



SDK zur CRM-Word-Schnittstelle

SDK zur

CRM Wordinterface – für Microsoft Dynamics CRM

Zu den Versionen 5.2.0 – 9.2.0

Inhalt

1. Vorwort3
2. Voraussetzungen.....4
3. Funktionsbeschreibung4
4. Technische Funktionsbeschreibung.....4
4.1 Query Expression4
4.2 Retrieve Multiple5
4.3 Aufruf Parameter5
4.4 Rückgabewerte.....5
5. Beispiele7
5.1 Retrieve Multiple mit Query Expression in C#.....7
5.2 Retrieve Multiple mit Query Expression in JavaScript9

1. Vorwort

Vielen Dank, dass Sie sich für unsere [CRM-Word-Schnittstelle](#) entschieden haben.

Nun haben Sie die Möglichkeit, die Funktionalitäten der Word-Schnittstelle für eigene programmatische Anpassungen nutzen zu können.

Bei der CRM-Word-Schnittstelle handelt es sich um ein [Add-On für Microsoft Dynamics CRM](#).

Auf den folgenden Seiten finden Sie eine ausführlichen Software Developers/Development Kit für die Versionen 5.2.0 - 9.2.0 der Word-Schnittstelle für Microsoft Dynamics CRM.

2. Voraussetzungen

Damit alle Funktionalitäten der CRM Word Schnittstelle genutzt werden können, ist eine installierte und funktionsfähige Version der Word Schnittstelle für Microsoft Dynamics CRM ab Version 5.2.0 notwendig.

3. Funktionsbeschreibung

Dieses SDK dient dazu, die Funktionalitäten der Word Schnittstelle in eigenen programmatischen Anpassungen zu nutzen.

Diese Funktionalitäten können Sie zur Bearbeitung, Kontrolle der CRM Daten, die an das Dokument geschickt werden oder zur Automatisierung der Nachbearbeitung des Ergebnisdokumentes nutzen.

Damit diese Funktionalitäten nutzbar sind, ist ein vorhandenes Word Template nötig. Es wird eine Abfrage mit den Parametern „Templateid“, „Entityname“ und „DatensatzId“ an das CRM geschickt das Ergebnis dieser Abfrage ist ein ausgefülltes Dokument.

4. Technische Funktionsbeschreibung

Zu Beginn sind folgende Informationen nötig, welche in eine XML eingefügt werden. Diese sind Requesttype, und können nur die Einträge „Document“ oder „DocumentFile“ sein.

Weiter müssen der Entityname und die Guid des Datensatz, von dem Daten abgefragt werden, die zu nutzende Word Template Guid und als letztes die Server URL mit Organisationsnamen in die XML eingetragen werden.

Es wird eine Query Expression erzeugt. Diese wird per Retrieve Multiple an das CRM geschickt. Das Plug-In der Word Schnittstelle empfängt diese Query Expression und verarbeitet diese.

Bedingungen für die Query Expressen sind:

Zwingend erforderliche Angaben für die Query Expression sind (siehe Beispiel im Anhang):

- als Entitätsnamen nur „donau_wordtemplate“
- als ColumnSet ist nur „donau_automation“ gültig
- es ist die Angabe von 2 Conditions nötig
 - o Attribute „donau_automation“ mit dem Operator „eq“ und als Wert die Parameter XML
 - o Attribute „donau_wordtemplateid“ mit dem Operator „eq“ und als Wert die jeweilige TemplateId
 - o Wird nur eine der beiden Conditions angegeben, erscheint eine Plug-In Fehlermeldung

Als Ergebnis kommt eine Entity Collection zurück. Diese enthält genau ein Attribut mit Namen „donau_automation“. Der Wert (Value) des Attributes enthält entweder eine Guid des Anhangs an dem das Dokument angehangen wurde oder eine Base 64 kodierte Zeichenkette.

4.1 Query Expression

<http://msdn.microsoft.com/en-us/library/gg328300.aspx>

Query Expression werden in Methoden benutzt, die mehr als einen Datensatz als Ergebnis liefern. Wie in der Methode Retrieve Multiple wird mittels der Query Expression das Ergebnis spezifiziert.

4.2 Retrieve Multiple

<http://msdn.microsoft.com/en-us/library/microsoft.xrm.sdk.iorganizationservice.retrievemultiple.aspx>

Gibt eine Liste von Entity Instanzen die per Query spezifiziert sind.

4.3 Aufruf Parameter

Parameter:

Requesttype

Mögliche Werte:

Type: String, Wert: Document

Type: String, Wert: DocumentFile

entityname

Type: String, Wert: Schemaname der Entität von der Daten abgefragt werden.

entityid

Type: String, Wert: GUID des Datensatzes von dem die Daten abgefragt werden.

serverurl

Type: String, Wert, URL des Organization Service

Beispiel: <http://ServerUrl/Organisationsnamen/XRMServices/2011/Organization.svc>

templatetype

Type: String, Wert: donau_wordtemplate

responsetype

Type: String, Wert: Format des Dokuments als Docx oder Pdf

instantupload

Type: String, Wert: direkter Upload nach der Erstellung als true oder false

Beispiel:

```
<cwirequest>
  <requesttype>Document</requesttype>
  <entityname>" + entityName + "</entityname>
  <entityid>" + entityId + "</entityid>
  <serverurl>" + serverUrl + "</serverurl>
  <templatetype>donau_wordtemplate</templatetype>
  < responsetype >" + responseType + "</ responsetype >
  < instantupload >" + instantUpload + "</ instantupload >
</cwirequest>
```

4.4 Rückgabewerte

Es werden durch die Abfrage per „Retrieve Multiple“ 2 unterschiedliche Ergebnisse geliefert je nach angegebenem Request Type.

Das Ergebnis der Abfrage mit der Methode „Retrieve Multiple“ ist vom Typ „Entity“ und hat im Attribute „donau_automation“ als Wert die Ergebnis XML.

Je nach ausgewähltem Request Type erhält man eine GUID oder das Dokument als Base64 Zeichenkette.

Als Ergebnis erhält man eine Entität mit einem Attribute mit Namen „donau_automation“ und im Value Steht das Ergebnis.

1. Bei der Angabe des Request Type „Document“ erhält man im Value die GUID der Notiz an der das Dokument angehängt wurde
2. Beim Request Type „DocumentFile“ erhält man als Ergebnis einen base64 codierten String.

5. Beispiele

5.1 Retrieve Multiple mit Query Expression in C#

```
//Definition der Parameter: EntityName und EntityId
String entityName = "quote";
String entityId = "BACEA15E-D06E-E211-807B-00155D001B13";
Guid templateId = new Guid("A9350428-DA3A-E211-90AB-00155D00F32B");
String orgUrl = "https://Organisationsnamen.crm4.dynamics.com";

String connectionString = $"Url={orgUrl}; Username={USER}; Password='{PASSWORD}';
AuthType=OAuth; AppId=51f81489-12ee-4a9e-aaae-a2591f45987d; RedirectUri=app://58145B91-0C36-
4500-8554-080854F2AC97; LoginPrompt=Auto";

// Definition der Parameter
//RequestType Document
const string requestType = "Document";
//RequestType Documentfile
//const string requestType = "DocumentFile";

// Definition der Parameter
//ResponseType Docx
const string responseType = "Docx";
//ResponseType Pdf
//const string responseType = "Pdf";

// Definition der Parameter
// direkter Upload nach Erstellung
const string instantUpload = "true";
// nur ausgeben des Dokumentes (ohne Upload)
// const string instantUpload = "false";

String parameters = @"<cwirequest> +
                    <instantupload> + instantUpload + "</instantupload >" +
                    <responsetype> + responseType + "</responsetype>" +
                    <requesttype> + requestType + "</requesttype>" +
                    <entityname> + entityName + "</entityname>" +
                    <entityid> + entityId + "</entityid>" +
                    <serverurl> + orgUrl + "</serverurl>" +
                    "</cwirequest>";

//Definition der Condition
var condition = new ConditionExpression("donau_automation",
                                       ConditionOperator.Equal,
                                       parameters);

//Definition des Filters
var filter = new FilterExpression(LogicalOperator.And);
filter.Conditions.Add(condition);

//TemplateId
var idCondition = new ConditionExpression("donau_wordtemplateid",
                                       ConditionOperator.Equal,
                                       templateId);

filter.Conditions.Add(idCondition);

//Erstellung der Abfrage, mittels Funktion „QueryExpression“
QueryExpression query = new QueryExpression("donau_wordtemplate");
```

```
query.ColumnSet = new ColumnSet("donau_automation");
query.Criteria = filter;

using (CrmServiceClient crmService = new CrmServiceClient(connectionString))
{
    //Message RetrieveMultiple absetzen mit Übergabe der erstellten Query
    EntityCollection responseEntityCol = crmService.RetrieveMultiple(query);

    //Ergebnis GUID (RequestType Document)
    //oder ein Base 64 String (RequestType DocumentFile)
    String result =
        responseEntityCol.Entities[0].Attributes["donau_automation"].ToString();
}
```


5.2 Retrieve Multiple mit Query Expression in JavaScript

```

/// <reference path="Donaubauer.Library.js" />

//Initialisierung
var $crm = Donaubauer.Crm;
var $js = Donaubauer.JavaScript;
var $jq = Donaubauer.$;
var $cs = new Donaubauer.Crm.OrganizationService();

var entityName = "quote";
var entityId = "B30A4B90-B1AC-E211-B889-00155D001B13";
var templateId = "A9350428-DA3A-E211-90AB-00155D00F32B";
var serverUrl = "http://Servername/Organisationsnamen/XRMServices/2011/Organization.svc";

//RequestType Document
var requestType = "Document";

//RequestType DocumentFile
var requestType = "DocumentFile";

// ResponseType Docx
var responseType = "Docx";

//ResponseType PDF
var responseType = "Pdf";

// Instant Upload
var instantUpload = "true";

// kein Instant Upload
var instantUpload = "false";

var parameters = @"<cwirequest>"+
    "<instantupload>" + instantUpload + "</instantupload>"+
    "<responsetype>" + responseType + "</responsetype>"+
    "<requesttype>" + requestType + "</requesttype>"+
    "<entityname>" + entityName + "</entityname>"+
    "<entityid>" + entityId + "</entityid>"+
    "<serverurl>" + serverUrl + "</serverurl>"+
    "</cwirequest>";

//Definition der Condition
var condition = new $crm.ConditionExpression("donau_automation",
    $crm.ConditionOperator.Equal, parameters);

//Definition des Filters
var filter = new $crm.FilterExpression($crm.LogicalOperator.And);
filter.Conditions.push(condition);

//TemplateId
var idCondition = new $crm.ConditionExpression("donau_wordtemplateid",
    $crm.ConditionOperator.Equal,
    new $crm.GuidAttribute(templateId));

filter.Conditions.push(idCondition);

//Erstellung der Abfrage, mittels Funktion „QueryExpression“
var query = new $crm.QueryExpression("donau_wordtemplate");

```

```
query.ColumnSet = new $crm.ColumnSet(["donau_automation"]);
query.Criteria = filter;

var responseEntityCol = $cs.RetrieveMultiple(query);

//enthält entweder eine GUID oder einen Base64 String
var result = responseEntityCol.Entities[0].Attributes.donau_automation.Value;
```

Sollten Sie weitere Fragen zur CRM-Word-Schnittstelle haben oder einen weiterführenden Support benötigen, so zögern Sie bitte nicht uns unter support@donaubauer.com zu kontaktieren.

DONAUBAUER
AKTIENGESELLSCHAFT

Wurzener Straße 10
04774 Dahlen / Sa.

Tel.: +49 (34361) 826-0

Mail: sales@donaubauer.com

Web: www.donaubauer.com

Informationen zur CRM-Word-Schnittstelle: www.crmwordschnittstelle.de

Vorstand: Andreas Donaubauer
Aufsichtsratsvorsitzende: Andrea Striegler
Amtsgericht Leipzig
HRB 23063
Ust.-ID: DE814819651