

|  |
| --- |
|  |
| Manual de Consumo Servicio Reporte de Eventos - INDICA |
|  |

|  |
| --- |
| * Elaborado por   Jorge Puerta |
| * Aprobado |
| * Fecha de elaboración   2019-05-28 |
| * Fecha última modificación   2019-05-29 |

|  |  |
| --- | --- |
| TABLA |  |
| DE CONTENIDO |  |

[1 Objetivo 3](#_Toc10039411)

[2 Rutas 3](#_Toc10039412)

[3 Autenticación y Autorización 3](#_Toc10039413)

[4 Uso del Servicio cliente DEMO Windows 4](#_Toc10039414)

[4.1 Reporte 5](#_Toc10039415)

[4.2 Consulta 7](#_Toc10039416)

[5 Consumo del servicio con .Net 8](#_Toc10039417)

[5.1 Reporte de Eventos 10](#_Toc10039418)

[5.2 Consulta de Eventos 10](#_Toc10039419)

[6 Consumo del servicio web con Java 12](#_Toc10039420)

# Objetivo

Ilustrar el procedimiento para adicionar la referencia al servicio web de reporte de Eventos y ejecutar las diferentes operaciones que expone el servicio.

# Rutas

**PRUEBAS**

http://serviciosxmcal.xm.com.co/ServicioEventos/SrvReporteEventos.svc

<http://serviciosxmcal.xm.com.co/ServicioEventos/SrvConsultaEventos.svc>

**PRODUCCIÓN**

<http://serviciosxm.xm.com.co/ServicioEventos/SrvReporteEventos.svc>

<http://serviciosxm.xm.com.co/ServicioEventos/SrvConsultaEventos.svc>

**DEMOS**

Los demos se encuentran en TFS XM en el repositorio del proyecto INDICA:

Demo de Java: /FUENTE/Demo/JAVA

Demo de .NET: /FUENTE/Demo//WINDOWS

# Autenticación y Autorización

**Autenticación:** Integrada Windows

**Autorización:** Grupo de usuario **INDICA**

Sólo los usuarios o grupos del dominio que sean miembros del grupo **INDICA** del Directorio Activo, pueden ejecutar las diferentes operaciones que ejecuta el servicio.

El usuario con el cual se hace uso del servicio debe estar registrado dentro del rol “Agente” del aplicativo INDICA.

# Uso del Servicio cliente DEMO Windows

Para validar el uso del servicio desde la red corporativa de cada agente se allega un cliente DEMO Windows para que realicen las respectivas validaciones, el cual funciona de la siguiente manera.

Primero configurar el archivo ClienteServicioReportes.exe.config (que se allega con el ejecutable del aplicativo) las rutas de los servicios que se van a consumir (preconfiguradas para el ambiente de pruebas)

<client>

<endpoint address="http://serviciosxmcal.xm.com.co/ServicioEventos/SrvReporteEventos.svc"

binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding\_IReporteEventosService"

contract="ReporteEventos.IReporteEventosService" name="BasicHttpBinding\_IReporteEventosService" />

<endpoint address="http://serviciosxmcal.xm.com.co/ServicioEventos/SrvConsultaEventos.svc"

binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding\_IConsultaEventosService"

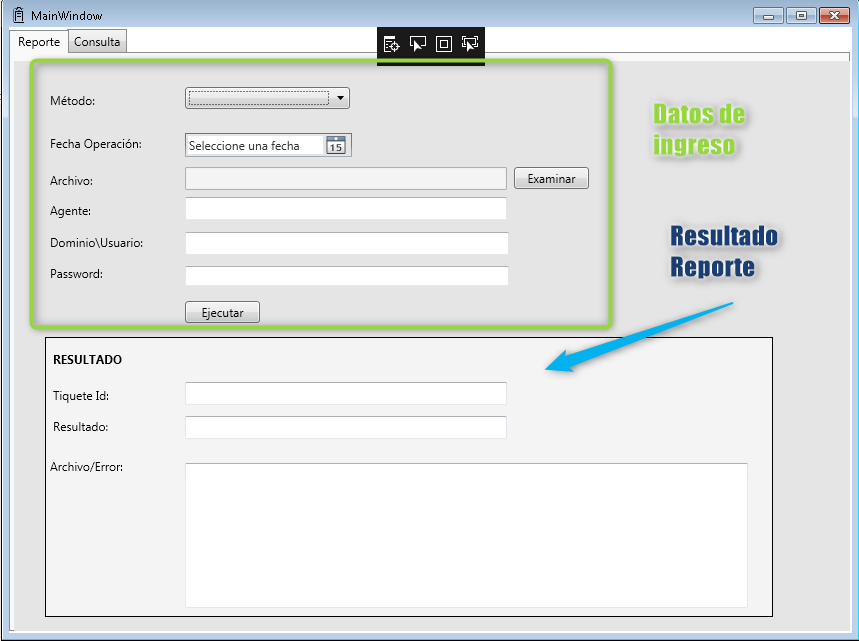
contract="ConsultaEventos.IConsultaEventosService" name="BasicHttpBinding\_IConsultaEventosService" />

</client>

Una vez realizado lo anterior ejecutar el archivo ClienteServicioReportes.exe

Se tienen dos pestañas reporte y consulta

## Reporte

****

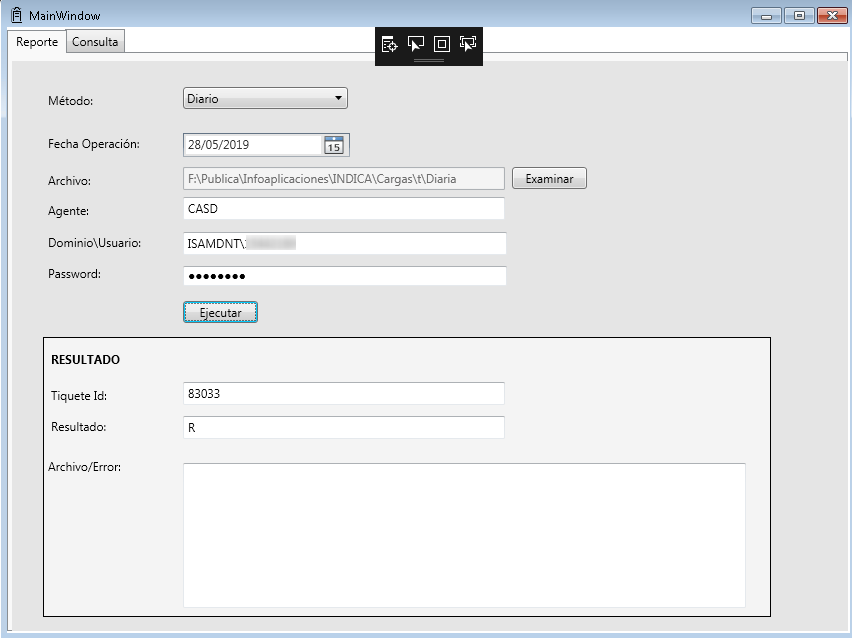
En esta pestaña se solicita la información descrita

* Método: Diario, mensual, Alto Impacto, SSPD (No para agentes)
* Fecha de operación: Fecha a la que corresponde la información del archivo adjunto
* Archivo: Ruta del archivo a cargar
* Agente: Código ASIC (4 letras) del agente al que corresponde el reporte
* Dominio\Usuario: Dominio y usuario con el cual se autentica dentro de la red de XM debe ser el mismo autorizado dentro del aplicativo INDICA
* Password: Password correspondiente al usuario anterior

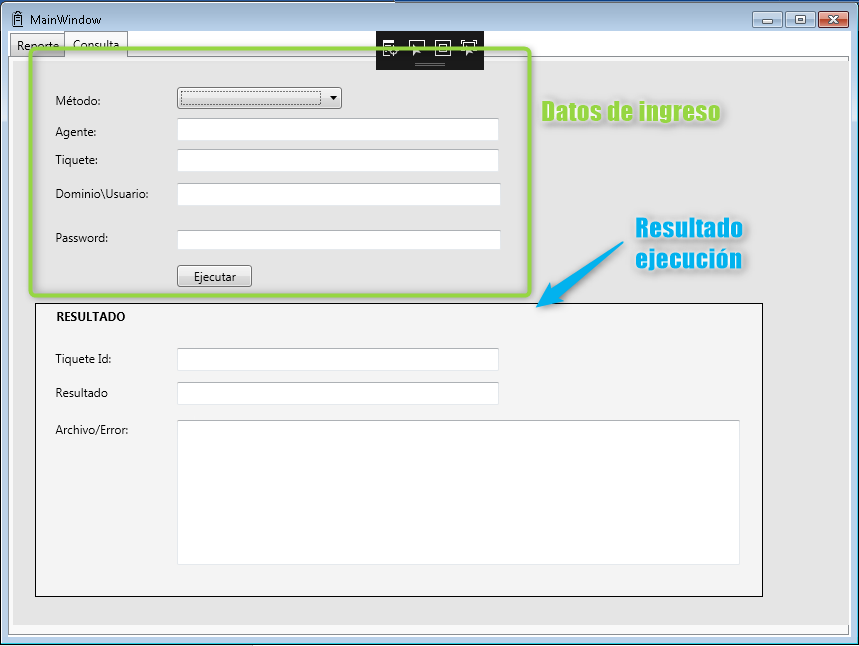
Como resultado de la ejecución de la carga se entrega:

* Tiquete Id: Tiquete – número con el cual se identificará el reporte de información
* Resultado: Aquí se informa el estado inicial del reporte puede tener los siguientes valores R, E y F
  + R: Recibido – El reporte fue recibido y actualmente se está procesando
  + E: Exitoso – El reporte fue procesado y validado
  + F: Fallido – El proceso de reporte fallo, por lo general viene acompañado de archivo de texto de errores.
* Archivo/Error: archivo de texto con la descripción de los 1000 primeros errores encontrados

Ejemplo ejecución:



## Consulta



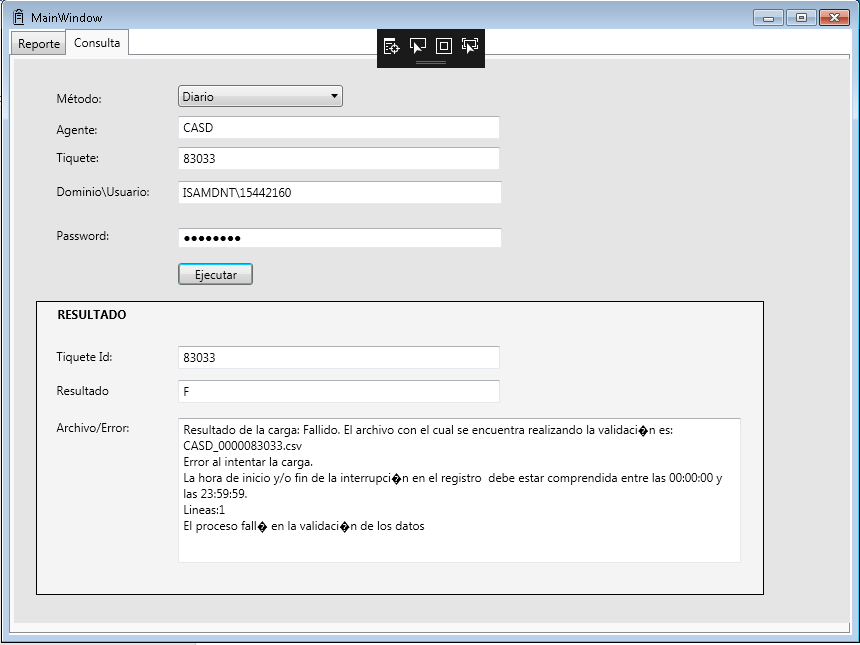
En esta pestaña se solicita la información descrita

* Método: Diario, mensual, Alto Impacto, SSPD (No para agentes)
* Agente: Código ASIC (4 letras) del agente al que corresponde el reporte
* Tiquete: número con el cual se identificó el reporte de información
* Dominio\Usuario: Dominio y usuario con el cual se autentica dentro de la red de XM debe ser el mismo autorizado dentro del aplicativo INDICA
* Password: Password correspondiente al usuario anterior

Como resultado de la ejecución de la carga se entrega:

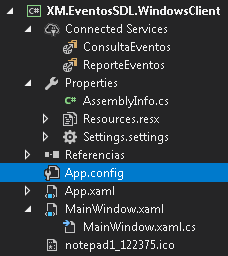
* Tiquete Id: Tiquete – número con el cual se identificará el reporte de información
* Resultado: Aquí se informa el estado inicial del reporte puede tener los siguientes valores R, E y F
  + R: Recibido – El reporte fue recibido y actualmente se está procesando
  + E: Exitoso – El reporte fue procesado y validado
  + F: Fallido – El proceso de reporte fallo, por lo general viene acompañado de archivo de texto de errores.
* Archivo/Error: archivo de texto con la descripción de los 1000 primeros errores encontrados

Ejemplo ejecución:



# Consumo del servicio con .Net

Para allegar como ejemplo se allega el proyecto XM.EventosSDL.WindowsClient para el consumo del servicio.



En este encontrará todo el consumo del servicio dentro de una aplicación simple de WPF construida con Visual Studio 2017.

Como configuración inicial se deberán de definir las rutas del servicio dentro de los Endpoints que se encuentran dentro del archivo de configuración App.config:

<client>

<endpoint address="http://serviciosxmcal.xm.com.co/ServicioEventos/SrvReporteEventos.svc"

binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding\_IReporteEventosService"

contract="ReporteEventos.IReporteEventosService" name="BasicHttpBinding\_IReporteEventosService" />

<endpoint address="http://serviciosxmcal.xm.com.co/ServicioEventos/SrvConsultaEventos.svc"

binding="basicHttpBinding" bindingConfiguration="BasicHttpBinding\_IConsultaEventosService"

contract="ConsultaEventos.IConsultaEventosService" name="BasicHttpBinding\_IConsultaEventosService" />

</client>

**Errores de Negocio**

Este servicio puede devolver de manera directa errores de negocio:

**EN001** - El mensaje de consulta de eventos no cumple con el formato y la estructura establecida

**EN002** - El reporte de los eventos no se encuentra dentro del plazo regulatorio

**EN003** - El operador de red no se encuentra activo para la fecha de operación

**EN004** - El operador de red no se encuentra habilitado para el reporte de eventos

**EN005** – Se presentaron inconsistencias

**EN006** - Autenticación fallida para el Operador de Red dentro del servicio

**EN007** - No se adjuntó el archivo

**EN008** - Identificador de reporte inválido

## Reporte de Eventos

Dentro de archivo de código fuente MainWindow.xaml.cs encontrará el método

private void btnEjecutar\_Click(object sender, RoutedEventArgs e)

Este método tiene dentro del mismo el ejemplo de consumo de cada uno de los métodos del servicio de reportes (Reporte Diario, Mensual, Alto Impacto y SSPD – Este último no aplica para agentes).

Es de anotar que cada uno de estos métodos recibe un objeto Request con la siguiente información:

* Agente: Agente que está realizando el proceso de carga
* Fecha Operación: fecha a la que corresponde la información que se está reportando (En caso de la carga mensual el primer día del mes Ejemplo:01/01/2019)
* PayLoad: donde se envia archivo .zip que contiene uno y solo un archivo de texto (se sugiere .csv) en el cual cada campo se encuentra separado por comas (,)

La respuesta de cada método es un objeto Response con la siguiente información:

* Identificador del reporte: Tiquete – número con el cual se identificará el reporte de información
* Estado Reporte: Aquí se informa el estado inicial del reporte puede tener los siguientes valores R, E y F
  + R: Recibido – El reporte fue recibido y actualmente se está procesando
  + E: Exitoso – El reporte fue procesado y validado
  + F: Fallido – El proceso de reporte fallo, por lo general viene acompañado de archivo de texto de errores.
* Payload: archivo de texto con la descripción de los 1000 primeros errores encontrados.

## Consulta de Eventos

Dentro de archivo de código fuente MainWindow.xaml.cs encontrará el método

private void btnEjecutarConsulta\_Click(object sender, RoutedEventArgs e)

Este método tiene dentro del mismo el ejemplo de consumo de cada uno de los métodos del servicio de consulta de reportes (Diario, Mensual, Alto Impacto y SSPD – Este último no aplica para agentes).

Es de anotar que cada uno de estos métodos recibe un objeto Request con la siguiente información:

* Agente: Agente que está realizando el proceso de carga
* Tiquete: Este es el número entregado en el momento del reporte de la información

La respuesta de cada método es un objeto Response con la siguiente información:

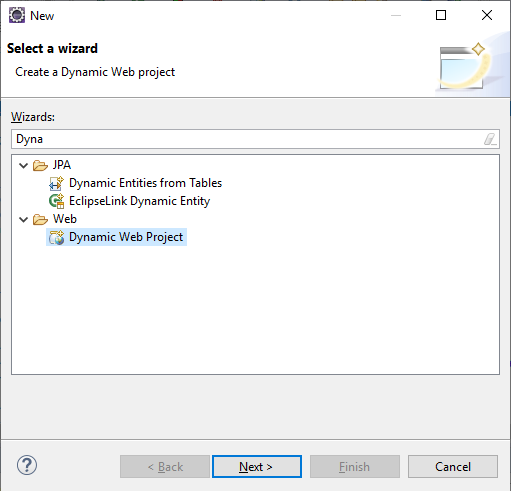
* Identificador del reporte: Tiquete – número con el cual se identificará el reporte de información
* Estado Reporte: Aquí se informa el estado inicial del reporte puede tener los siguientes valores R, E y F
  + R: Recibido – El reporte fue recibido y actualmente se está procesando
  + E: Exitoso – El reporte fue procesado y validado
  + F: Fallido – El proceso de reporte fallo, por lo general viene acompañado de archivo de texto de errores.
* Payload: archivo de texto con la descripción de los 1000 primeros errores encontrados.

# Consumo del servicio web con Java

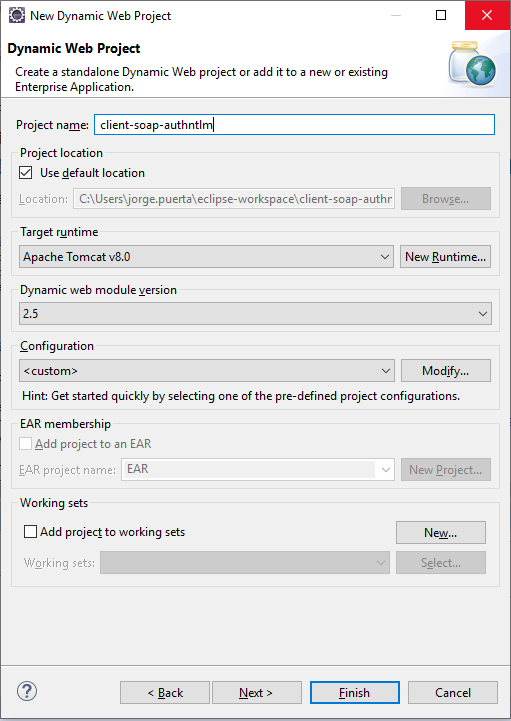
* Abrir el entorno de desarrollo para java, para el caso del ejemplo usaremos eclipse.



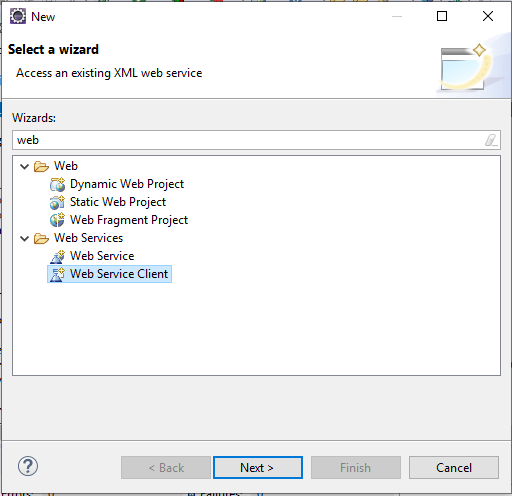
* Cree un proyecto en el cual debe adicionar la referencia al servicio Web. Para el caso del ejemplo crearemos un nuevo proyecto **Dynamic Web Project**, dado que la autenticación para el servicio se hace a través de NTLM



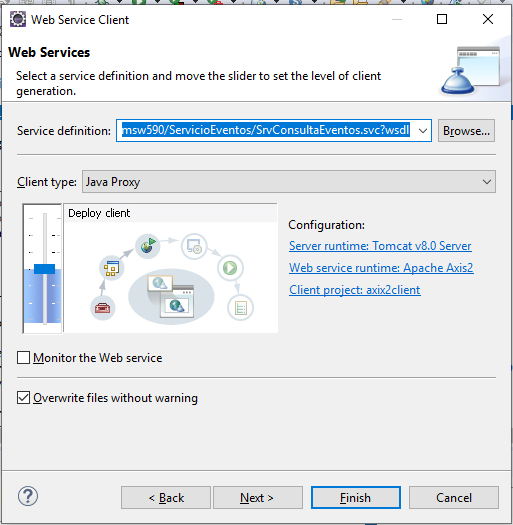
* Seleccionamos el tipo de proyecto y diligenciamos los campos obligatorios.



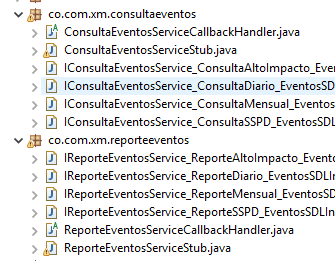
* Una vez hayamos creado el proyecto, hacemos clic derecho sobre el proyecto y seleccionamos la opción New 🡪 Other 🡪 Web Service Client .



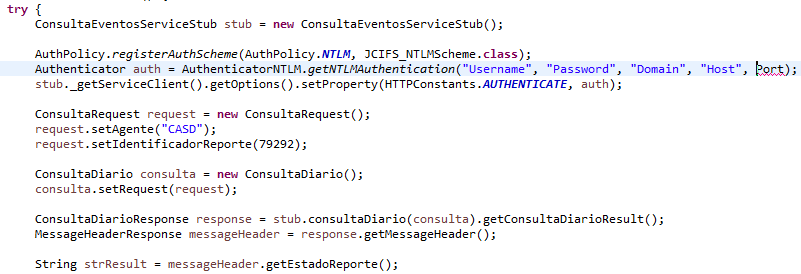
* En el campo **Service definition**, ingresamos la url descrita en [Rutas](#_Rutas), se configura el web service runtime con axis2 y hacemos clic en el botón **finish**.



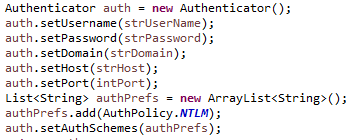
* El proceso anterior genera automáticamente el cliente para cada servicio configurado



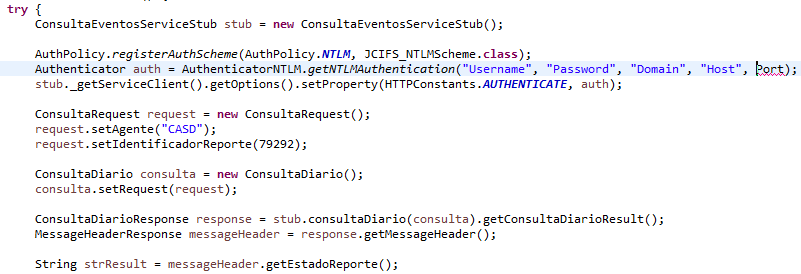
* Para consumir el servicio se utilizan las clases **Stub** generadas



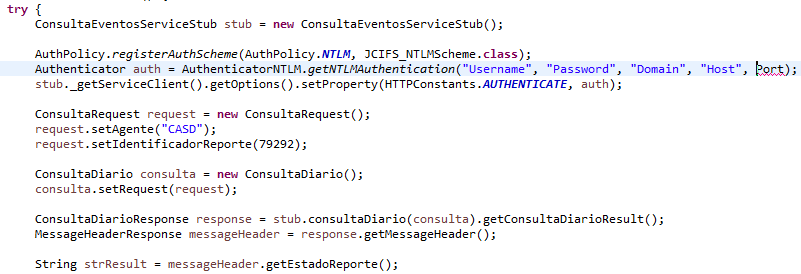
* La autenticación al servicio es mediante NTLM



* Se adicionan las políticas y configuración para autenticarse en el servicio



* Se instancia el **request** y se obtiene el **response** con el resultado del método invocado del servicio



* Finalmente, el **result** nos trae un objeto de tipo **MessageHeader** el cual contienen el estado del reporte y el identificador del reporte.

