripcion>Entrevista de Nuevos Clientes</Descripcion>  
    <ClaseDoc>Datos Basicos y Autorizaciones</ClaseDoc>  
    <CodCategoria>1</CodCategoria>  
    <CodTipDocumento>12</CodTipDocumento>  
  </TipoDocumento>  
  <TipoDocumento>  
    <NumCaso>143</NumCaso>  
    <Fecha>11/25/2004 4:00:27 PM</Fecha>  
    <Descripcion>Novado</Descripcion>  
    <ClaseDoc>Documentos de Garantias</ClaseDoc>  
    <CodCategoria>15</CodCategoria>  
    <CodTipDocumento>96</CodTipDocumento>  
  </TipoDocumento>  
</GetTipoDocumento>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de salida de los servicios Web's, los demás están definidos en el XML de salida del servicio **PutTipoDocumento_chk**.

UpdateMenuUsuario

Actualiza el menú para los accesos de un usuario

XML de entrada:

```
<UpdateMenuUsuario>
  <sistema>DocumentWarehouse</sistema>
  <usuario>admin</usuario>
  <UsuarioMenu>Jose_Araujo</UsuarioMenu>
  <password />
  <token>oqxgcc3pyiibetvrrnfnoq55</token>
  <impresion>N</impresion>
  <menu>1</menu>
  <menu>2</menu>
  <menu>7</menu>
  <menu>13</menu>
</UpdateMenuUsuario>
```

En donde los Tag's token, usuario, password y sistema son descritos en la sección de la estructura básica de entrada. El Tag <impresion> indica si el usuario tiene derecho de imprimir documentos y el <menu> identifica la opción del menú al cual tendrá acceso el usuario.

XML de salida:

```
<UpdateMenuUsuario xmlns="">
  <status>0</status>
  <mensaje></mensaje>
</UpdateMenuUsuario>
```

Los tag's <status> y <mensaje> fueron definidos en la estructura básica de la salida de los servicios Web's.