

Generowanie faktury zakupu w Comarch ERP XL z wykorzystaniem skryptów C#

Spis treści artykułu

- [Informacje podstawowe](#)
- [Konfiguracja typu obiegu](#)
- [Generowanie dokumentu FZ do Comarch ERP XL – jak działa](#)

Informacje podstawowe

Uwaga

Aby przykład działał poprawnie, konieczne jest spełnienie następujących warunków:

1. [Instalacja systemu Comarch ERP XL na serwerze IIS](#)
2. [Poprawna konfiguracja współpracy pomiędzy systemami Comarch ERP XL oraz Comarch BPM \(dawniej DMS\)](#)
3. [Nadanie użytkownikowi IISUSER uprawnień do zapisu w katalogu bin\Scriptcs_bin oraz katalogu bin\.scriptcs_cache, które znajdują się w plikach aplikacji serwerowej Comarch BPM \(dawniej DMS\)](#)
4. [Skopiowanie biblioteki cdn_api\[wersja\].net.dll do katalogu bin\Scriptcs_bin](#), biblioteka znajduje się w katalogu, w którym zainstalowano Comarch ERP XL, w przykładach wykorzystano cdn_api20232.net.dll
5. Należy sprawdzić, czy w katalogu bin aplikacji web Comarch BPM (dawniej DMS) znajduje się plik cdn_sys.dll, jeżeli jest

należy **usunąć go** z tej lokalizacji.

Skrypty C# zostały wykorzystane w kontrolkach typu Własna akcja i Komunikat. Skrypt dla kontrolki typu własna akcja utworzono na podstawie wzorca API XL

Wskazówka

Rekomendujemy wykorzystanie wzorca API XL jako elementu bazowego na potrzeby własnych rozwiązań.

Wzorzec API XL dostępny jest na indywidualnych stronach dla Partnerów w obszarze Comarch BPM\Przykłady\ERP XL: [Wzorzec API XL](#)

Konfiguracja typu obiegu

Przykład pokazuje, w jaki sposób administrator Comarch BPM (dawniej DMS) może skonfigurować definicję obiegu, aby generować dokument „Faktura zakupu” do systemu Comarch ERP XL, bez pomocy kontrolki typu Dokument ERP XL. W tym celu należy:

1. **dodać nowy typ obiegu** w ramach zakładki  **[Definicje obiegu dokumentów]**
2. **nadać nazwę typowi obiegu** – np. *Generowanie faktury zakupu* i prefix – np. GEN
3. **nadać danemu administratorowi uprawnienie do inicjowania dokumentu** na zakładce „Ustawienia obiegu” w sekcji „Uprawnienia do inicjowania dokumentu”

Generowanie faktury zakupu

Prefix: GEN

KARTA OBIEGU SCHEMAT OBIEGU **USTAWIENIA OBIEGU**

Ogólne

▼

Lista dokumentów jako książka nadawcza

▼

Ustawienia dla aplikacji mobilnej

▼

Filtry

▼

Uprawnienia do inicjowania dokumentu

▲

Nazwa spółki	Typ	Wartość	
Firma X	Pracownik	Administrator systemu	
Dodaj			
Obserwator dokumentu			

Nadawanie uprawnień do inicjowania dokumentu dla administratora – przykład

4. W ramach zakładki „**Schemat obiegu**” **utworzyć etap początkowy „Generuj dokument” i etap końcowy „Koniec” oraz dodać uprawnienie dla danego administratora do etapów.**

Generowanie faktury zakupu

Prefix: GEN

←

🏠

⚙️

🗑️

KARTA OBIEGU

SCHEMAT OBIEGU

USTAWIENIA OBIEGU

🏠

📄

👤

ACD

👥

🔍

⚙️

👤

📱

Etap

Szukaj

🗑️ ⬆️ ⬇️ ➕

Generuj dokument

Koniec

Etap

OGÓLNE

UPRAWNIENIA

PRZEKAZANIE

MAIL

KONTROLKI

Cofnij dokument

Właściciel dokumentu

Uprawnienia do etapu

Uprawnienia dodatkowe

Zawężanie uprawnień na wcześniejszym etapie

Dostęp do dok. dla operatora niepracującego z dok.

Uprawnienia

Nazwa spółki	Typ	Wartość
Firma X	Pracownik	Administrator systemu
Dodaj		

Dodawanie etapów i uprawnień

5. W ramach zakładki „Karta obiegu” **dodać następujące kontrolki:**

A Kontrolka typu Kontrahent – „**Kontrahent**” (identyfikator: Kontrahent)

Kontrahent	
Nazwa wyświetlana	Kontrahent
Nazwa (identyfikator)	Kontrahent
Prezentuj na całej szerokości	☐
Ustaw kontrolkę na początku wiersza	☐
Nazwa spółki	Firma X
Pokaż na liście	☐
Inicjowanie wartości	
Inicjowanie	Brak
Zmiana wartości (Obserwator)	
Kontrolki powiązane	Dodaj
Wzór na wartość	Wyrażenie
	Ustaw

Kontrolka „Kontrahent”

B Kontrolka typu Dane tabelaryczne – „Elementy”

(identyfikator: Elementy) – w ramach kontrolki należy:

- w zakładce „Listy” dodać następujące listy:
 - „Jm” z wartością „Jednostki miary” w polu „Inicjowanie”
 - „VAT” z wartością „Stawki VAT” w polu „Inicjowanie”

Dane tabelaryczne

OGÓLNE LISTY INICJOWANIE WARTOŚCI OGRANICZENIE EDYCJI

Lista 1

Nazwa

Jm

Inicjowanie

Jednostki miary

Rodzaj identyfikatora

Int

Nazwa spółki

Firma X

Pokaż

Usuń

Tryb pracy

Lista

Lista 2

Nazwa

VAT

Inicjowanie

Stawki VAT

Rodzaj identyfikatora

Int

Nazwa spółki

Firma X

Pokaż

Usuń

Tryb pracy

Lista

Dodaj

Kontrolka „Elementy”, zakładka „Listy”

- w zakładce „**Ogólne**” dodać następujące kolumny:
 - kolumna typu **Towar** – „**Towar**” (identyfikator: Towar)
 - kolumna typu **Liczba stałoprzecinkowa** – „**Ilość**” (identyfikator: Ilosc)
 - kolumna typu **Lista** – „**Jm.**” (identyfikator: Jm) z wartością „**Jm**” wybraną w polu „**Listy**”
 - kolumna typu **Lista** – „**VAT**” (identyfikator: VAT) z wartością „**VAT**” wybraną w polu „**Listy**”
 - kolumna typu **Liczba stałoprzecinkowa** –

- „Cena netto” (identyfikator: CenaNetto)
- kolumna typu Liczba stałoprzecinkowa – „Cena brutto” (identyfikator: CenaBrutto)
- kolumna typu Liczba stałoprzecinkowa – „Wartość Netto” (identyfikator: WartoscNetto)
- kolumna typu Liczba stałoprzecinkowa – „Wartość Brutto” (identyfikator: WartoscBrutto)

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Nazwa wyświetlana

Elementy

Nazwa (identyfikator)

Elementy

Prezentuj na całej szerokości

☒

Ustaw kontrolkę na początku wiersza

☐

Liczba prezentowanych wierszy

5

Kolumny

Kolumna 1

Towar

Nazwa kolumny

Towar

Identyfikator kolumny

Towar

Nazwa spółki

Firma X

Kolumna 2

Liczba stałoprzecinkowa

Nazwa kolumny

Ilość

Identyfikator kolumny

Ilość

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość

2

Kolumna 3

Lista

Nazwa kolumny

Jm.

Identyfikator kolumny

Jm

Listy

Jm

Kontrolka „Elementy”, zakładka „Ogólne” – widok kolumn 1-3

Dane tabelaryczne

Kolumna 4

Lista



Nazwa kolumny

VAT

Identyfikator kolumny

VAT

Listy

VAT



Kolumna 5

Liczba stałoprzecinkowa



Nazwa kolumny

Cena netto

Identyfikator kolumny

CenaNetto

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość

2



Kolumna 6

Liczba stałoprzecinkowa



Nazwa kolumny

Cena brutto

Identyfikator kolumny

CenaBrutto

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość

2



Kolumna 7

Liczba stałoprzecinkowa



Nazwa kolumny

Wartość Netto

Identyfikator kolumny

WartoscNetto

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość

2



Kolumna 8

Liczba stałoprzecinkowa



Nazwa kolumny

Wartość Brutto

Identyfikator kolumny

WartoscBrutto

Liczba miejsc po przecinku, do której będzie zaokrąglana wartość

2



Kontrolka „Elementy”, zakładka „Ogólne” – widok kolumn

- w zakładce „Inicjowanie wartości” :
 - w ramach pola „Kontrolki powiązane” kliknąć przycisk „Dodaj” i wybrać kontrolkę „Elementy”
 - w ramach pola „Wzór na wartość” wybrać opcję „SQL OD”, kliknąć „Ustaw”, a następnie **wprowadzić następujące zapytanie:**

```

if @^InitSenderControlName@ = 'Elementy'
Begin

    declare @vat decimal(22,4) = 0.0
    declare @netto decimal (22,4) = 0.0
    declare @brutto decimal (22,4) = 0.0
    declare @ilosc decimal (22,4) = 0.0

    if @Elementy_Column@ in (2,4,5)
    Begin
        select @netto = CenaNetto, @ilosc = Ilosc
    from   @Elementy@
    where  POS = @Elementy_Row@
        select @vat = case
            when VAT = 1 then 23
            when VAT = 2 then 8
            when VAT = 3 then 0
            when VAT = 4 then 0
            when VAT = 5 then 0
            when VAT = 6 then 7
            when VAT = 7 then 5
            else 0 end
        from   @Elementy@ where POS = @Elementy_Row@
        set @brutto = ((@netto*@vat)/100)+@netto

        update @Elementy@ set CenaBrutto = @brutto,
        WartoscBrutto = @ilosc * @brutto,
        WartoscNetto = @ilosc * @netto where POS =
        @Elementy_Row@
        select * from @Elementy@
    End

```

```

if @Elementy_Column@ = -1
Begin
    update @Elementy@ set Ilosc = 1, JM = 1, VAT = 1
    where POS = @Elementy_Row@
    select * from @Elementy@
End
End

```

Wprowadzone zapytanie należy **zapisać**, klikając w przycisk „Zapisz” w oknie „Wyrażenie kontrolki”

Dane tabelaryczne

OGÓLNE
LISTY
INICJOWANIE WARTOŚCI
OGRANICZENIE EDYCJI

Inicjowanie wartości

Inicjowanie

Brak ▼

Kontrolki powiązane

Elementy (Elementy)

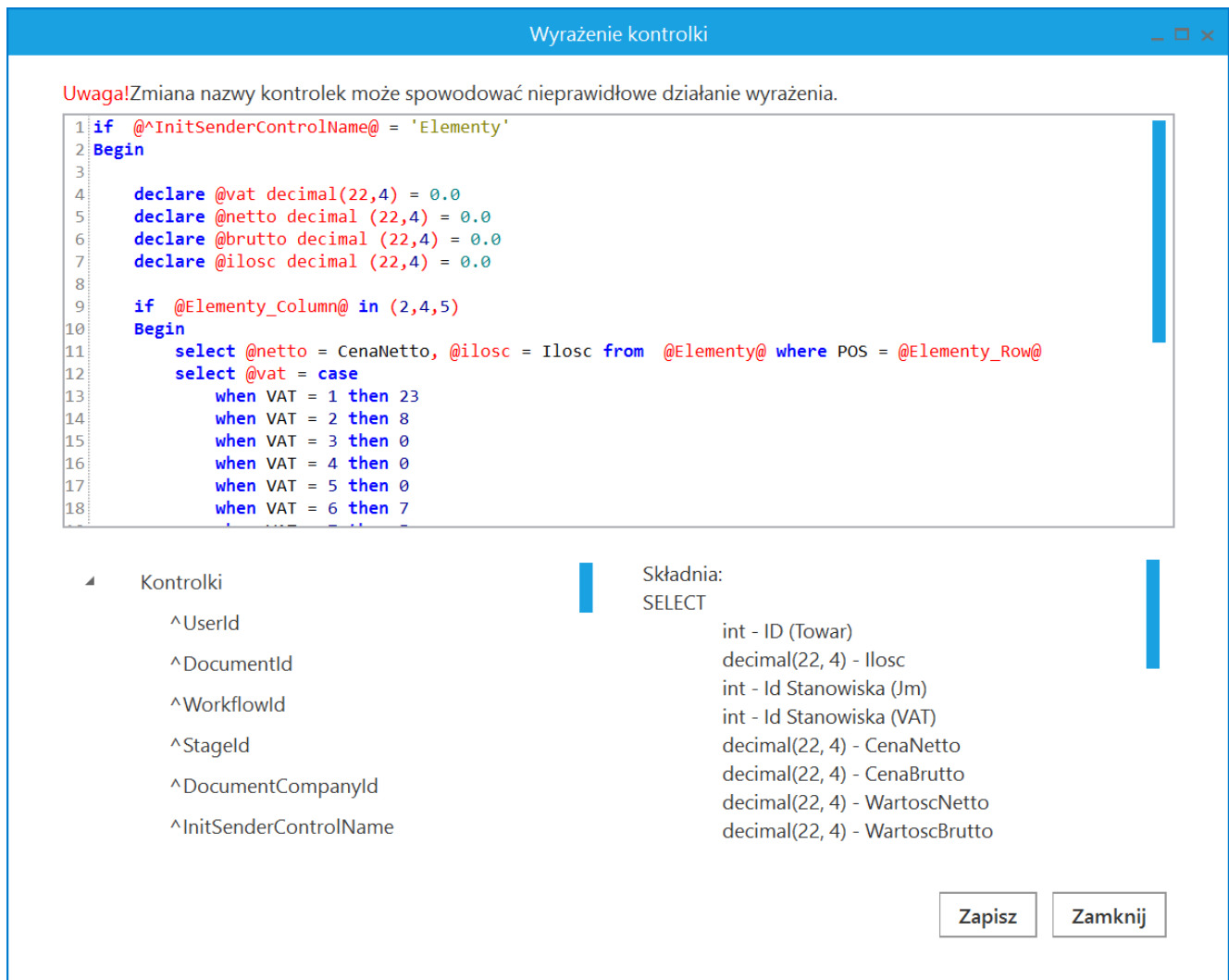
Dodaj

Wzór na wartość

▼

Ustaw

Kontrolka „Elementy”, zakładka „Inicjowanie wartości”



Okno „Wyrażenie kontrolki” z wprowadzonym zapytaniem

- w zakładce „Ograniczenie Edycji” zaznaczyć checkboxy na obydwóch etapach dla kolumn: „Cena netto”, „Cena brutto”, „Wartość Netto”, „Wartość Brutto”

Dane tabelaryczne

OGÓLNE

LISTY

INICJOWANIE WARTOŚCI

OGRANICZENIE EDYCJI

Zaznaczenie opcji spowoduje zablokowanie funkcjonalności lub edycji poniższych parametrów.

Etap	☐ Zaznacz wszystko Zablokuj dodanie wierszy	☐ Zaznacz wszystko Zablokuj usuwanie wierszy	☐ Zaznacz wszystko Towar Towar	☐ Zaznacz wszystko Ilość Ilość	☐ Zaznacz wszystko Jm Jm.	☐ Zaznacz wszystko VAT VAT	☐ Zaznacz wszystko CenaNetto Cena netto	☒ Zaznacz wszystko CenaBrutto Cena brutto	☒ Zaznacz wszystko WartośćNetto Wartość Netto	☒ Zaznacz wszystko WartośćBrutto Wartość Brutto
Generuj dokument	☐	☐	☐	☐	☐	☐	☐	☒	☒	☒
Koniec	☐	☐	☐	☐	☐	☐	☐	☒	☒	☒

Kontrolka „Elementy”, zakładka „Ograniczenie Edycji”

C kontrolka typu Własna akcja – „Generuj dokument” (identyfikator: GenerujDokument) – w ramach definicji kontrolki należy kliknąć w link „Ustaw” w polu „C# Script”, następnie w otwartym oknie „Skrypt C#” w polu „Miejsce wykonania skryptu” wybrać „Proces (exe)” (jeśli istnieje opcja wyboru) i wprowadzić następujący kod:

```
using System.Threading;
using System.Runtime.InteropServices;
using cdn_api;

static ThreadLocal<bool> threadToClarionAttached = new
ThreadLocal<bool>(() => false);

[DllImport("ClaRUN.dll")]
private static extern void AttachThreadToClarion(int flag);
public void AttachThreadToClarion()
{
    try
    {
        if (!threadToClarionAttached.Value)
        {
            AttachThreadToClarion(1); // C8
            threadToClarionAttached.Value = true;
        }
    }
    catch (DllNotFoundException ex)
    {
        // -----
        // Tu przekaz informacje o statusie działania skryptu
        // -----
        Globals.MainFrame.GenerujDokument_CSSSCRIPT.Text =
ex.Message;
    }
}
```

```

AttachThreadToClarion();

int _lSesjaID = 0;
string blad = "";

try
{
    int documentId = 0;
    int wersjaApi = odpowiedniawersjaapi;

    var loginInfo = new XLLoginInfo_odpowiedniawersjaapi()
    {
        Wersja = odpowiedniawersjaapi,
        ProgramID = "Comarch BPM",
        Winieta = -1,
        TrybWsadowy = 1,
        Baza = "nazwafirmy",
        OpeIdent = "Login",
        OpeHaslo = "hasło",
        SerwerKlucza = @"serwerklucza"
    };

    var XLLoginResult = cdn_api.cdn_api.XLLogin(loginInfo, ref
_lSesjaID);
    if (XLLoginResult != 0)
    {
        blad = "Funkcja XLLogin zwróciła błąd nr: " +
XLLoginResult;
        throw new Exception();
    }

    // -----
-----
    // Tu dodaj kod obsługi API
    // -----
-----

        var dokumentNagInfo = new
XLDokumentNagInfo_odpowiedniawersjaapi
    {
        Wersja = odpowiedniawersjaapi,

```

```

        Typ = 1521, //FZ
        KntTyp = 32,
        KntNumer = Globals.MainFrame.Kontrahent.Id ?? 0, //
Pobierz GidNumer kontrahenta
        RodzajZakupu = 1,
        //Avista = 1
    };

    var XLNowyDokumentResult = cdn_api.cdn_api.XLNowyDokument
(_lSesjaID, ref documentId, dokumentNagInfo);
    if (XLNowyDokumentResult != 0)
    {
        blad = "Funkcja XLNowyDokument zwróciła błąd nr: " +
XLNowyDokumentResult;
        throw new Exception();
    }

    //var element;

    for (int i=0; i<Globals.MainFrame.Elementy.RowCount; i++)
    {

        var element = new
XLDokumentElemInfo_odpowiedniawersjaapi();

        element.Wersja = odpowiedniawersjaapi;
        element.TwrTyp = 16;

        element.Ilosc =
Globals.MainFrame.Elementy.Items[i].Ilosc.Value.ToString();
        element.TwrNumer =
Globals.MainFrame.Elementy.Items[i].Towar.Id;
        element.JmZ =
Globals.MainFrame.Elementy.Items[i].Jm.Text;
        element.Cena =
Globals.MainFrame.Elementy.Items[i].CenaNetto.Value.ToString()
;

        var XLDodajPozycjeResult =
cdn_api.cdn_api.XLDodajPozycje(documentId, element);
        if (XLDodajPozycjeResult != 0)

```

```

        {
            blad = "Funkcja XLDodajPozycje zwróciła błąd nr: "
+ XLDodajPozycjeResult;
            throw new Exception();
        }
    }

    var zamkniecieDokumentuInfo = new
XLZamkniecieDokumentuInfo_odpowiedniawersjaapi
    {
        Wersja = odpowiedniawersjaapi,
        Tryb = 1
    };

    var XLZamknijDokumentResult =
cdn_api.cdn_api.XLZamknijDokument(documentId,
zamkniecieDokumentuInfo);
    if (XLZamknijDokumentResult != 0)
    {
        blad = "Funkcja XLZamknijDokument zwróciła błąd nr: "
+ XLZamknijDokumentResult;
        throw new Exception();
    }
}
catch (Exception ex)
{
    if (blad == "")
        blad = "Nieokreślony błąd: " + ex.Message;
}
finally
{
    var XLLogoutResult = cdn_api.cdn_api.XLLogout(_lSesjaID);
    if (XLLogoutResult != 0)
    {
        blad = "Funkcja XLLogout zwróciła błąd nr: " +
XLLogoutResult;
    }
    // -----
    -----
    // Tu przekaż informacje o statusie działania skryptu

```

```
// -----
-----
if ( blad == "" )
    Globals.MainFrame.GenerujDokument_CSSSCRIPT.Text =
"Operacja zakończona sukcesem.";
else
    Globals.MainFrame.GenerujDokument_CSSSCRIPT.Text =
blad;
}
```

gdzie **należy zmienić następujące wartości:**

odpowiedniawersjaapi – **wersja API ERP XL** zgodna z wersją pliku
cdn_api[wersja].net.dll

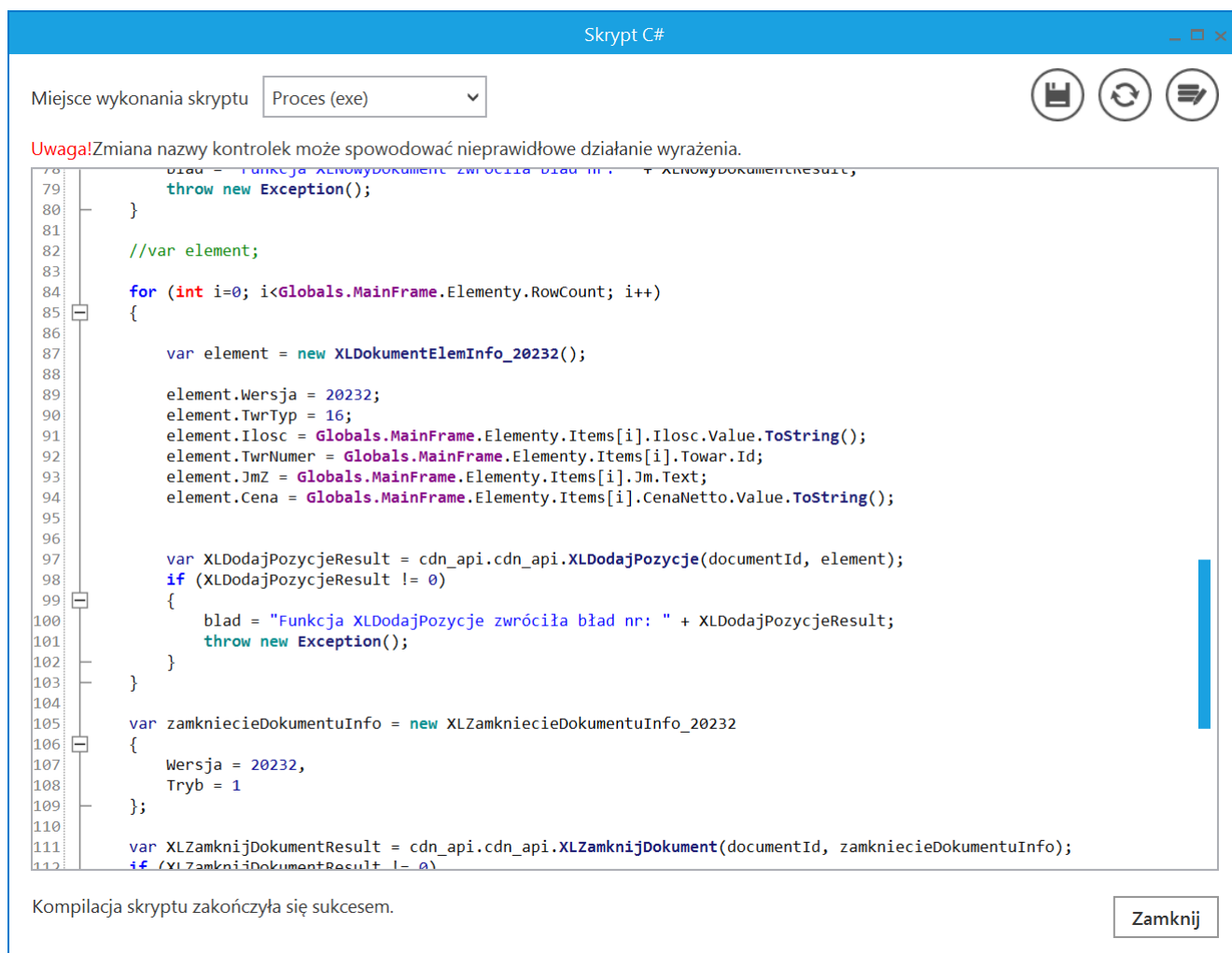
nazwafirmy – **nazwa Firmy Comarch ERP XL** (nazwę firmy można
sprawdzić w managerze baz Comarch ERP XL)

Login – **login operatora Comarch ERP XL**

hasło – **hasło operatora Comarch ERP XL**

serwerklucza – **serwer klucza licencji**

Po wprowadzeniu kodu należy **nacisnąć przycisk**  **[Kompiluj i zapisz]**, a następnie, gdy w dolnej części okna zostanie wyświetlony napis „Kompilacja skryptu zakończyła się sukcesem” nacisnąć przycisk **„Zamknij”**.



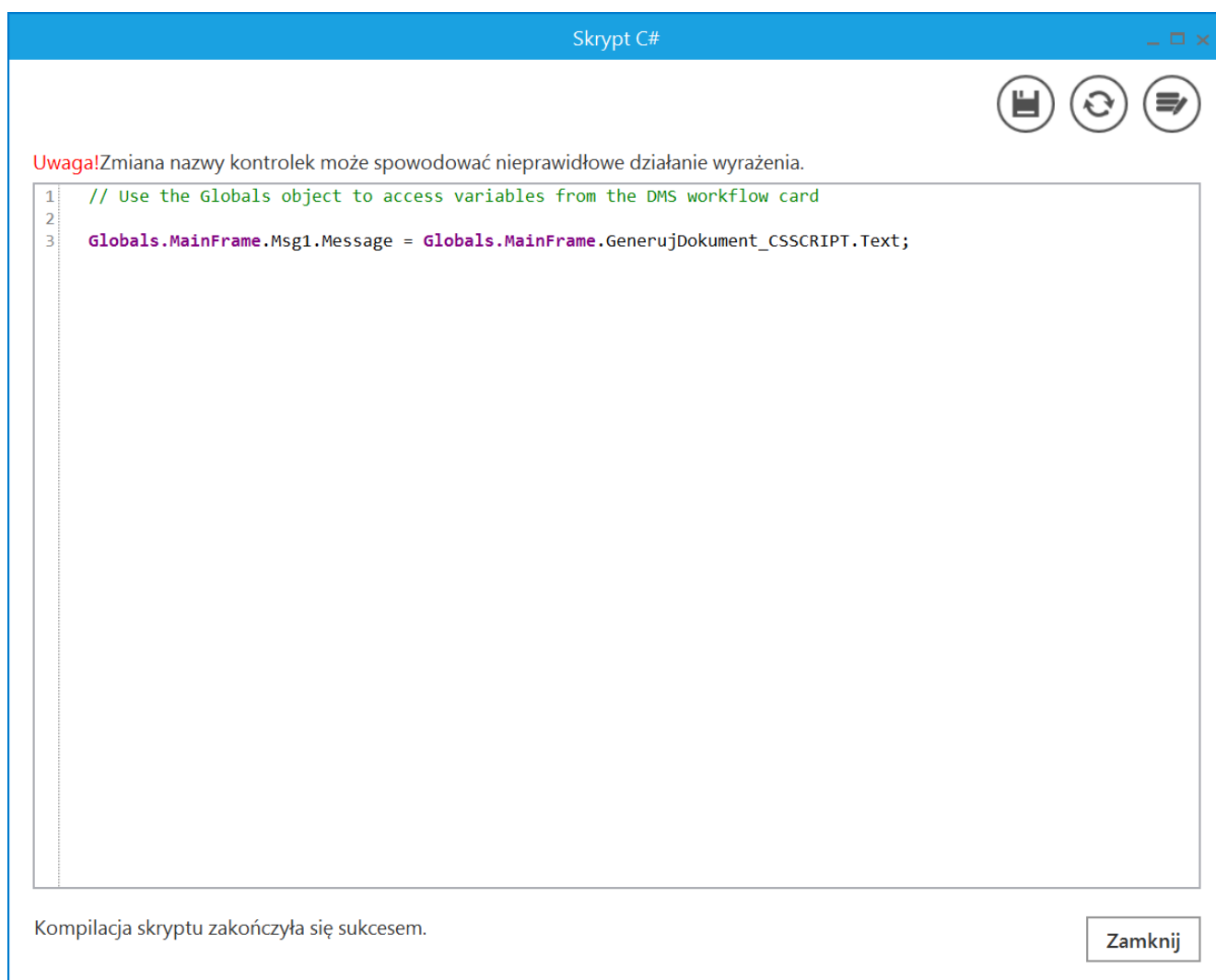
Widok okna „Skrypt C#” po skompilowaniu skryptu

D Kontrolka typu Komunikat – należy w ramach pola:

- „Nazwa (identyfikator)” wprowadzić nazwę „**Msg1**”
- „Tytuł” wprowadzić tekst „**Status wykonania API ERP XL**”
- „Kontrolki powiązane” kliknąć w link „**Dodaj**” i wybrać kontrolkę „**Generuj dokument**”
- „Wzór na wartość” wybrać wartość „**C# Script**”, kliknąć „**Ustaw**”, a następnie wprowadzić następujący kod:

```
Globals.MainFrame.Msg1.Message
= Globals.MainFrame.GenerujDokument_CSSSCRIPT.Text;
```

Po wprowadzeniu kodu należy **nacisnąć przycisk**  **[Kompiluj i zapisz]**, a następnie, gdy w dolnej części okna zostanie wyświetlony napis „Kompilacja skryptu zakończyła się sukcesem” nacisnąć przycisk **„Zamknij”**.



Widok okna „Skrypt C#” po skompilowaniu skryptu

6. **zapisać typ obiegu** za pomocą przycisku  **[Zapisz]**

Zapisywanie typu obiegu

Generowanie dokumentu FZ do Comarch ERP XL – jak działa

Po zdefiniowaniu typu obiegu operator przechodzi na zakładkę



[Dokumenty], klika na typ obiegu „**Generowanie faktury**

zakupu” i naciska przycisk  **[Dodaj]**. Następnie kolejno:

1. **zapisuje** nowy dokument, naciskając  **[Zapisz]**

2. **wybiera kontrahenta** w ramach kontrolki „**Kontrahent**”

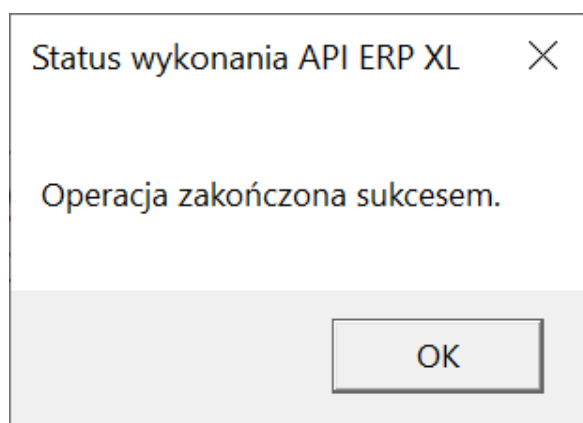
3. w ramach kontrolki „**Elementy**” dodaje towar w kolumnie „**Towar**”, wybiera opcję w kolumnach „**Jm.**” i „**VAT**” oraz wpisuje wartość w kolumnie „**Cena netto**” (jeśli poprawnie skonfigurowano obieg, wówczas kolumny „**Cena brutto**”, „**Wartość Netto**” i „**Wartość Brutto**” powinny zostać uzupełnione automatycznie)

Uwaga

Należy zwrócić uwagę, aby wartość wybrana w kolumnie „VAT” dla danego towaru była zgodna ze stawką przypisaną do tego towaru w Comarch ERP XL.

4. naciska przycisk „**Generuj dokument**” – wówczas następuje generowanie dokumentu – poniżej przycisku widoczne są poruszające się punkty

5. **Jeśli generowanie dokumentu zakończyło się powodzeniem**, wówczas na środku ekranu wyświetlany jest następujący komunikat:



Komunikat wyświetlany, jeśli generowanie dokumentu zakończyło się powodzeniem

Film

Aby zobaczyć przebieg generowania dokumentu do Comarch ERP XL, kliknij w poniższy film:

<https://pomoc.comarch.pl/dms/wp-content/uploads/2024/06/genxl.mp4>

Wygenerowany dokument jest już **widoczny w Comarch ERP XL**:

Zamówienia [FirmaKSI] - Comarch ERP XL 2024.0.3

Dokument - Faktura zakupu [FZ-11/24] - zostanie zmieniony

Ogólne | Kontrahent | Kaucje | Płatności | Nagłówki | VAT | Intrastat | KSeF | Atrybuty | Załączniki

Kontrahent: FIR1 | **Kraj:** Kraków

Netto: 400,00 | **Brutto:** 492,00 | **Kaucje:** 0,00 | **Zapłacono:** 0,00 | **Do zapłaty:** 492,00

Transakcja: Krajowa | **Rodzaj transportu:** 3 | **Transport drogowy**

Płatność: Gotówka | **0 dni** | **Do:** 05.06.2024

Lp.	Towar	Ilość	Jm.	Cena netto	Cena brutto	Nazwa
00001	T1	2,000	szt.	200,000 PLN	246,000 PLN	Towar 1

Filtr: Docelowy | MAG | ☐ Aktualizuj kaucje

Dokument wygenerowany z Comarch BPM (dawniej DMS) do Comarch ERP XL

Rozpoczynasz pracę z Comarch BPM (dawniej DMS) i chcesz dowiedzieć się, jak korzystać z programu? A może masz już podstawową wiedzę o Comarch BPM (dawniej DMS) i chcesz dowiedzieć się więcej?

[Sprawdź Szkolenia Comarch BPM!](#)

[Powrót do początku artykułu](#)