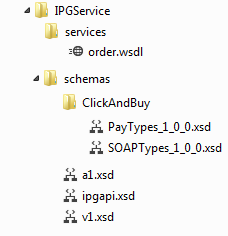
**.NET API Integration**

1. Install client and server certificates (The process is described in IPG\_IntegrationGuide\_API section 16.2 ASP)
2. Add Web Reference order.wsdl from <https://test.ipg-online.com/ipgapi/services/order.wsdl>

If you are unable to add the Web Reference from the *URL* you will need to have all schema files on your PC:

* order.wsd
* ipgapi.xsd
* a1.xsd
* v1.xsd
* PayTypes\_1\_0\_0.xsd
* SOAPTypes\_1\_0\_0.xsd

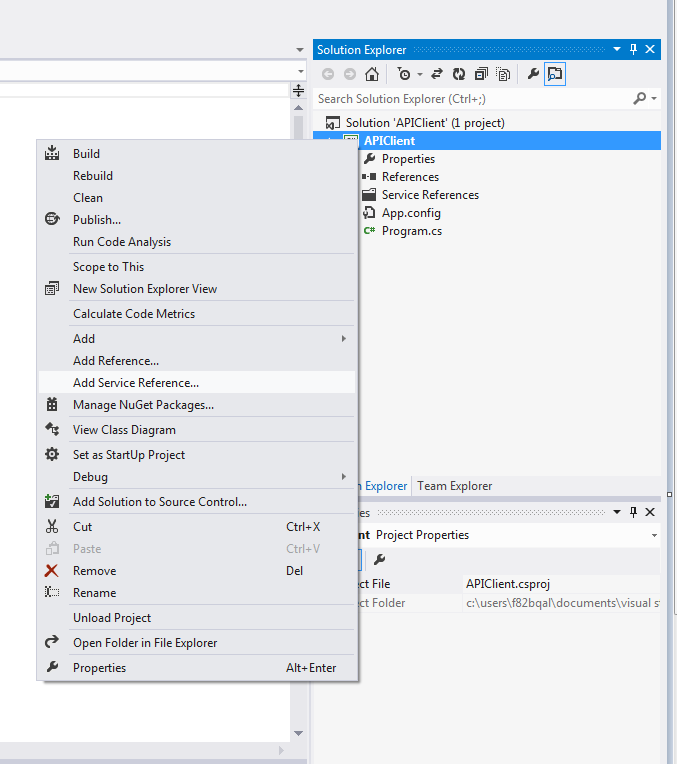
Schemas hierarchy:



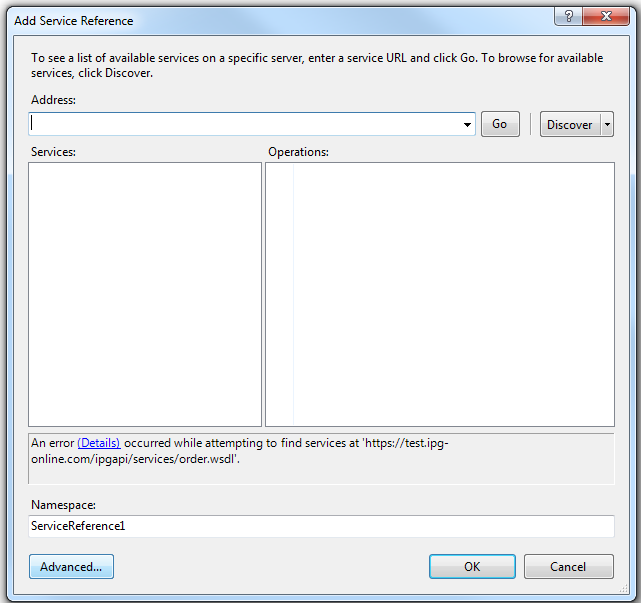
And use the path to the order.wsdl file you downloaded instead.

**Adding the Web Reference**

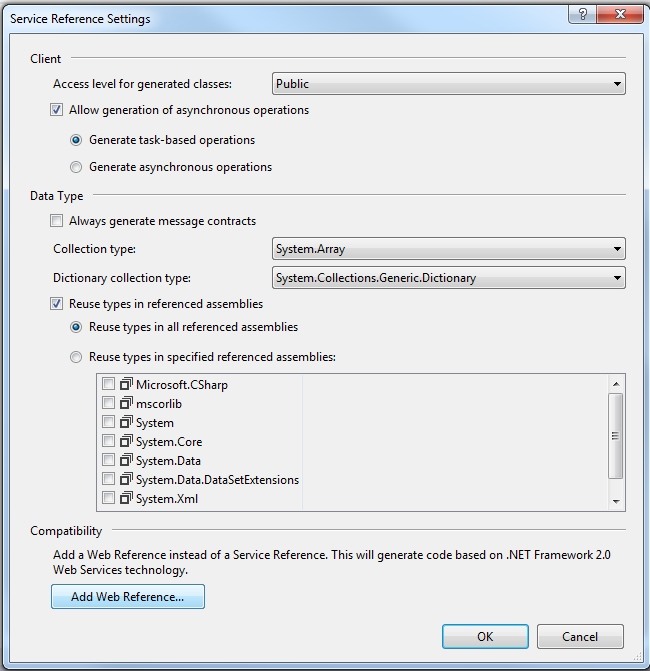
1. Right Click on your project and click on *Add Service Reference*



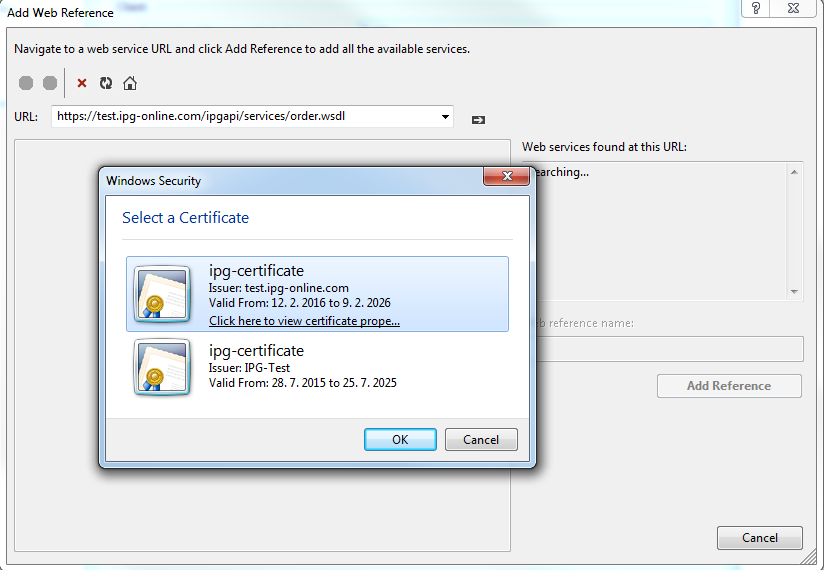
1. Click on *Advanced…* button



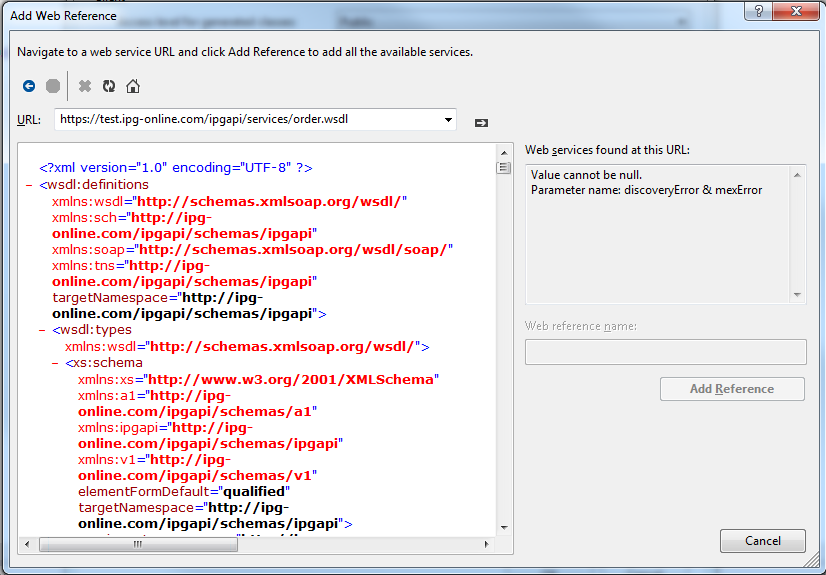
1. Click on *Add Web Reference…*

\

1. Give <https://test.ipg-online.com/ipgapi/services/order.wsdl> as the *URL* and press enter. You will be asked to select a certificate (the one you have installed). Select it and press *OK*.

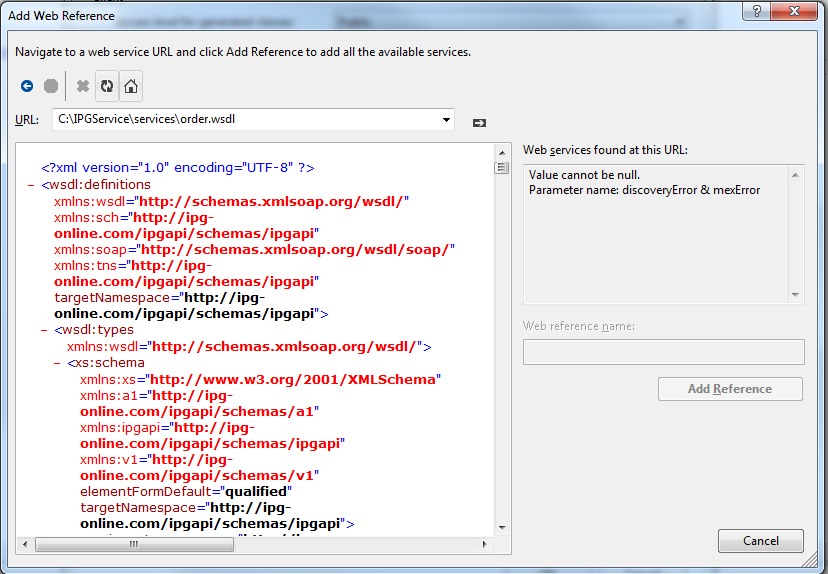


1. If you are unable to add it as Web Reference with the *URL* <https://test.ipg-online.com/ipgapi/services/order.wsdl>, you will have to add it from your PC.

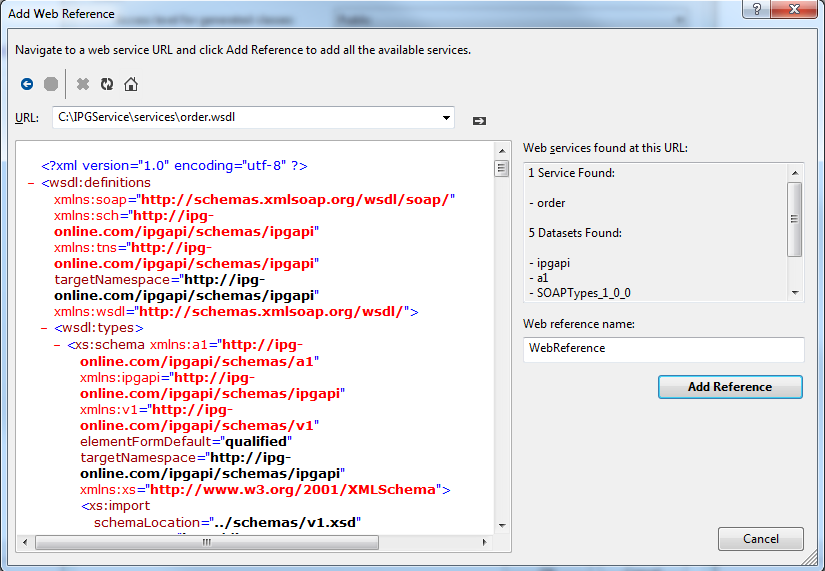


1. (In case you weren’t able to add the Web Reference with the *URL*)

Give as *URL* the path to the order.wsdl that you have on your PC. If you are adding the Web Reference this way you need to have all the schemas present on your computer.



1. The Service was found successfully. Name your Web Reference and click on *Add Reference*



The application will add the schema files into the project.

**Using the WSIPGAPIClient**

WSAPIAPIClient is a console Application that is a sample integration of the IPG API.

A sample code for creating Action request (InquiryOrder)

static void Main(string[] args)

{

// Disable Expect100Continue, when set to true (default value) error occures

// The request was aborted: Could not create SSL/TLS secure channel.

ServicePointManager.Expect100Continue = false;

//load certificate (the installed one)

X509Certificate2 certificate = null;

try

{

certificate = new X509Certificate2(@"c:\certificates\WS120995000.\_.4.p12", "tester02");

}

catch (Exception e)

{

Console.WriteLine("Problem loading certificate");

Console.WriteLine("Exception: " + e.Message);

}

// Create Request: IPGApiOrderService

IPGApiOrderService Request = new IPGApiOrderService();

String RequestResponse = "";

if (certificate != null)

{

// Add certificate to the Request

Request.ClientCertificates.Add(certificate);

//Specify the service URL

Request.Url = @"https://test.ipg-online.com:443/ipgapi/services";

//Set Network Credentials (certificate user and password) to the Request

Request.Credentials = new NetworkCredential("WS120995000.\_.4", "Tester02");

// If needed set Proxy settings to the Request

IWebProxy webProxy = new WebProxy("fdskbc1vip.sk.transacty.com", 8080);

webProxy.Credentials = new NetworkCredential("username", "password");

Request.Proxy = webProxy;

// create Request: Action: InquiryOrder

InquiryOrder oInquiryOrder = new InquiryOrder();

oInquiryOrder.StoreId = "120995000";

oInquiryOrder.OrderId = "C-0cc41568-aef6-45bb-8b26-5d1d5733191d";

ClientLocale oClientLocale = new ClientLocale();

oClientLocale.Country = "UK";

oClientLocale.Language = "en";

APIClient.WebReference.Action oAction = new APIClient.WebReference.Action();

oAction.Item = oInquiryOrder;

oAction.ClientLocale = oClientLocale;

IPGApiActionRequest ActionRequest = new IPGApiActionRequest();

ActionRequest.Item = oAction;

try

{

IPGApiActionResponse oResponse = Request.IPGApiAction(ActionRequest);

RequestResponse += "Succesfully:" + oResponse.successfully;

}

catch (Exception e)

{

RequestResponse = e.Message;

}finally{

Console.WriteLine(RequestResponse);

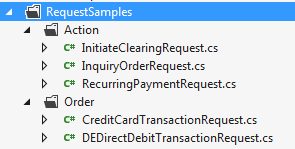
Console.ReadLine();

}

}

}

In the WSIPGClient the request samples are located in RequestSamples folder.



E.g. For Action Request InquiryOrder there is a corresponding class created InquiryOrdrRequest that creates the request.

InquiryOrderRequest Class:

class InquiryOrderRequest

{

public IPGApiActionRequest InquiryOrderActionRequest{ get; set; }

public InquiryOrderRequest()

{

InquiryOrder oInquiryOrder = new InquiryOrder();

oInquiryOrder.StoreId = "120995000";

oInquiryOrder.OrderId = "C-0cc41568-aef6-45bb-8b26-5d1d5733191d";

ClientLocale oClientLocale = new ClientLocale();

oClientLocale.Country = "UK";

oClientLocale.Language = "en";

WSIPGClient.WebReference.Action oAction;

oAction = new WSIPGClient.WebReference.Action();

oAction.Item = oInquiryOrder;

oAction.ClientLocale = oClientLocale;

InquiryOrderActionRequest = new IPGApiActionRequest();

InquiryOrderActionRequest.Item = oAction;

}

}

**Request / Response Logging**

In this sample the class *LoggerSoapExtension.cs* is used to log the outgoing request / incoming response to a file. All that is needed to do is to add the following configuration to the *app.config*and define the location where the files will be saved to (LOG\_DIRECTORY)

<system.web>

<webServices>

<soapExtensionTypes>

<add type="WSIPGClient.LoggerSoapExtension, WSIPGClient" priority="1" />

</soapExtensionTypes>

</webServices>

</system.web>

<appSettings>

<add key="LOG\_DIRECTORY" value="C:\log\" />

</appSettings>