



22.5.2023

MGS XRechnung.NET DLL

XRechnung ganz einfach erstellt



Michael Grube
MG SOFTWARE ENTWICKLUNG

Vorwort

Mit dieser dynamischen Link Library (DLL) Erstellen Sie in wenigen Augenblicken Ihre eigene XRechnung.

Was dazu notwendig ist, lesen Sie in diesen kleinen Abschnitten.

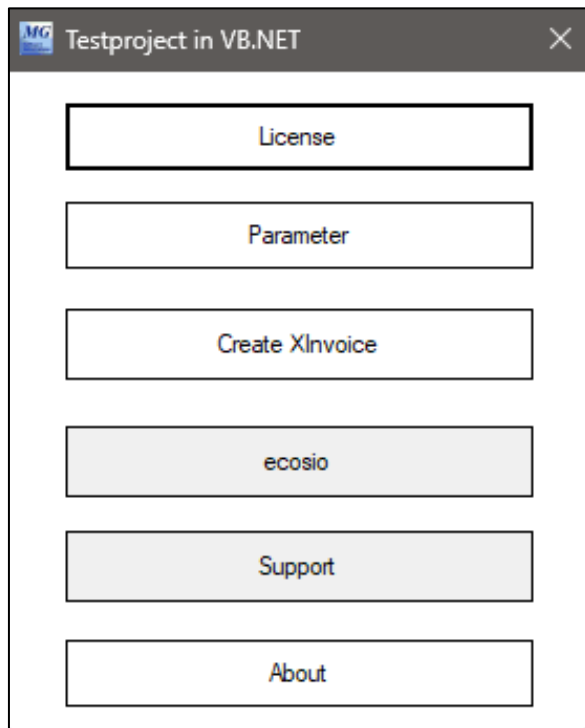
Inhalt

Vorwort	1
Beispiel-Programm	2
Quellcode VB	4
Quellcode C#	6
ecosio	7
Support.....	7
About	7
Abschluss.....	7

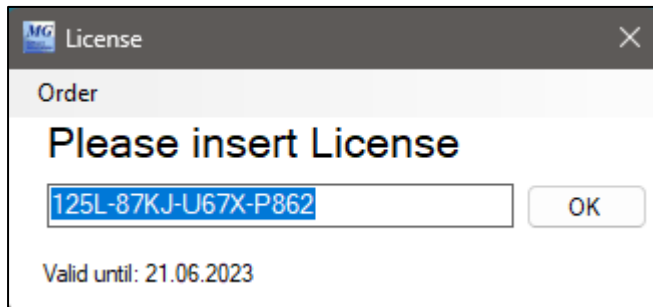
Beispiel-Programm

Die mitgelieferten Beispiel-Programme sind in Visual Basic.NET und C# geschrieben und zeigt Ihnen die Verwendung der DLL.

Da es sich hierbei um eine 32/64 Bit DLL handelt, die in VB.NET geschrieben wurde, können Sie die nahezu in allen Programmierumgebungen einsetzen, die .NET unterstützen.



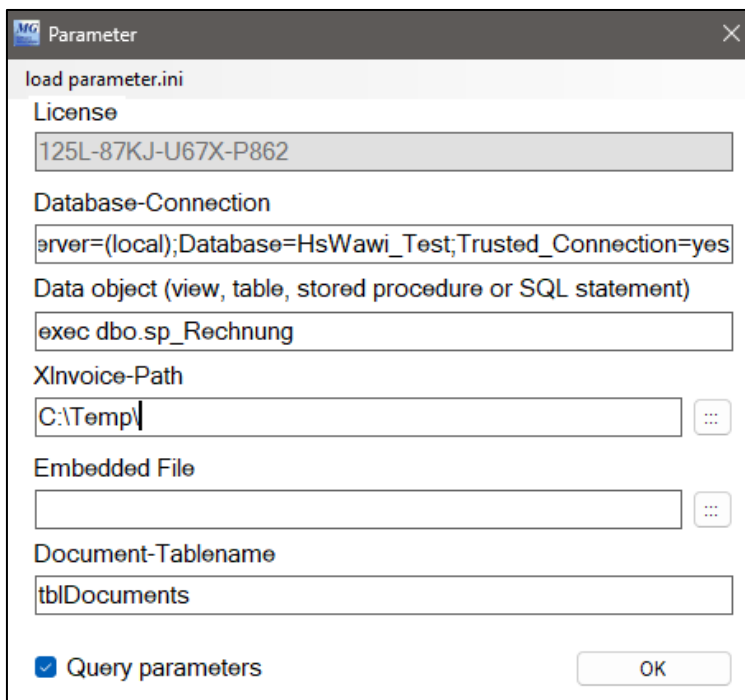
Da es sich um 2 Beispiele handelt, die ähnlich funktionieren, erläutere ich das VB.NET Projekt. Abgesehen von der Verwendung und Integration gibt es keine nennenswerten Unterschiede, wie der Abschnitt in C# zeigt.



Hier finden Sie die Verwendung Ihrer Seriennummer und über den Menüpunkt „Order“ können Sie Ihre Lizenz verlängern oder eine Bestellen.

Die erstmalige Verwendung der DLL erlaubt Ihnen, 30 Tage den vollen Funktionsumfang zu testen, oder 15 digitale Rechnungen zu Erstellen. Valid until zeigt, wann das Gültigkeitsdatum abgelaufen ist.

Bitte beachten Sie, dass die Lizenz pro PC gilt. Sollten Sie Dienstleister oder Software-Hersteller sein, so sprechen Sie mich an, um eine optimale Lizenz zu erhalten.



In diesem Beispielprogramm können Sie die Parameter über ein Fenster einstellen. Dieses ist Optional, denn wahrscheinlicher Weise werden Sie die DLL direkt aus Ihrem Quellcode ansprechen wollen.

Erläuterung der Schnittstelle

Die Daten für die XRechnung kommen aus einer Schnittstelle. Diese zeigt in diesem Fall auf einen Microsoft SQL-Server und die Daten liefert eine gespeicherte Prozedur Namens: sp_Rechnung, die wiederum ein Parameter „@BelegNr“ erfragt. Der Aufruf der Daten steht in der Variable „SQL“. Ob ein Parameter abgefragt wird, wird über True/False festgelegt. Die Schnittstelle liefert die Daten und ganz wichtig sind die einzelnen Spalten. Da hier eine rechtliche Vorgabe herrscht, müssen die Spalten bestimmte Namen tragen. Die Spalten werden als BT-Felder bezeichnet. Diese gefolgt von einer Nummer. Sie können also aus Ihrer Anwendung heraus, eine Schnittstelle schaffen, die alle notwendigen BT-Felder beinhaltet und die DLL erstellt ihre digitale Rechnung. Eine genaue Vorgabe und Versionierung der BT-Felder finden Sie hier

[Versionen und Bundles der XRechnung - XStandards Einkauf \(xeinkauf.de\)](#)

Quellcode VB

```
Imports MGSXRechnungDLL

Public Class Form1

    Dim SerienNumber As String = "125L-87KJ-U67X-P862"
    Dim ExportPath As String = "C:\Temp\"
    Dim DBConnect As String = "Driver={SQL Server};Server=(local);Database=HsWawi_Test;Trusted_Connection=yes"
    Dim SQL As String = "exec dbo.sp_Rechnung"
    Dim Parameter As Boolean = True
    Dim ImportFileName As String = "C:\Temp\Rechnung_2023-11.pdf"
    Dim TableName As String = "tblDocuments"
```

Wie Sie dem Beispielcode entnehmen können, benötigen Sie nicht viel, um eine XRechnung erstellen zu können. Ich habe die Variablen mit vorgefertigten Daten befüllt, um diese in die DLL zu verarbeiten.

```
Private Sub btnCreate_Click(sender As Object, e As EventArgs) Handles btnCreate.Click
    Dim bolConnect As Boolean
    Dim strParameter As String

    If Parameter = True Then
        strParameter = InputBox("Please insert Parameter", "Parameter", 3)
        If strParameter <> "" Then
            Using cls As MGSXRechnungDLL.clsAufruf = New MGSXRechnungDLL.clsAufruf
                bolConnect = cls.DLL_Init_DB(SerienNumber,
                    ExportPath & strParameter & ".xml",
                    DBConnect,
                    SQL, strParameter, Parameter,
                    ImportFileName,
                    TableName)

                If bolConnect = False Then
                    MsgBox("Connect to database is not possible." & vbCrLf & "Connection string incorrectly constructed.", vbInformation, "Note")
                Else
                    If cls.CreateXRechnung = True Then
                        If MsgBox("Validate XInvoice with ecosio?", vbYesNo + vbQuestion, "Note") = vbYes Then
                            Call Ecosio()
                        End If
                    End If
                End If
            End Using
        Else
            MsgBox("Parameter is empty", vbInformation, "Note")
        End If
    Else
        strParameter = InputBox("Please insert filename (without path and extension)", "Filename")
        Using cls As MGSXRechnungDLL.clsAufruf = New MGSXRechnungDLL.clsAufruf
            bolConnect = cls.DLL_Init_DB(SerienNumber,
                ExportPath & strParameter & ".xml",
                DBConnect,
                SQL, "", Parameter,
                ImportFileName,
                TableName)

            If bolConnect = True Then
                If cls.CreateXRechnung = True Then
                    If MsgBox("Validate XInvoice with ecosio?", vbYesNo + vbQuestion, "Note") = vbYes Then
                        Call Ecosio()
                    End If
                End If
            End If
        End Using
    End If
End Sub
```

Das ist die Initialisierung und benötigt die Parameter, um die Daten zu beziehen.

Hat die Verbindung geklappt, kann die XRechnung erstellt werden.

Quellcode C#

```
using System;
using System.Windows.Forms;

namespace WinXRechnungDLLTest_C-Sharp
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void btnCreate_Click(object sender, EventArgs e)
        {
            string SerienNumber = "125L-87KJ-U67X-P862";
            string ExportPath = @"C:\Temp\Export.xml";
            string DBConnect = "Driver={SQL Server};Server=(local);Database=HsWawi_Test;Trusted_Connection=yes";
            string SQL = "exec dbo.sp_Rechnung";
            bool Parameter = true;
            string ImportFileName = string.Empty;
            string TableName = "tblDocuments";
            string strParamter = "39994";
            bool dbconnect = false;

            using (MGSXRechnungDLL.clsAufruf cls = new MGSXRechnungDLL.clsAufruf())
            {
                dbconnect = cls.DLL_Init_DB(SerienNumber,
                                           ExportPath,
                                           DBConnect,
                                           SQL, strParamter, Parameter,
                                           ImportFileName,
                                           TableName);

                if (dbconnect == true)
                {
                    if (cls.CreateXRechnung() == true)
                    {
                        MessageBox.Show("Perfect, your invoice be created", "Note");
                    }
                }
            }
        }
    }
}
```

Der Abschnitt in C# zeigt lediglich die Erstellung der XRechnung. Wie Sie sehen, ist die Methodik relativ einfach aus dem VB-Projekt abzuleiten.

Achtung:

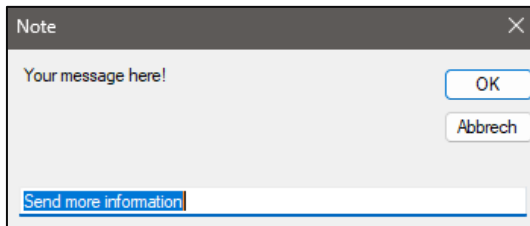
Sollte ihrer elektronischen Rechnung eine Datei beigefügt werden, so müssen Sie diese ebenfalls via Parameter mitgeben. Hier ist nur eine Datei vorgesehen. Achten Sie daher auf das Volumen und das Format. Im Regelfall handelt es sich um eine PDF-Datei, die mit in der XML-Datei eingeschlossen wird.

ecosio

Über ecosio haben Sie die Möglichkeit, die XML-Dateien auf Gültigkeit und Fehler zu überprüfen.

<https://ecosio.com/de/peppol-und-xml-dokumente-online-validieren/>

Support



Sie können mir eine E-Mail zum Thema zusenden.

About



Besuchen Sie mich doch mal im Netz. Schreiben eine Nachricht oder in dringenden Fällen können Sie mich auch telefonisch erreichen.

Abschluss

Sollten Sie Anpassungen, Fragen, Wünsche oder einfach nur Hilfe benötigen, so freue ich mich auf Ihre Nachricht.

Und nun viel Erfolg mit MGS XRechnung.NET DLL

Michael Grube

(Programmautor)