



Assure Pay
we manage your payments

Implementatie AssurePay ASP .NET

iDEAL



Inhoudsopgave

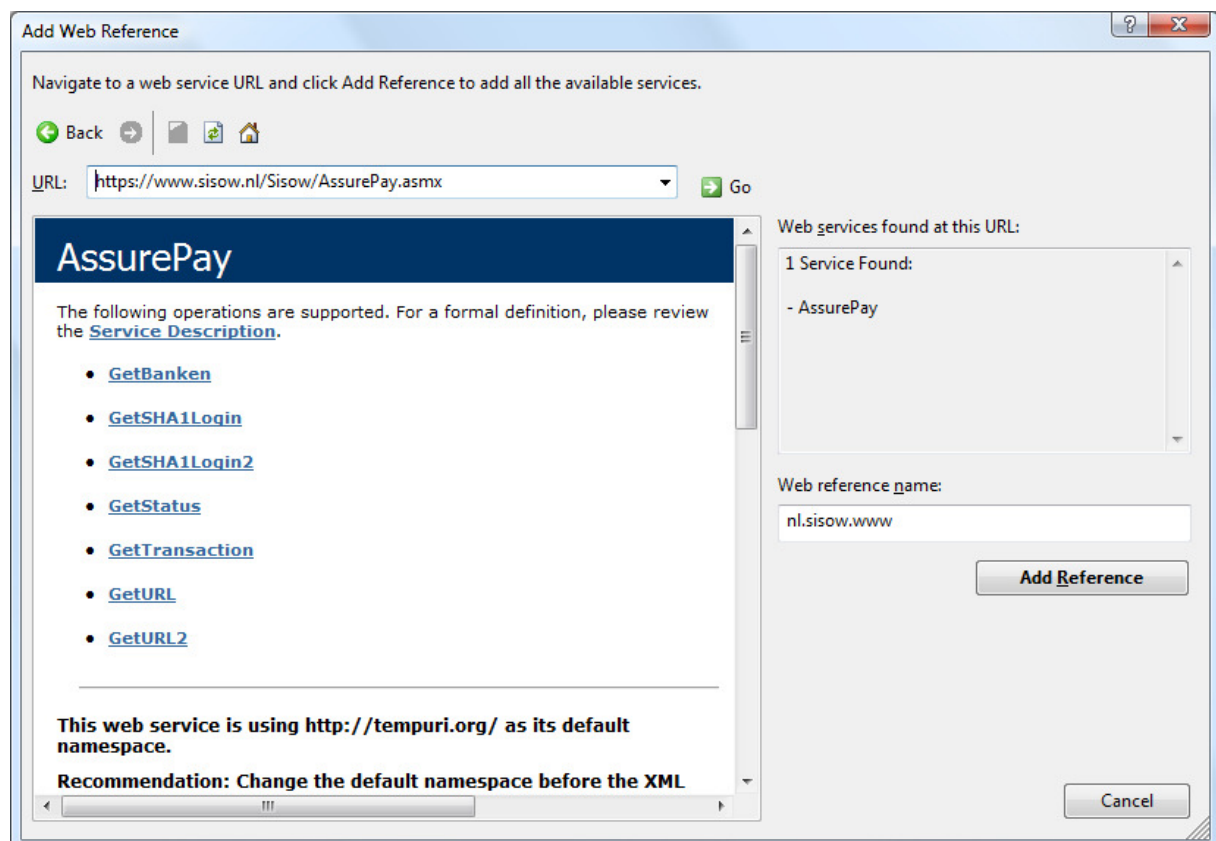
Microsoft ASP.NET.....	3
Classic ASP	9

Microsoft ASP.NET

Implementeren met Microsoft ASP.NET is zeer eenvoudig.

WebServices kunnen eenvoudig worden geconsumeerd.

Hieronder zie je hoe eenvoudig dit is in Microsoft's Visual Web Developer 2008 Express Edition:

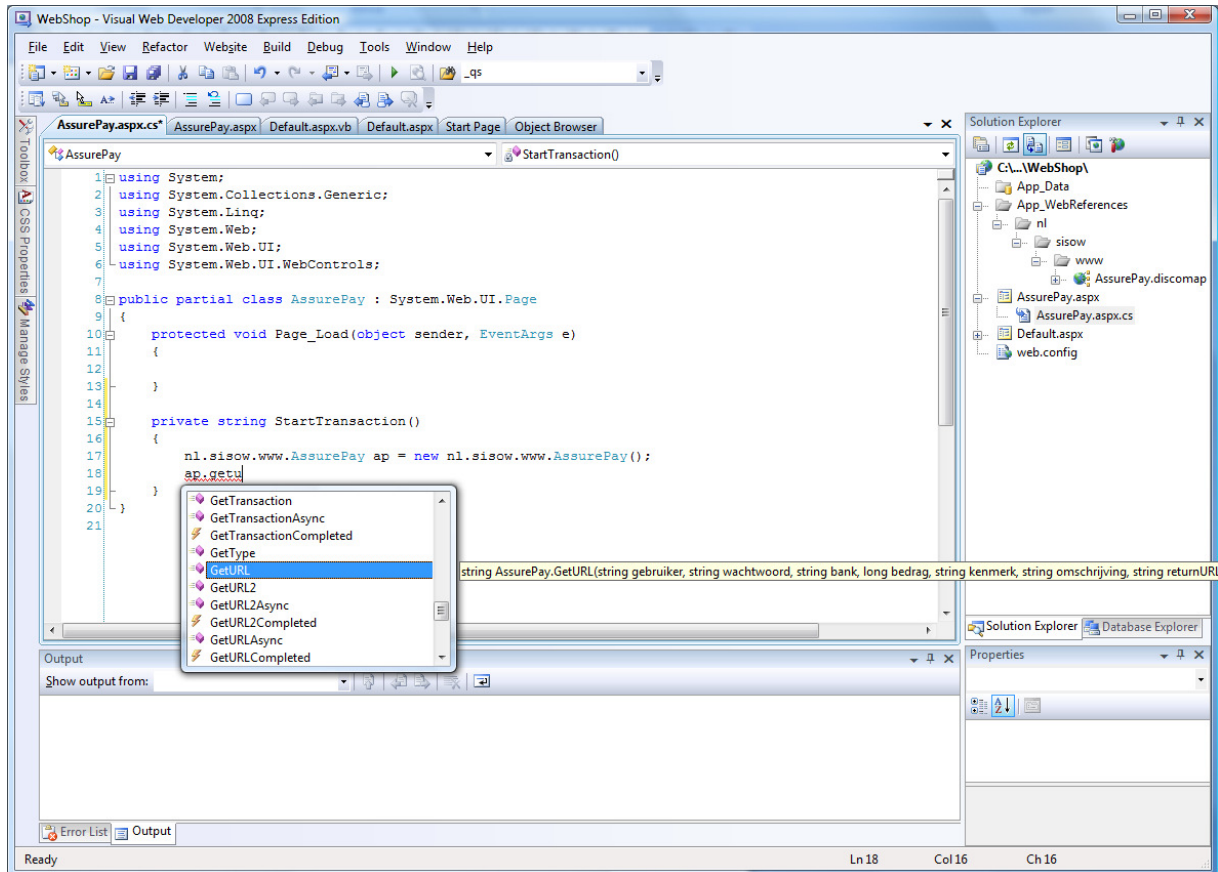


Vul "https://www.sisow.nl/Sisow/AssurePay.asmx" in in het URL veld en bovenstaand plaatje verschijnt.

Er wordt 1 Webservice gevonden met de volgende functies:

- GetBanken: hiermee kan de actuele lijst van aangesloten banken worden verkregen;
- GetSHA1Login en GetSHA1Login2: deze zullen in de nabije toekomst komen te vervallen. Oorspronkelijk bedoeld om een SHA1 waarde te verkrijgen van onder andere de componenten gebruikersnaam, wachtwoord, transactiebedrag en betalingskenmerk;
- GetStatus: hiermee kan de status van een iDEAL transactie worden bepaald;
- GetTransaction: idem als GetStatus, maar tevens met de mogelijkheid om alle informatie, verkregen uit de iDEAL transactie, op te vragen;
- GetURL: hiermee wordt de URL bepaald waarmee een iDEAL transactie kan worden gestart;
- GetURL2: idem, echter geeft deze functie ook de iDEAL Transaction ID terug;

Nadat de referentie naar de Webservice van Assure Pay is toegevoegd kan men uitermate simplistisch binnen de IDE (Integrated Development Environment) hiervan gebruik maken.
Zie onderstaand voorbeeld:



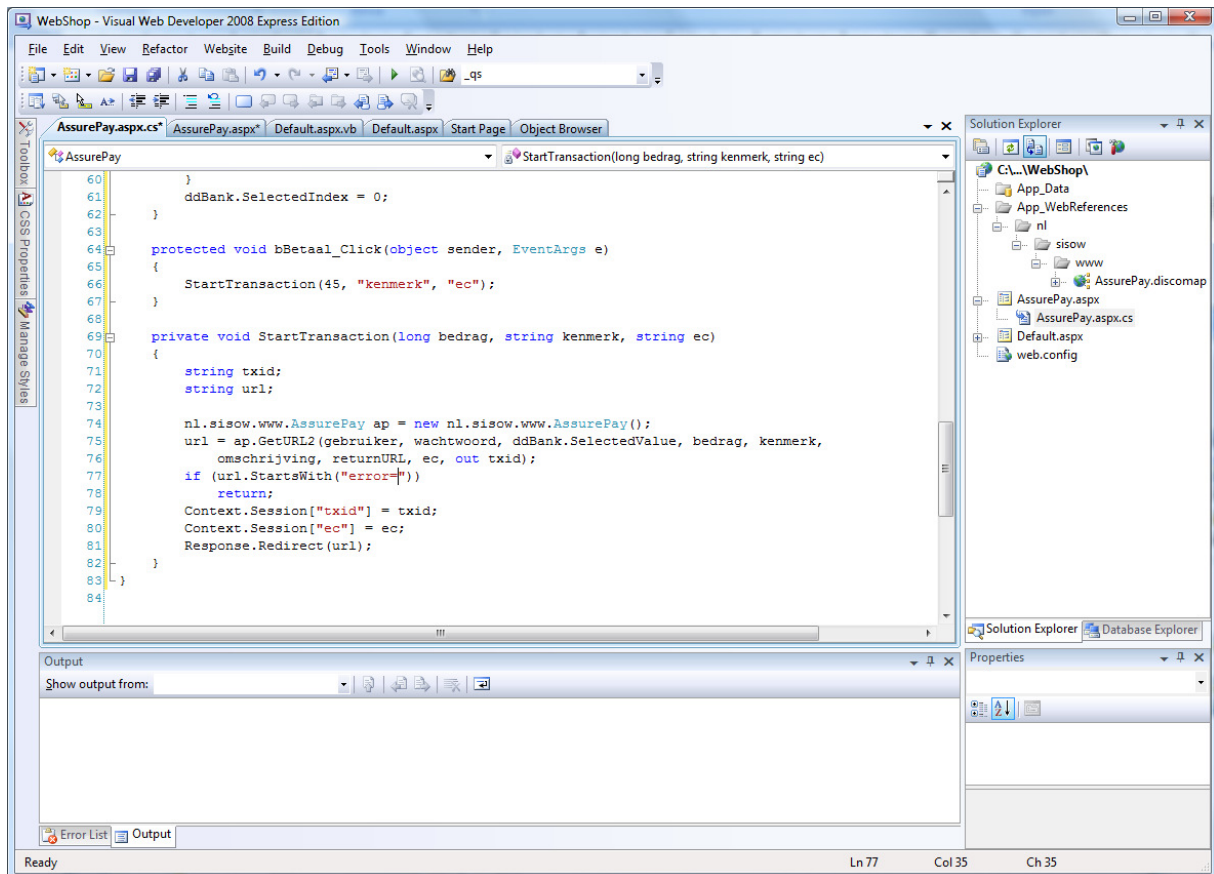
Er wordt een instantie van het AssurePay klasse geïnitieerd, waarna de functies beschikbaar komen alsof het een lokale klasse betreft.

In onderstaand voorbeeld zie je in de method `StartTransaction` hoe een iDEAL transactie gestart kan worden.

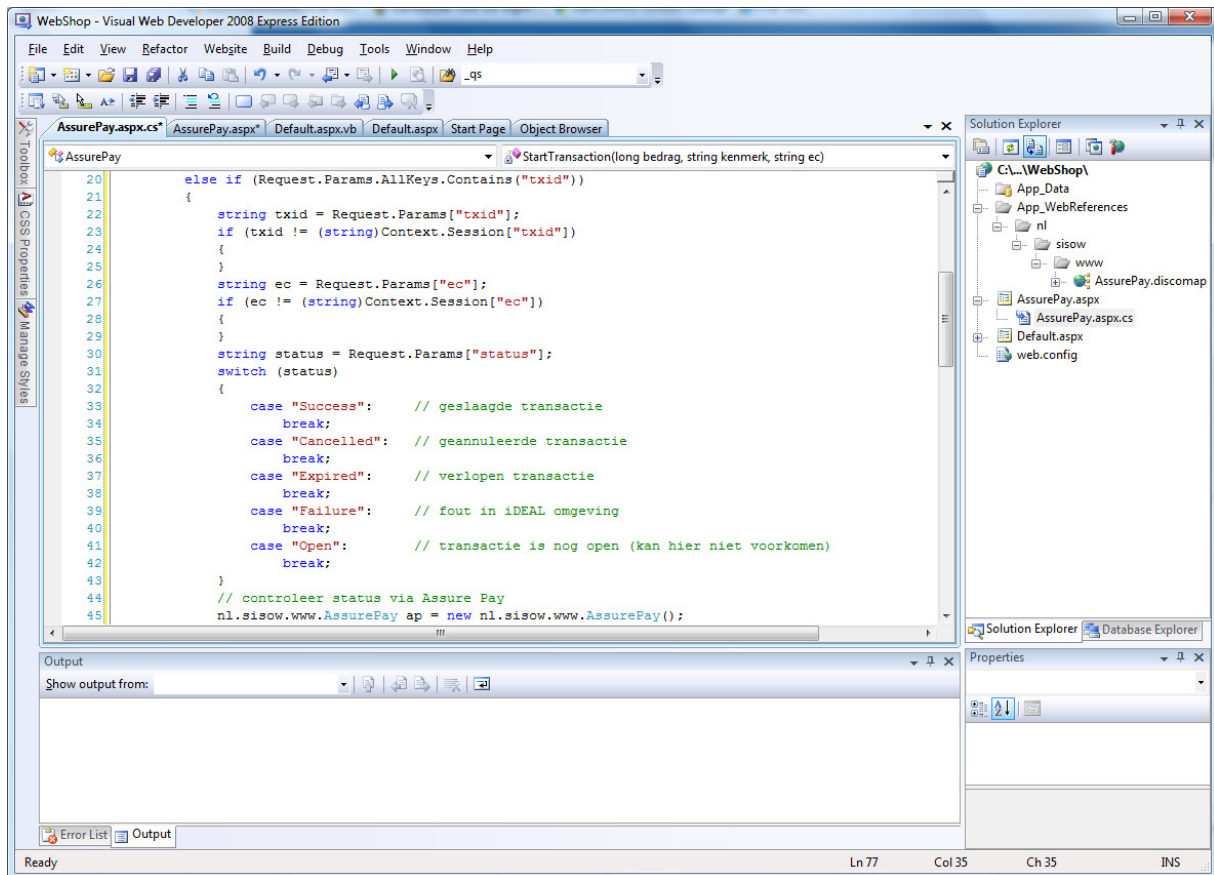
Aanroep van `GetURL2` retourneert of een fout of een URL.

Een fout is te herkennen, doordat de geretourneerde waarde dan begint met "error=".

Door middel van de aanroep "`Response.Redirect(url)`" wordt de iDEAL transactie gestart.

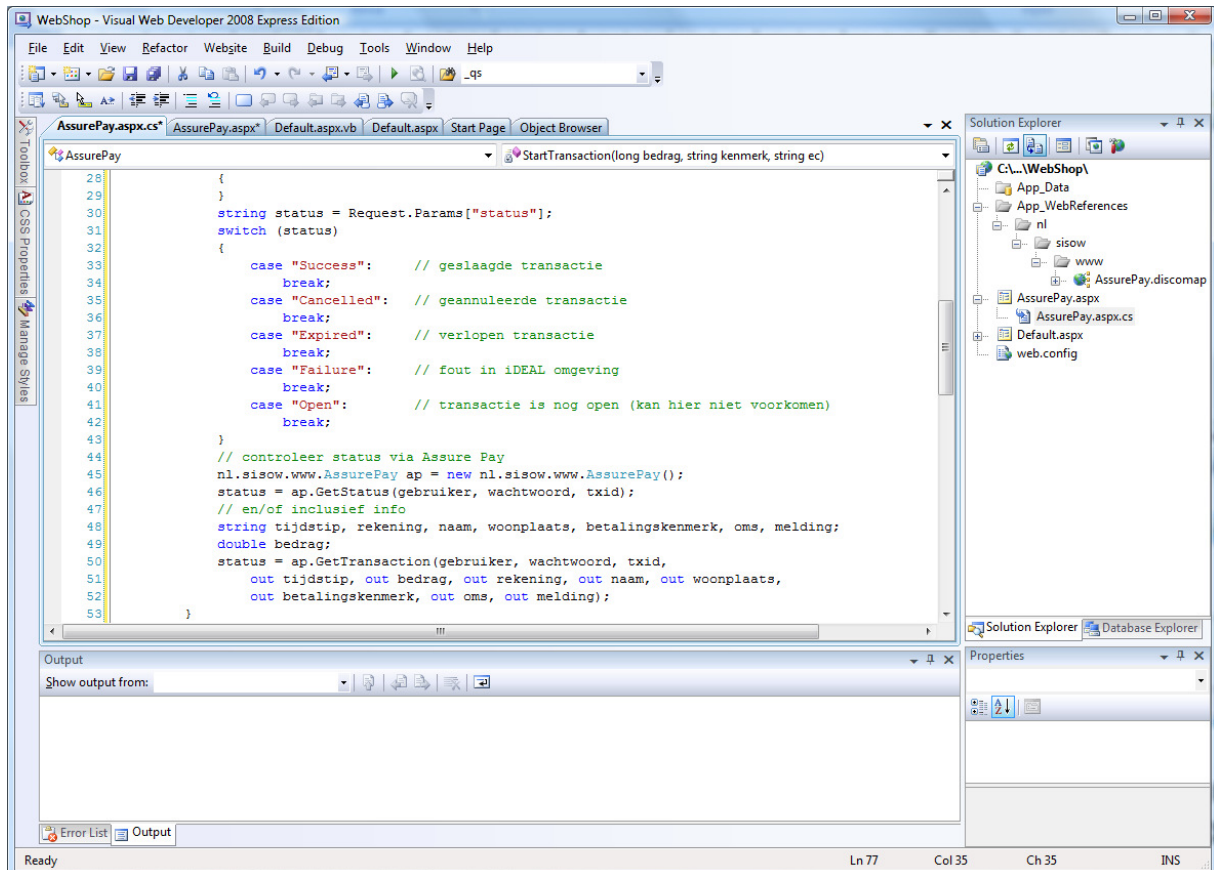


De terugkoppeling van het resultaat van de iDEAL transactie zou er als volgt uit kunnen zien:



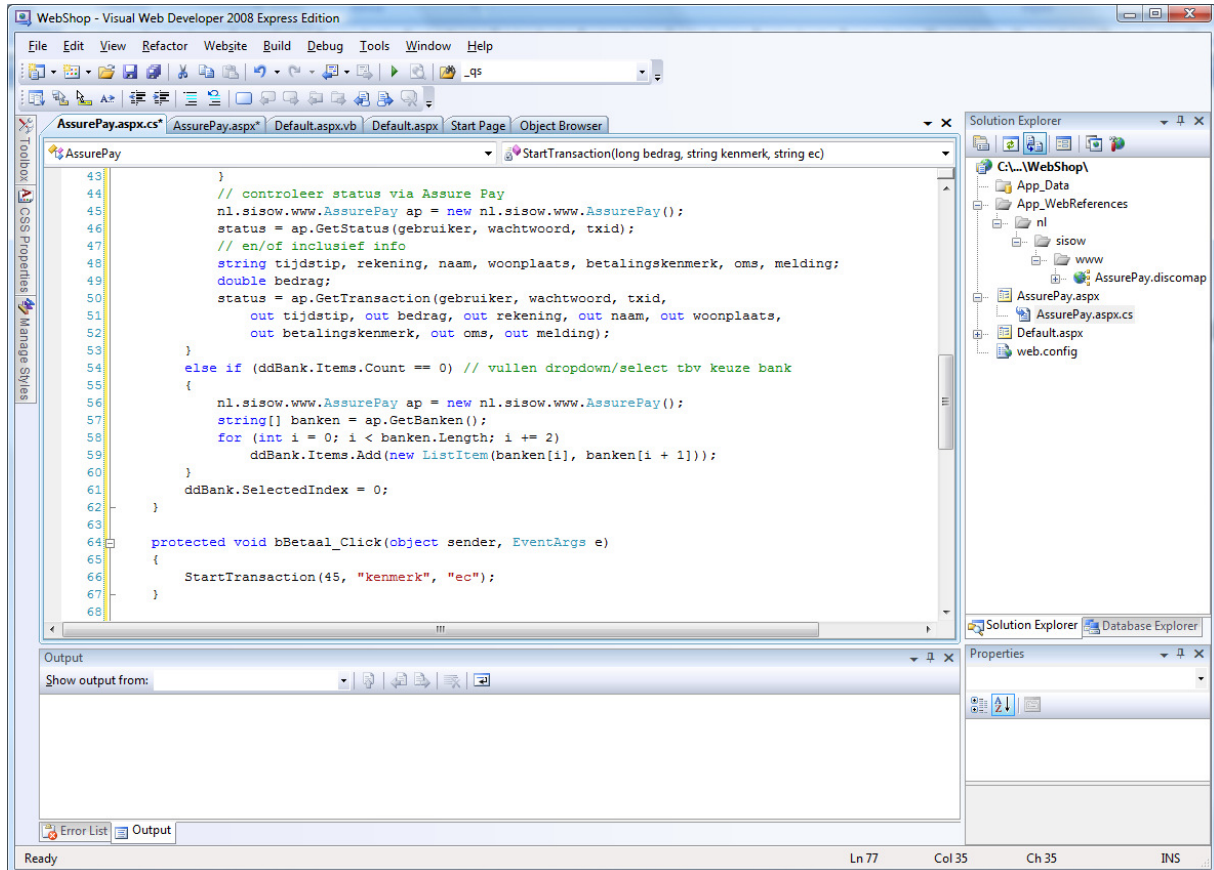
Controleer of de QueryString parameters "txid", "ec" en "status" aanwezig zijn. Eventueel de aanvullende controle met behulp van de aanverwant opgeslagen sessie variabelen "txid" en "ec".

Verdere controle, voor de volledige zekerheid (de status variabele kan gemanipuleerd zijn), gebeurt middels GetStatus en/of GetTransaction. Zie onderstaand voorbeeld:



```
28 {
29 }
30 string status = Request.Params["status"];
31 switch (status)
32 {
33     case "Success": // geslaagde transactie
34         break;
35     case "Cancelled": // geannuleerde transactie
36         break;
37     case "Expired": // verlopen transactie
38         break;
39     case "Failure": // fout in iDEAL omgeving
40         break;
41     case "Open": // transactie is nog open (kan hier niet voorkomen)
42         break;
43 }
44 // controleer status via Assure Pay
45 nl.sisow.www.AssurePay ap = new nl.sisow.www.AssurePay();
46 status = ap.GetStatus(gebruiker, wachtwoord, txid);
47 // en/of inclusief info
48 string tijdstip, rekening, naam, woonplaats, betalingskenmerk, oms, melding;
49 double bedrag;
50 status = ap.GetTransaction(gebruiker, wachtwoord, txid,
51     out tijdstip, out bedrag, out rekening, out naam, out woonplaats,
52     out betalingskenmerk, out oms, out melding);
53 }
```

Als laatste kan men hieronder zien hoe een dropdown (ddBank), voor de keuze van de gewenste bank, gevuld kan worden met de informatie verkregen uit de GetBanken functie:



Zo eenvoudig kan een iDEAL betaling met behulp van de WebService functies van Assure Pay worden geïmplementeerd in bijvoorbeeld de gratis IDE van Microsoft Visual Web Developer 2008 Express Edition.

Classic ASP

Voor Classic ASP is volgende (mogelijke) wrapper te gebruiken:

```
using System;

public class AssurePay
{
    private string user;
    private string password;

    MSXML2.XMLHTTP30Class xmlhttp;
    MSXML2.DOMDocument30Class domdoc;

    public AssurePay(string gebruiker, string wachtwoord)
    {
        user = gebruiker;
        password = wachtwoord;
    }

    private bool send(string method)
    {
        string msg;

        // opbouwen SOAP bericht
        msg = "<?xml version=\"1.0\" encoding=\"utf-8\"?>";
        msg += "<soap:Envelope xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\"
xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\"
xmlns:soap=\"http://schemas.xmlsoap.org/soap/envelope/\">";
        msg += "<soap:Body>";
        msg += method;
        msg += "</soap:Body>";
        msg += "</soap:Envelope>";

        // verstuur SOAP bericht
        try
        {
            xmlhttp = new MSXML2.XMLHTTP30Class();
            xmlhttp.open("POST", "https://www.sisow.nl/Sisow/AssurePay.asmx", false, null,
null);
            xmlhttp.setRequestHeader("Content-Type", "text/xml; charset=utf-8");
            xmlhttp.setRequestHeader("Content-Length", msg.Length.ToString());
            xmlhttp.send(msg);
        }
        catch (Exception)
        {
            return false;
        }

        // lees response
        try
        {
            domdoc = new MSXML2.DOMDocument30Class();
            domdoc.load(xmlhttp.responseXML);
        }
        catch (Exception)
        {
            return false;
        }

        return true;
    }

    public string[] GetBanken()
    {
        string[] banken = null;

        if (!send("<GetBanken xmlns=\"http://tempuri.org/\"/>"))
            return null;
    }
}
```

```

        try
        {
            MSXML2.IXMLDOMNode node =
domdoc.documentElement.selectNodes("//GetBankenResult")[0];
            int len = node.childNodes.length;
            banken = new string[len];
            for (int i = 0; i < len; i++)
                banken[i] = node.childNodes[i].text;
        }
        catch (Exception)
        {
            return null;
        }

        return banken;
    }

    public string GetStatus(string transaction)
    {
        string msg = "<GetStatus xmlns=\"http://tempuri.org/\">";
        msg += "<gebruiker>" + user + "</gebruiker>";
        msg += "<wachtwoord>" + password + "</wachtwoord>";
        msg += "<transaction>" + transaction + "</transaction>";
        msg += "</GetStatus>";

        if (!send(msg))
            return null;

        try
        {
            return domdoc.documentElement.selectNodes("//GetStatusResult")[0].text;
        }
        catch (Exception)
        {
            return null;
        }
    }

    public string GetTransaction(string transaction,
        out string stamp, out string amount, out string account, out string name, out string
city, out string purchaseid, out string description, out string message)
    {
        stamp = amount = account = name = city = purchaseid = description = message = "";

        string msg = "<GetTransaction xmlns=\"http://tempuri.org/\">";
        msg += "<gebruiker>" + user + "</gebruiker>";
        msg += "<wachtwoord>" + password + "</wachtwoord>";
        msg += "<transaction>" + transaction + "</transaction>";
        msg += "</GetTransaction>";

        if (!send(msg))
            return null;

        try
        {
            domdoc = new MSXML2.DOMDocument30Class();
            domdoc.load(xmlhttp.responseXML);
            stamp = domdoc.documentElement.selectNodes("//stamp")[0].text;
            amount = domdoc.documentElement.selectNodes("//amount")[0].text;
            account = domdoc.documentElement.selectNodes("//account")[0].text;
            name = domdoc.documentElement.selectNodes("//name")[0].text;
            city = domdoc.documentElement.selectNodes("//city")[0].text;
            purchaseid = domdoc.documentElement.selectNodes("//purchaseid")[0].text;
            description = domdoc.documentElement.selectNodes("//description")[0].text;
            message = domdoc.documentElement.selectNodes("//message")[0].text;
            return domdoc.documentElement.selectNodes("//GetTransactionResult")[0].text;
        }
        catch (Exception)
        {
            return null;
        }
    }

    public string GetURL(string bank, long bedrag, string kenmerk, string omschrijving, string
returnURL, string entrance)
    {
        string msg = "<GetURL xmlns=\"http://tempuri.org/\">";

```

```

msg += "<gebruiker>" + user + "</gebruiker>";
msg += "<wachtwoord>" + password + "</wachtwoord>";
msg += "<bank>" + bank + "</bank>";
msg += "<bedrag>" + bedrag + "</bedrag>";
msg += "<kenmerk>" + kenmerk + "</kenmerk>";
msg += "<omschrijving>" + omschrijving + "</omschrijving>";
msg += "<returnURL>" + returnUrl + "</returnURL>";
msg += "<entrance>" + entrance + "</entrance>";
msg += "</GetURL>";

if (!send(msg))
    return null;

try
{
    return domdoc.documentElement.selectNodes("//GetURLResult")[0].text;
}
catch (Exception)
{
    return null;
}
}

public string GetURL2(string bank, long bedrag, string kenmerk, string omschrijving,
string returnUrl, string entrance, out string txid)
{
    txid = "";

    string msg = "<GetURL2 xmlns=\"http://tempuri.org/\">";
    msg += "<gebruiker>" + user + "</gebruiker>";
    msg += "<wachtwoord>" + password + "</wachtwoord>";
    msg += "<bank>" + bank + "</bank>";
    msg += "<bedrag>" + bedrag + "</bedrag>";
    msg += "<kenmerk>" + kenmerk + "</kenmerk>";
    msg += "<omschrijving>" + omschrijving + "</omschrijving>";
    msg += "<returnURL>" + returnUrl + "</returnURL>";
    msg += "<entrance>" + entrance + "</entrance>";
    msg += "</GetURL2>";

    if (!send(msg))
        return null;

    try
    {
        txid = domdoc.documentElement.selectNodes("//txid")[0].text;
        return domdoc.documentElement.selectNodes("//GetURL2Result")[0].text;
    }
    catch (Exception)
    {
        return null;
    }
}
}

```

Een andere mogelijkheid, voor Classic ASP, om met de gateway van Assure Pay te koppelen is de REST methode.