dox42 D365 CE| Dynamics CRM

Documentation

Christian Bauer, Lisa Pulsinger

TABLE OF CONTENTS

DOCUMENT DETAILS

Version: dox42 Dynamics CRM V 4.4

Author: Christian Bauer, Lisa Pulsinger

Date: 20 February 2023

1. What is dox42 D365 CE|Dynamics CRM?

The dox42 D365 CE|CRM data source enables you to use Dynamics CRM data for document automation. Additionally, other data sources (e.g. SharePoint, XML/JSON, SQL, Webservices, Excel, etc…) can be integrated into one document at the same time.

dox42 D365 CE|CRM is an additional module for dox42. dox42 D365 CE|CRM contains a dox42 Custom Data Source, a dox42 Custom Output Action for MS Dynamics CRM/MS Dynamics 365 for Customer Engagement (D365 CE) and a dox42 solution for easy integration with your MS Dynamics CRM/CE interface.

2. Installation of the dox42 Office Add-Ins

Save the dox42 D365 CE|CRM files to a fixed directory on the computer, on which you want to use dox42. To do that, just unzip the file **dox42DynamicsCRM_V*.zip.**

Start Word/Excel, activate the dox42-Ribbon and select **„Data Map".**
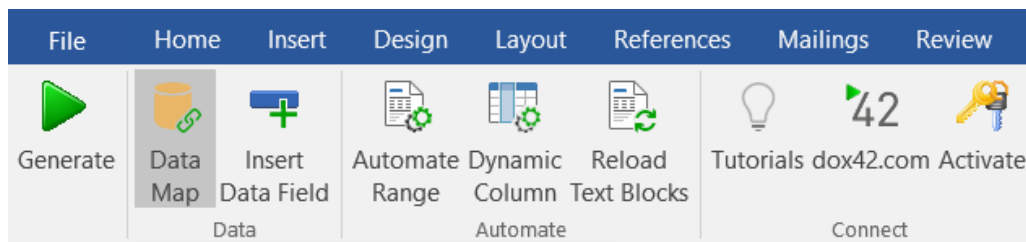
In the Data Map Designer select **„Data Map".** Open the File-Menu and select **„Custom Datasources".**
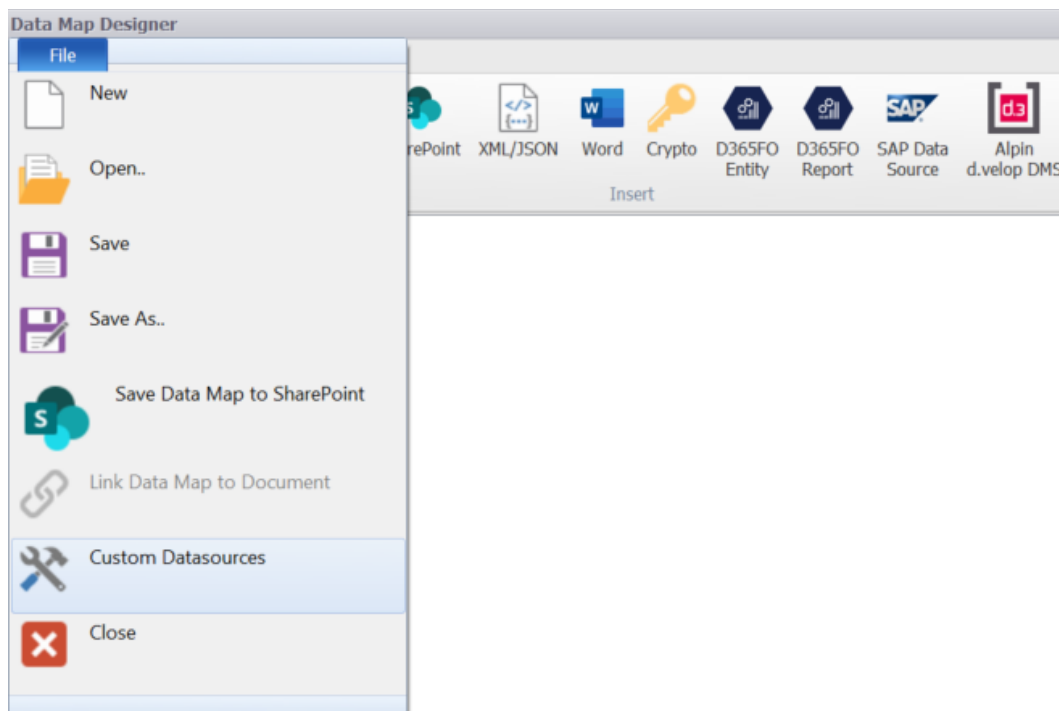
In the dialogue „**Register Custom Datasources",** you can import the Import Configuration **(dox42DynamicsCRMmport.config).**
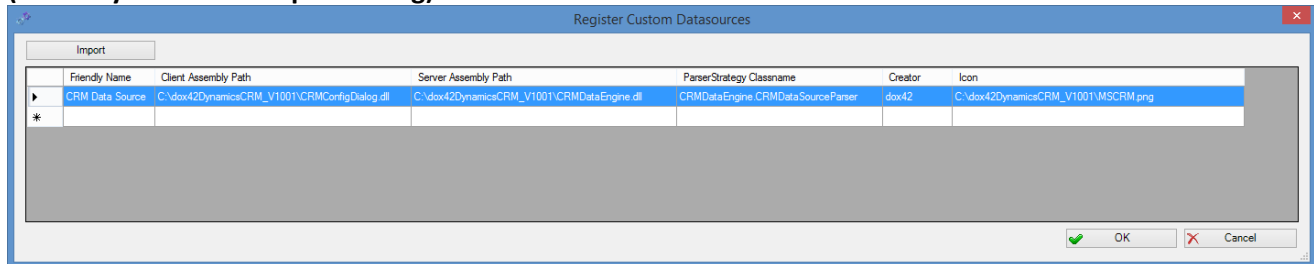
Now, the dox42 D365 CE|CRM Data Source is ready for use as any other dox42 data source.
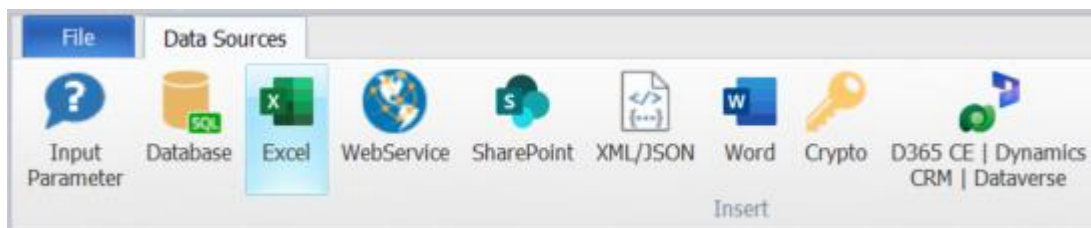
Now you need to activate your license key. To do that, simply select your D365 CE|CRM Data Source, open the configuration dialogue and move to the register **"License".** Insert your license key here and click "Activate".
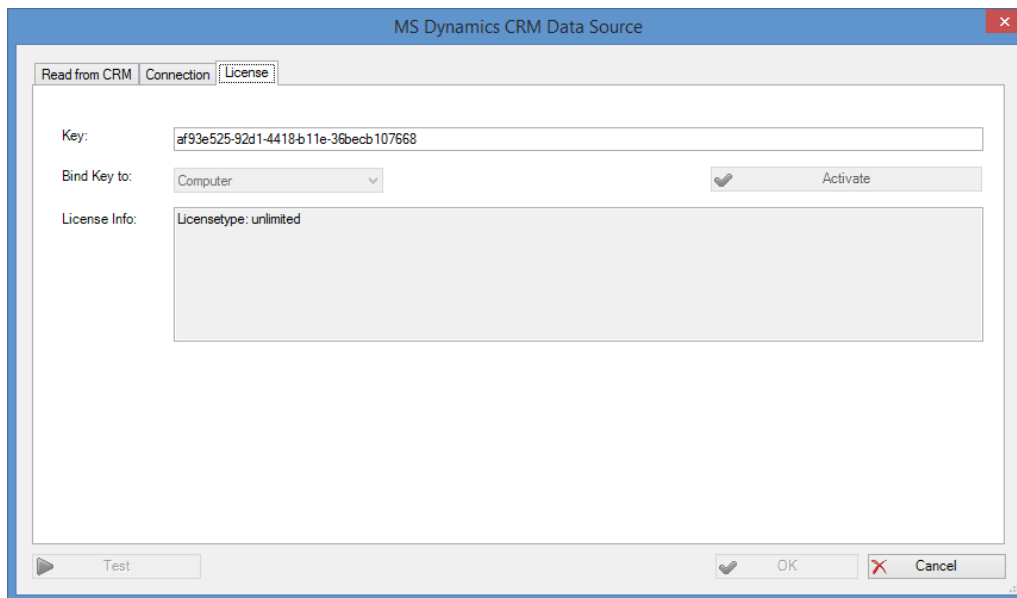
In case you get the error: Microsoft.IdentityModel could not be found, please make sure that Windows Identity Foundation is installed on your machine.
You can activate Windows Identity Foundation via the Control Panel > Turn Windows Features on or off > Tick the box at "Windows Identity Foundation 3.5".

Also you need .NET Framework 4.5.2 or higher installed.

3. Installation of dox42 D365 CE|CRM for dox42 Server (on Premise)

In order to use dox42 D365 CE|CRM on the dox42 Server, please copy the following assemblies to the /Bin directory of your Server:

- ✔ CRMDataEngine.dll
- ✔ CRMLicence.dll
- ✔ CRMOutputAction.dll
- ✔ Microsoft.Crm.Sdk.Proxy.dll
- ✔ Microsoft.Xrm.Sdk.dll
- ✔ Microsoft.Identity.Client.dll

You need .NET Framework 4.5.2 or higher installed on the server.

Now dox42 D365 CE|CRM Data Source has to be registered in the **web.config** of your server.

```
<customDataSources>
        <add key="CRMDataEngine.CRMDataSourceParser"
              value="CRM Data Source;c:\dox42Server\bin\CRMDataEngine.dll;"/>
</customDataSources>

<customOutputActions>
        <add key="CRMOutputAction.CRMOutputAction"
              value="CRMOutputAction; c:\dox42Server\bin\CRMOutputAction.dll;" />
 </customOutputActions>
```

Furthermore, please insert the license key in the **web.config**:

```
<appSettings>
        <add key="LicenseFilePath" value="c:\dox42Server\License" />
        <add key="LicenseKey" value="…"/>
        <add key="dox42CRMLicenseKey" value="…"/>
        …
</appSettings>
```

Also have a look at the dox42 Server Documentation: http://www.dox42.com/Modules/dox42-Server.

4. Data integration with dox42 D365 CE|CRM

4.1. Connection to MS Dynamics CRM or D365 CE

In the data map designer of your add-in, configure the connection to CRM in the register **"Connection".** You can define data fields of any other data source for every value. This way, you could keep the configuration for example in an Excel or XML file. Of course, you can simply enter the values as well. Your username and password can be encrypted with the dox42 Crypto Data Source.

If you connect to MS CRM online (MS Dynamics 365 for Customer Engagement), then your CRM URL would look similar to: https://<Yourcompany>.crm4.dynamics.com

You can also connect to MS CRM 2011, 2013, 2015, 2016 on-premises.

The dox42 D365 CE|CRM Data Source will call the XRMService.

Use "Impersonate User (GUID)" together with a Service-User Username/Password to impersonate a specific user. For impersonation with the currently logged in user, use the input Parameter UserID.

You may also specify Proxy settings, these are used from the dox42 Add-In only, and ignored on the dox42 Server.

4.2. Connection to your D365 CE | CRM with Azure AD App User with Client Secret/Certificate

> **Microsoft has deprecated the WS-Trust authentication type security protocol and hence connecting via Username and Password may not be possible for D365 CE tenants going forward.**

In this case we either recommend you to connect via Azure Active Directory Impersonation, or with an Azure AD Application user with Client Secret and/or Certificate. The later authentication method is needed for certain scenarios, for example when using dox42 with Microsoft Power Automate Flows. **For more information on this set-up, please see chapter 7.**

4.3. Connection to D365 CE | CRM with Azure Active Directory Impersonation

Instead of connecting to your D365 CE or Dynamics CRM instance with your username and password, or with an AAD application user, you can connect with Azure Active Directory Impersonation (see a detailed instruction on how to set up your AAD configuration in chapter 6). **This is the authentication method we recommend for most dox42 Online or Azure Active Directory use cases.**

4.4. Read Data from MS Dynamics CRM or Dynamics 365 for Customer Engagement

Press "Connect" to read the list of available entities from your CRM. Select the desired entity in the dropdown "Entity" and tick the desired fields in the list.

To set a filter just enter a value or choose a data field and choose the desired filter.

Dynamics CRM stores data/time fields in UTC. To transform all date/time fields to local time check "Date/Time Fields in Local Time (instead of UTC)".

To get the CRM Labels as mouse-over tooltip in the dox42 data field explorer check "Use Labels." You need dox42 version 3.6 or higher to use this function.

You may overwrite the labels adding you own comments. Also you can always refresh the labels and reload the fields from CRM by clicking "Refresh Labels and Fields".

Another possibility to select which data you want to read from CRM, is to use the dox42 Fetch XML functionality. Connect to your Dynamics CRM, select your Entity and click the Fetch XML button. You can now simply paste your Fetch XML here.
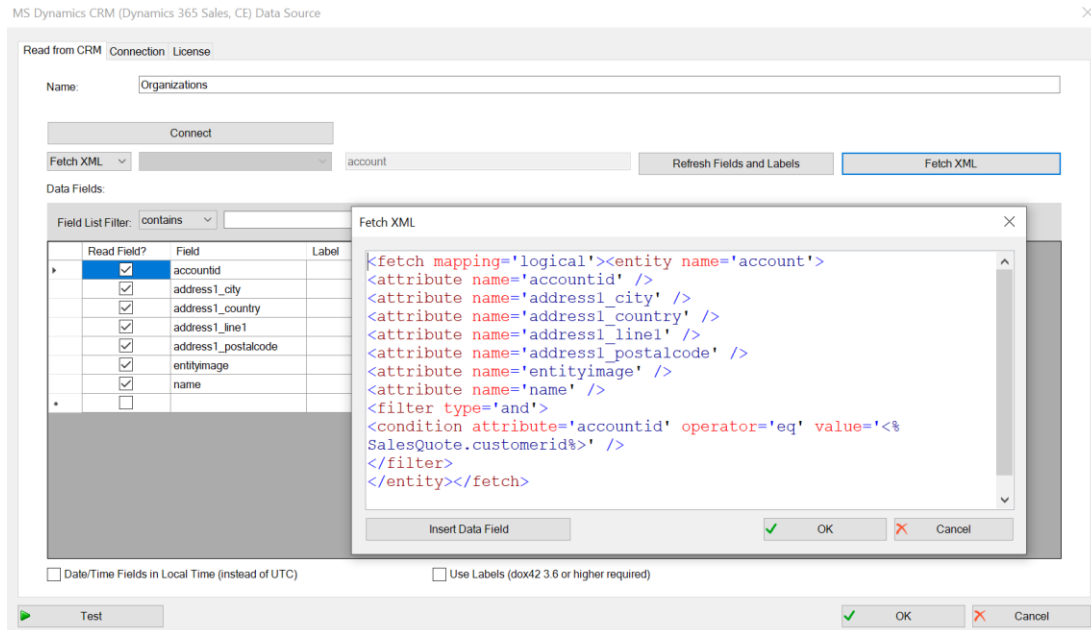
Now you can use data fields from MS Dynamics CRM like any other dox42 data fields:
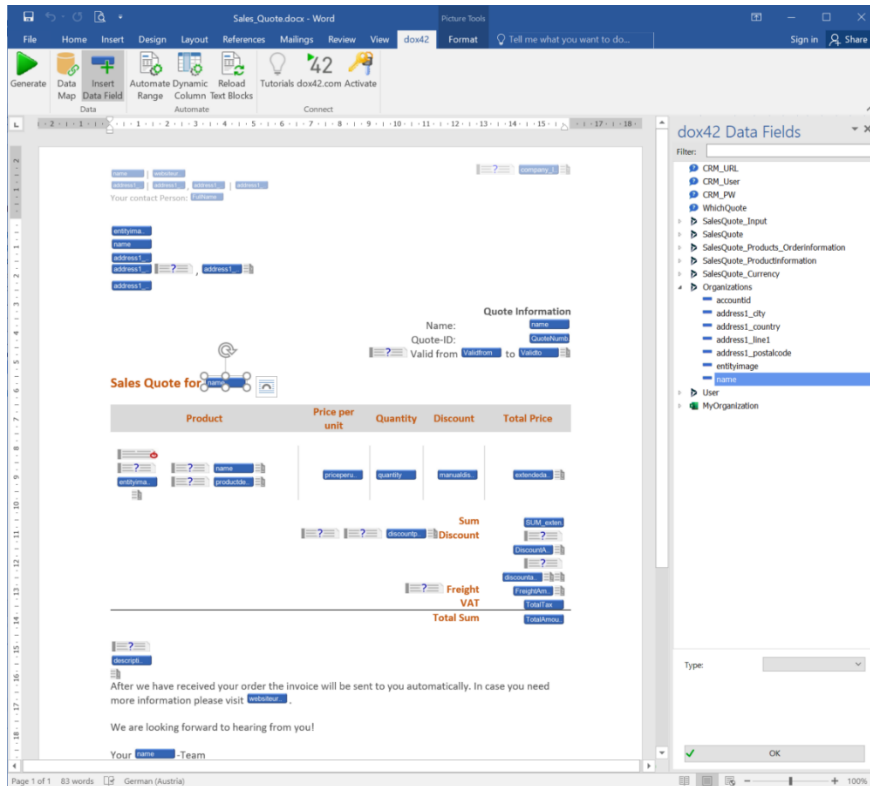
4.5. How to get a full list of your fields and entities

The dox42 D365 CE|CRM Data Source reads the field and entity names from MS Dynamics CRM. As there can be differences between the field name and the display name (e.g. because of language settings) we recommend you to work with the field and entity list given from MS Dynamics CRM. To get this list of field and entity names please do the following (for MS CRM Online, MS CRM 2013 and 2015 on-premises):

1. On the navigation bar click **Microsoft Dynamics CRM > Settings**.

2. Again on the navigation bar click **Settings** > **Customizations** > **Customize the System**.

3. To get the entity name:

   1. In the new window under **Components**, expand **Entities** and from this full list click on the entity you want.

   2. The next view contains the name and the display name of the entity.

4. To get the field name:

   1. In the new window under **Components**, expand **Entities** and from this full list expand the entity you want.

   2. Click **Fields** to get the list which contains names and display names.

5. dox42 D365 CE|CRM Output Action

Using the dox42 D365 CE|CRM Output Action for your dox42 Server calls, you can save generated documents to CRM as an annotation of an entity. E.g. you can attach a generated offer-document as an annotation to the corresponding offer in CRM.

Since version 4.4 you can use the mode "email" to create CRM email activities with the generated document in the attachment. The emails are directly opened in CRM with pre-populated sender, receiver, subject and body and are attached as an annotation to the entity/timeline.

You can use the dox42 D365 CE|CRM Output Action in the same way as any other dox42 output action, which can be found in the dox42 Server documentation. Use the dox42 Server Designer to test your calls. It can be downloaded from www.dox42.com/modules/download.
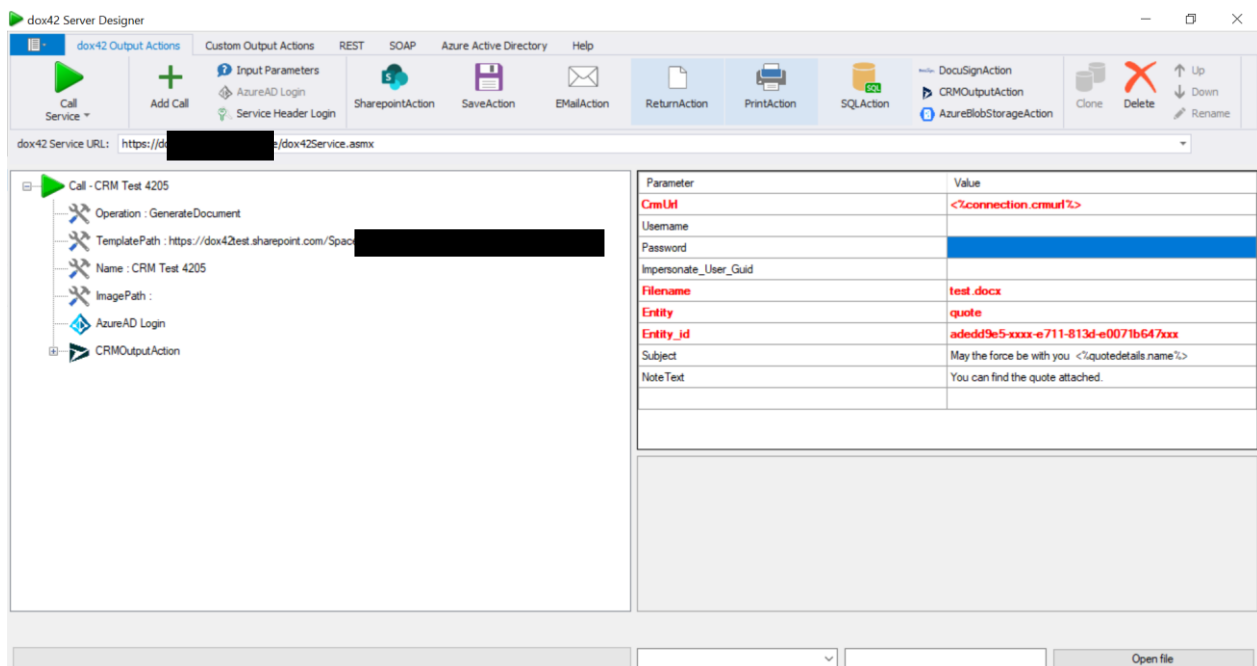
5.1. Parameter

Parameter name	Requirement	Effect
CrmUrl	mandatory	https://yourcompany.crm4.dynamics.com
Username	optional	If you use a MS Dynamics CRM data source with Azure Active Directory in your document, you may omit the credentilas to use AAD authentication.
Password	optional	If you use a MS Dynamics CRM data source with Azure Active Directory in your document, you may omit the credentilas to use AAD authentication.
Impersonate_User_Guid	optional	To impersonate a different user. This is only possible together with Username/Password set.
Filename	mandatory	The filename of the document. The extension will define the format.
Entity	mandatory	The type of the entity you want to attach the document, e.g. account.
Entity_id	mandatory	The id of the entity you want to attach the document. E.g. <%ID%> to pull the ID dynamically from your input parameter called "ID"
Subject	optional	You may use other data fields to create a dynamic subject (=Title), e.g.: "Offer for <%Customer.Name%>".
NoteText	optional	You may use other data fields to create a dynamic Description, e.g.: "Offer for <%Customer.Name%>".
Mode	optional (available since v 4.4.)	email; needed if you want to create a CRM email on the timeline.
CRMEmailSender	optional	The e-mail sender for a CRM email, can be a userid
CRMEmailReceiver	optional	The e-mail sender for a CRM email, can be a userid
CRMEmailSubject	optional	The Subject of the CRM email
CrmEmailBody	optional	The body text of the CRM email

6. Azure Active Directory Integration with D365 CE|CRM

Using dox42 with AAD and/or with **dox42 Online** (Saas), requires the following properties:

- **Application ID >** is provided after the app registration in the Microsoft Admin Center
- **Redirect URLs**
- **Tenant ID**
- **Client Secret for dox42 Server**

For the AAD integration, you have to configure a few settings in the Microsoft *admin center*, in your dox42 Server and in the dox42 Add-In Datamap Designer.

Please note: for the following steps you need (global-) Admin rights:

- App Registration

- Enabling required permissions

- Entering Redirect URL(s)

- Configuring AAD in the dox42 Add-In and **consent the** new application for the first time

6.1. dox42 App Registration

Firstly, you need to create an application in the Microsoft Azure AD admin center to set the AAD permissions for the dox42 Add-in and Server.

- Open the Office 365 Admin Center

- Go to -> **Admin centers** -> **Azure AD,** or click here in order to log in immediately:
  https://aad.portal.azure.com

- Click on **All services -> App registrations**

- **Click on New Registration**
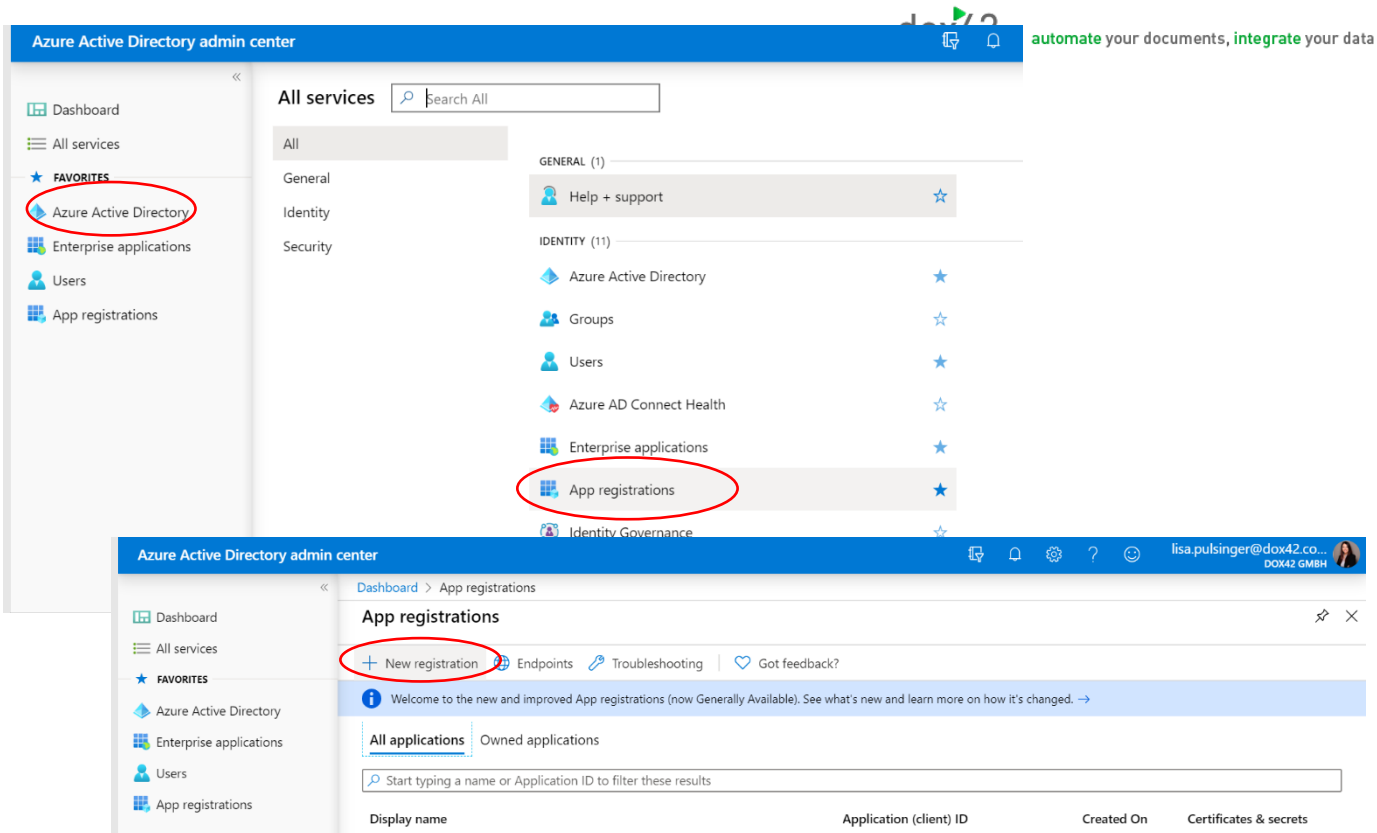
Now configure your dox42 application:

Name your application accordingly, and choose which accounts should have access to your application. Also add a Web Redirect URI containing the link to your dox42 Server instance and click on Register.
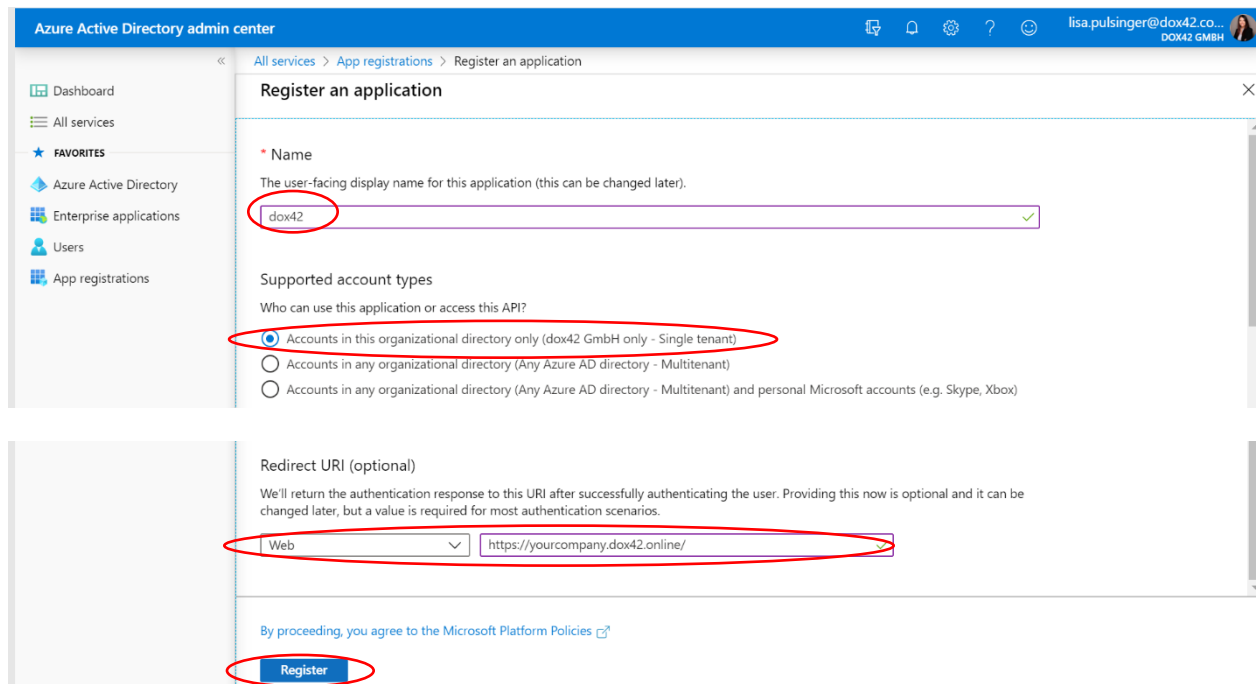
> **Note: In this documentation, we use the redirect URL of a dox42 Online server (SaaS):**
>
> https://yourcompany.dox42.online/
>
> If you are registering a dox42 **on-premise server**, use the URL of your dox42 server:
> https://yourdox42server.yourcompany.com

Now that you have registered your application, you can copy the Application (client ID) and Directory (tenant ID) (screenshot below 1) and paste it into an Excel Config file. This allows you to manage your configuration details centralized. See Chapter 6.1.6 for more information on Config files.
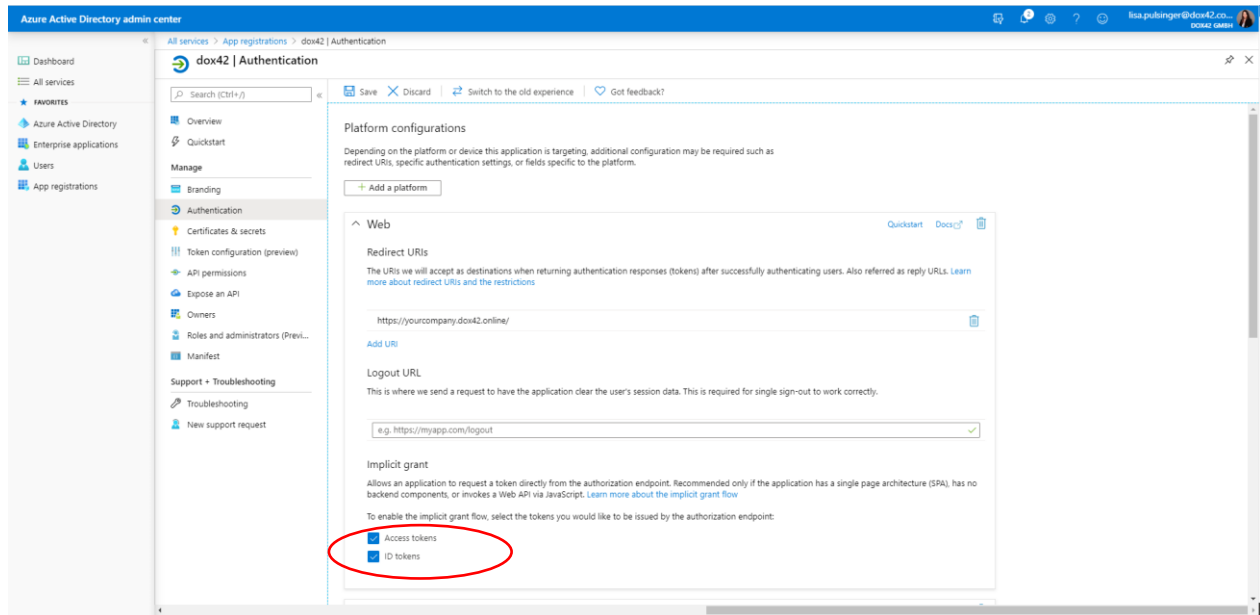
6.1.1. Token Permissions

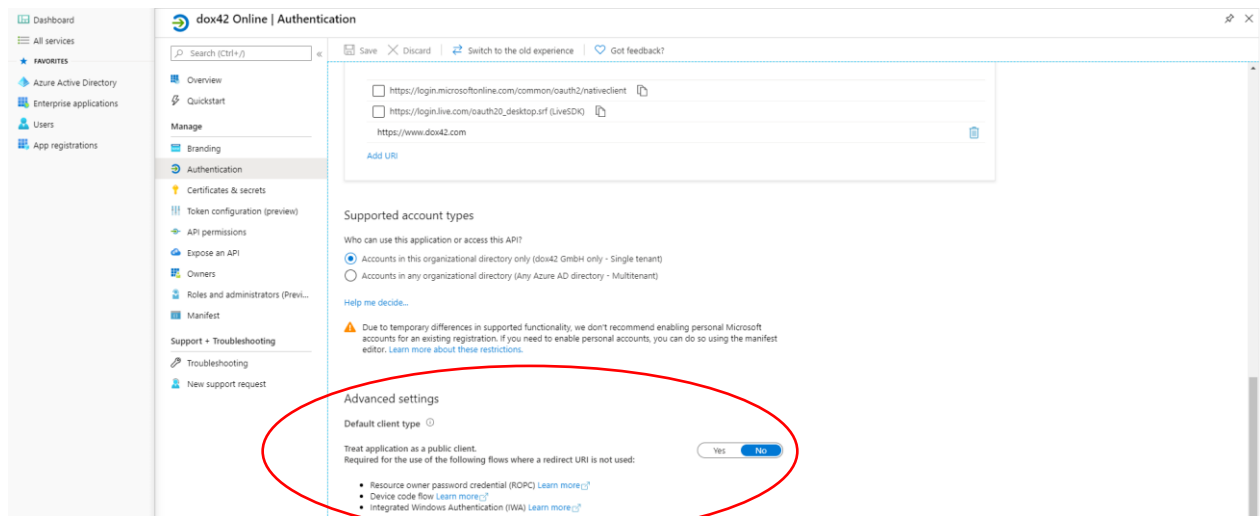Next, you need to grant token permissions for your app.

Within your newly added application, click on Redirect URIs (2).
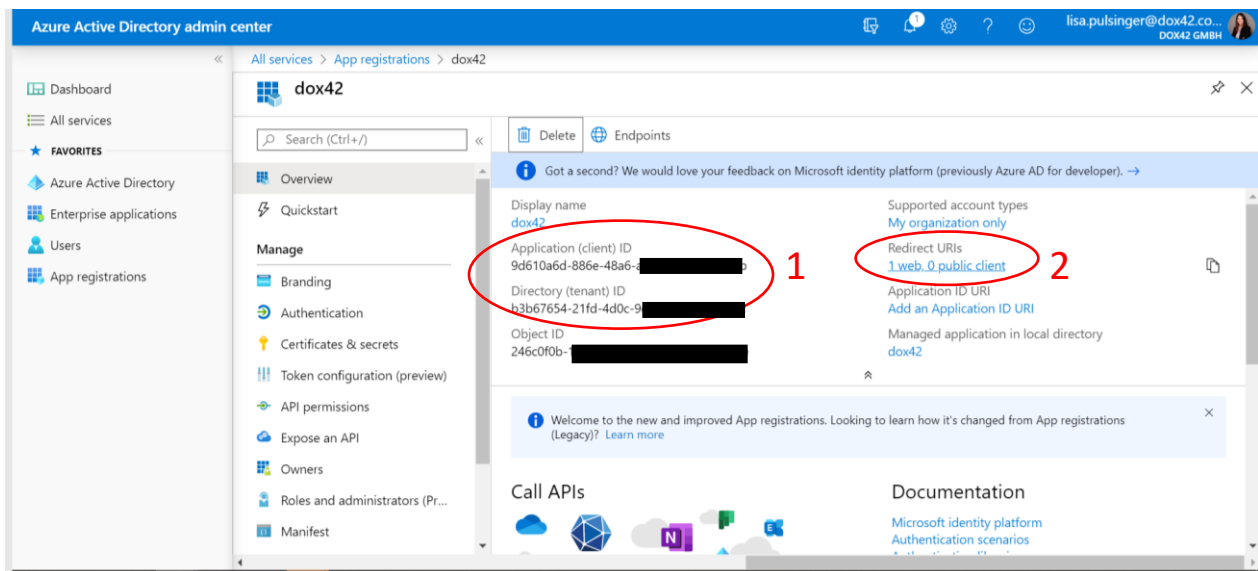
Now, on the In the Implicit grants section of the Authentication register, tick both "Access tokens" and "ID tokens":

Scroll down further to find additional settings. Under Advanced Settings keep "Treat application as a public client" set on No.

6.1.2. Enter Redirect URIs

Now scroll up again to the beginning of your Authentication page. You need to add Redirect URIs next. Please specify the following:

- your dox42 online Server URL:
  https://yourcompany.dox42.online/

- your Dynamics 365 CE URL in the following format: https://Yourcompany.crm4.dynamics.com/

- any additional URIs for services you want to call. Enter this re-direct URI if you are calling dox42 from MS PowerAutomate (MS Flow): https://global.consent.azure-apim.net/redirect

The URL that you include as the Server URI (http://.....dox42.online/) is the one you will need to include in the dox42 Server Configuration in the dox42 Datamap (see Chapter 6.4).

Note: You may NOT use wildcards (*). Microsoft changed this behavior during 2018, so you may find older working App registrations using wildcards, but new App registrations will not work with wildcards.
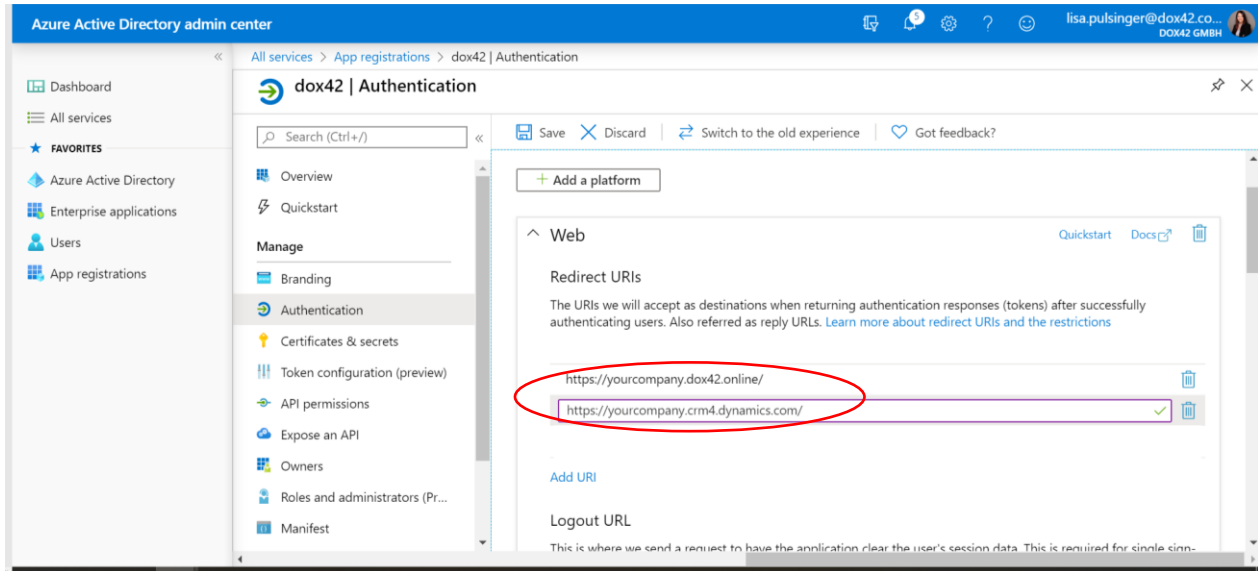

6.1.3. Add a mobile and desktop application for your dox42 Add-in

Now you need to add your dox42 Add-in as a "mobile and desktop application" to your app. On the top of the Authentication page, click on "Add a platform" and then on "Mobile and desktop application".
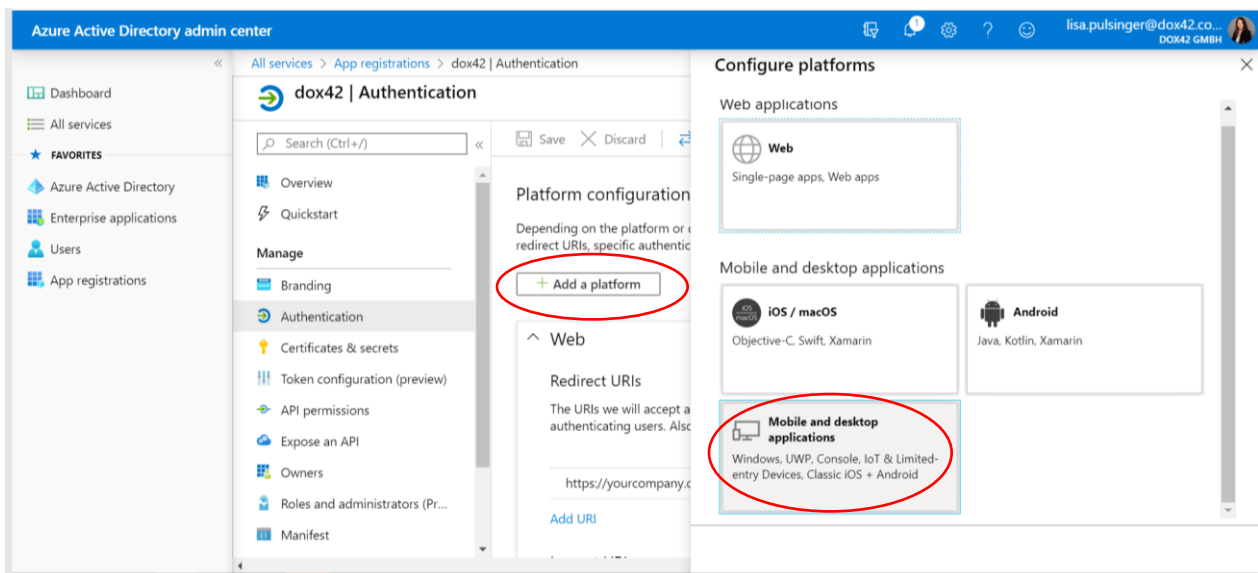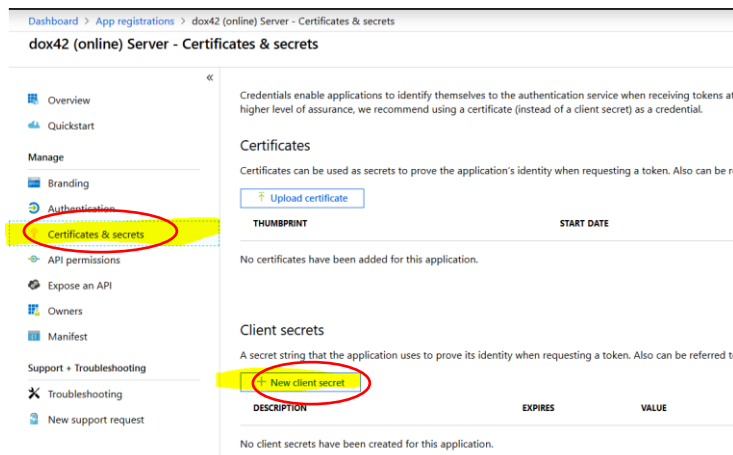
Add a Custom redirect URI for your dox42 Add-in – we are using https://www.dox42.com. Do not use the same redirect URI as for your web application. Then click on Configure.
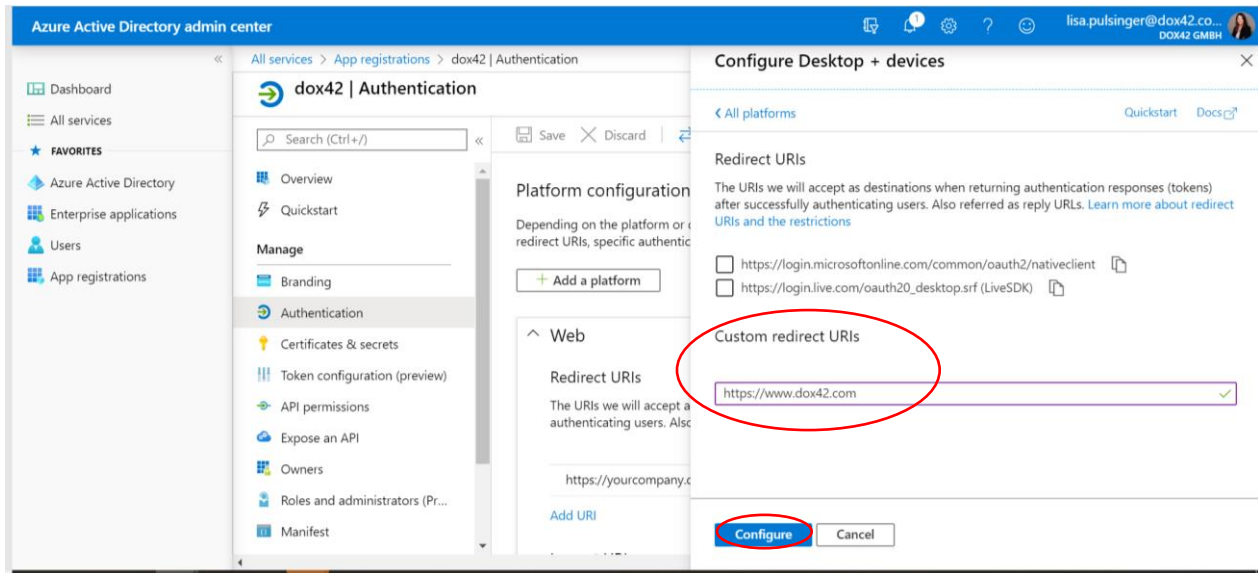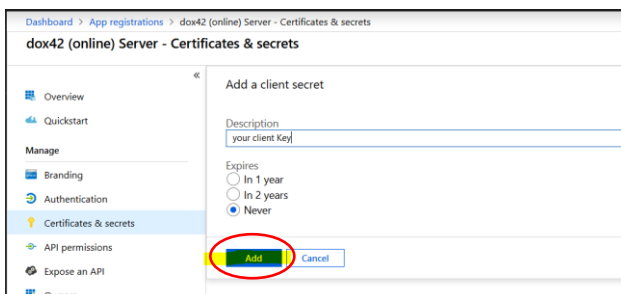

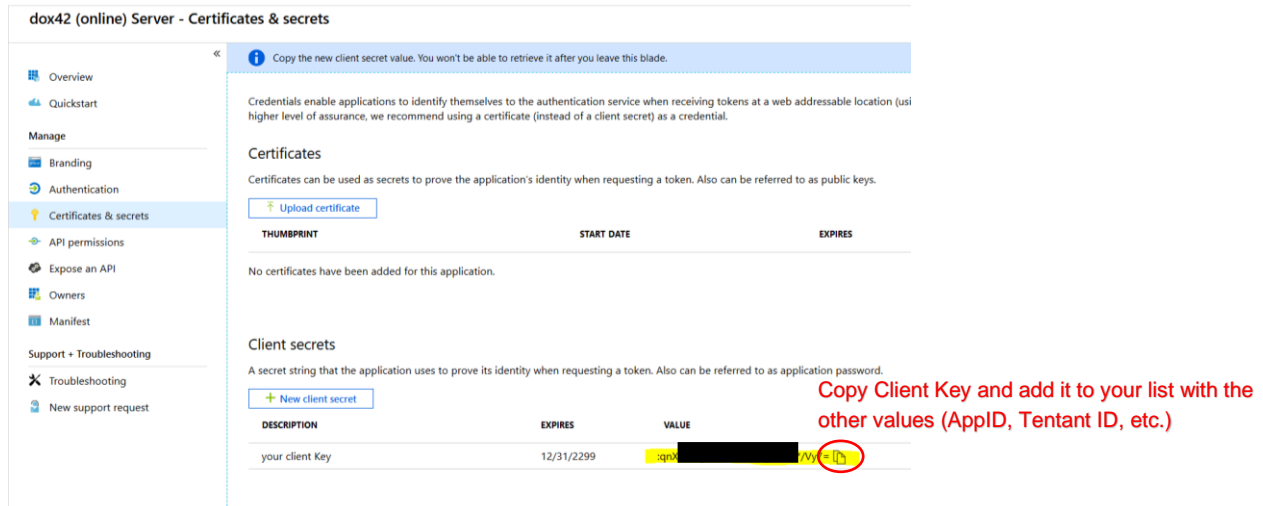### 6.1.4. Add a Client Secret

In the register Certificates and Secrets, you need to add a new client secret.

Select your required expiry date (we recommend using "Never") and click Add.

Make sure to COPY the ClientKey, as it will be hidden afterwards and you will <u>not</u> be able to retrieve it again:

We copied the key into our Excel Config File, together with the other values of the app registration. When working with an Excel config file, we recommend to encrypt the values of the client key by using the dox42 Crypto Data source.

6.1.5.   Add API Permissions

Move to the register API Permissions and click "Add a permission". You need to grant the application delegated permissions for Dynamics CRM and SharePoint Online (as your templates need to be stored on SharePoint Online, and you may be using dox42 SharePoint Output Actions).
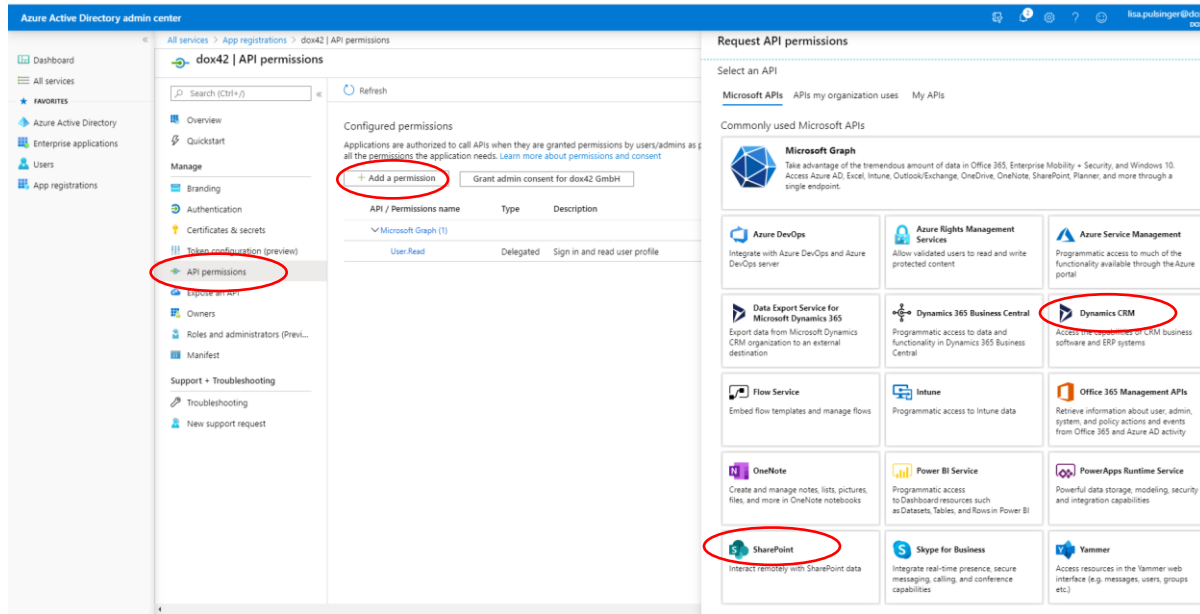
For Dynamics CRM you need to add Delegated user impersonation permissions.
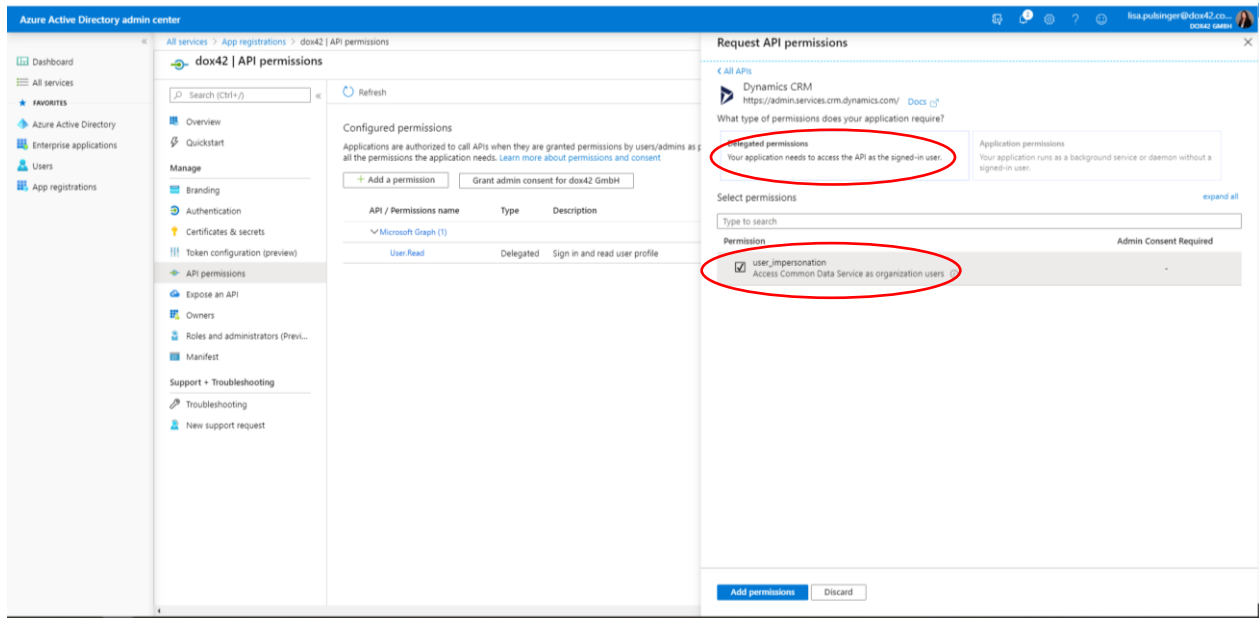
Please select the permissions according to your requirements and systems you are using dox42 with.

The dox42 server requires SharePoint read rights and optional write rights, in case you want to store your generated documents on SharePoint, including meta data. Please grant the following delegated permissions: AllSites.Read, AllSits.Write
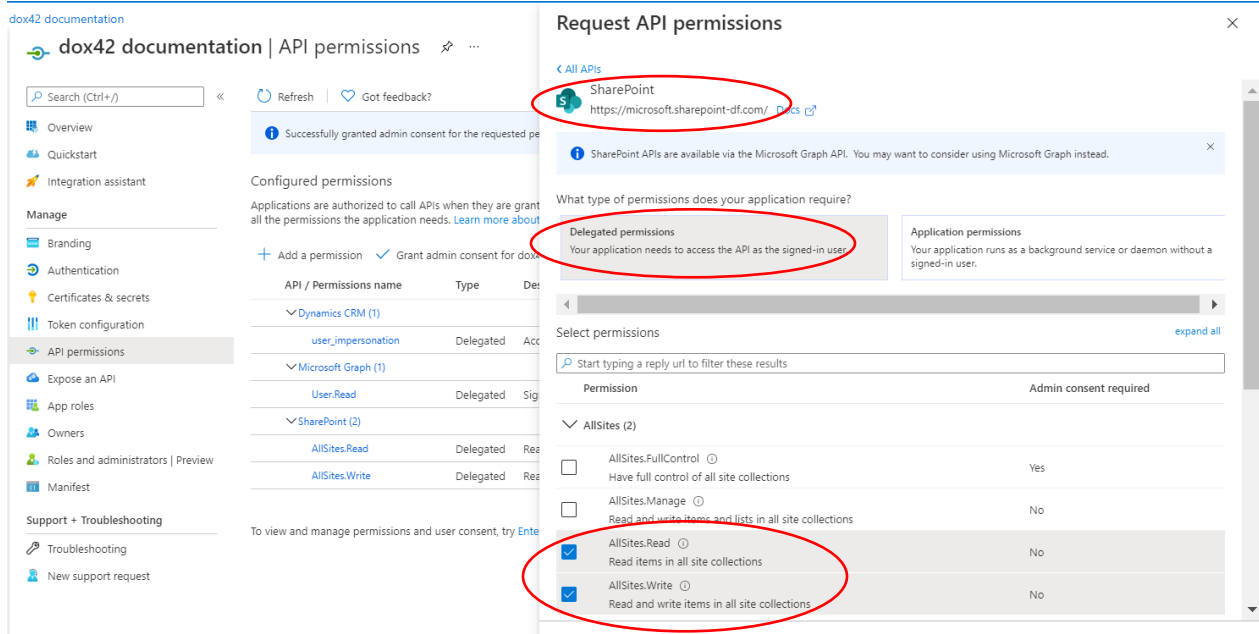
Next click on "Grant Admin Consent" for Your Company, to give your application the necessary permissions.
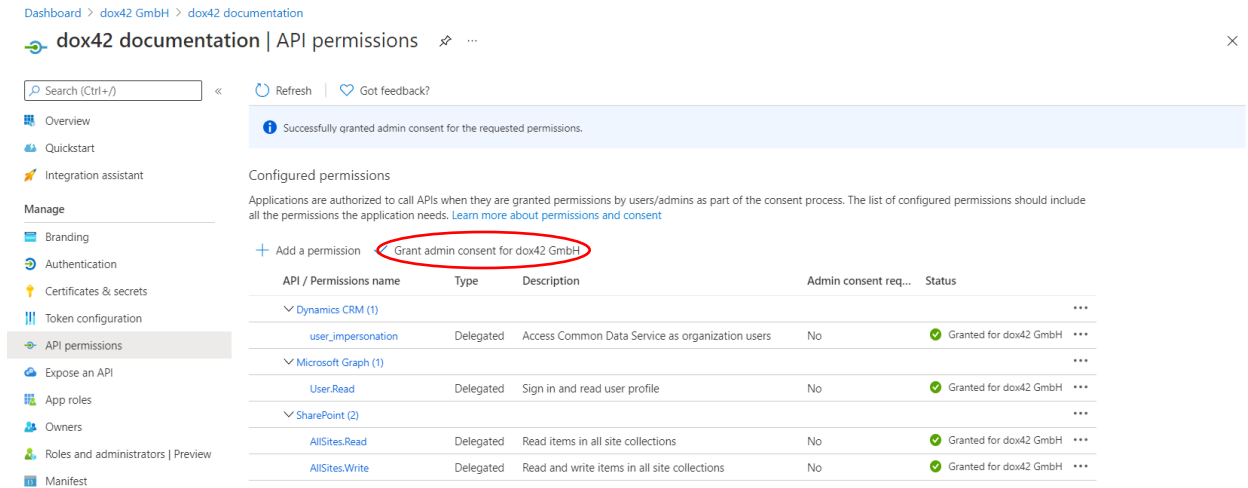
Your app registration is complete now, next you need to configure your dox42 Server accordingly.

6.1.6. Excel Config File

We recommend using this file to integrate all the D365 CE | CRM connection information dynamically into your dox42 data sources. That saves a lot of time during the template design process and avoids copy & pasting, as you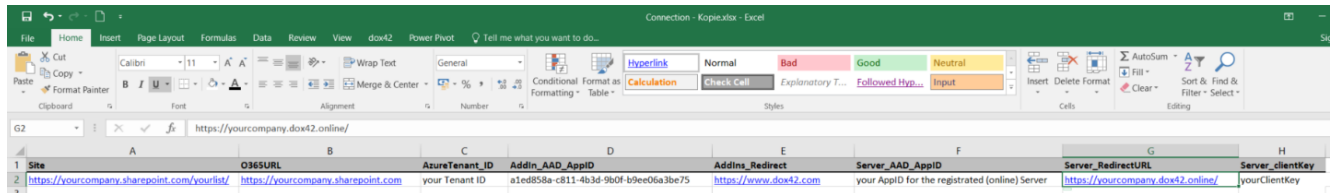 can store your credentials centrally. Please make sure to encrypt your Client Secret with the dox42 Crypto Data Source, before storing it there. You can find a sample of a config file in the CRM_ProductList_JSON_sampletemplate folder of your download package.

How to set up a connection file and how to use the crypto data source is explained in this tutorial video: https://www.dox42.com/Resources?search=/first%20dox42%20document/i&

6.2. Further information AAD App Registration

https://blogs.msmvps.com/windsor/2017/03/12/walkthrough-building-a-custom-web-api-for-use-with-sharepoint-online/

6.3. dox42 Online Configuration

Once your set-up for your Azure AD app is finalized, you need to complete the configuration in your dox42 Server. If you work with dox42 Online, follow this chapter. Otherwise configure your web.config as shown in chapter 6.5.

Please log into your dox42 Online management UI with the username and password received from dox42. The URL for your management UI looks similar to this: https://yourcompany.dox42.online/MAUI
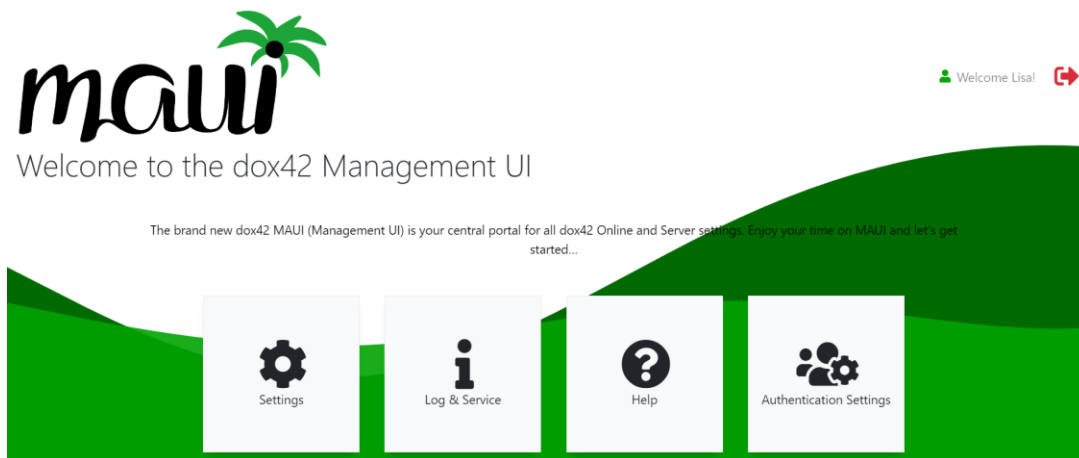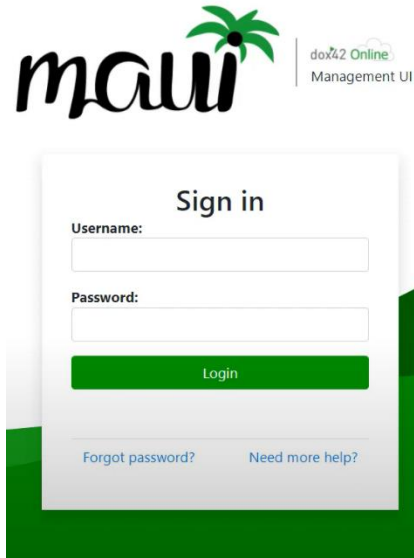
In this dox42 Class tutorial video, we give a detailed explanation of the dox42 Online Management UI (MAUI): https://www.dox42.com/Resources?search=/maui/i&

6.3.1. Add your Azure AD App registration settings

1. To complete the dox42 Online set-up, click on the Settings page on your MAUI home screen.

2. Then click on "Add" in the "SharePoint Online Sites for dox42 Templates" section and enter your SharePoint Online sites, where you call your dox42 templates from. Please make sure to add a Slash (/) in the end of the site URL. If you call templates from various SharePoint sites, please add all of them. E.g. https://yourtenant.sharepoint.com/yoursite/

   Each SharePoint site needs to be added with the App ID, Tenant ID and the generated Client Secret value of the dox42 App registered in the Azure AD Admin Center (see chapter 6). **Don't forget to click on Save & Deploy after each site you add to the list.**

3. Configure your trusted template locations. Click on "Add" in the "Trusted Template Locations" sections of your settings page. Your templates have to be stored on SharePoint Online, hence your trusted template location could be https://yourcompany.sharepoint.com/, or you could also specify a specific SharePoint site. **Don't forget to click on Save & Deploy after every Trusted Template Location you add.**

4. Upload your Aspose (dox42 OEM Developer) license (not applicable for trial installations)

5. If you want to send emails via dox42 online, also add the email server, address and password at the top section of the page. Again, **don't forget to click on Save & Deploy after you've added the email configuration.**

6.4. Configure Azure AD in dox42 Data Map Designer

Now, to connect to D365 CE | Dynamics CRM | Dataverse datasources, you need to enter the connection information in the dox42 Add-In Data Map Designer

6.4.1. Option 1: Using an Excel config file filling in the information dynamically

After connecting the Excel config file, (see tutorial "My first dox42 document") select the D365 CE|CRM data source within the Data Map Designer (1). Then fill in your D365 CE | Dynamics CRM URL (2), Click on "Dynamics 365 Azure Active Directory Impersonation" (3) and fill in the fields by clicking on "…" (4) and select them from your Excel data source as shown below (5).

After you have selected your values, click on Consent for both the Add-in and Server (6). **Azure global admin rights are required**. **This step can be skipped, if you clicked on "grant permissions" within the AAD admin center already (see Chatper 6.1.5)**.
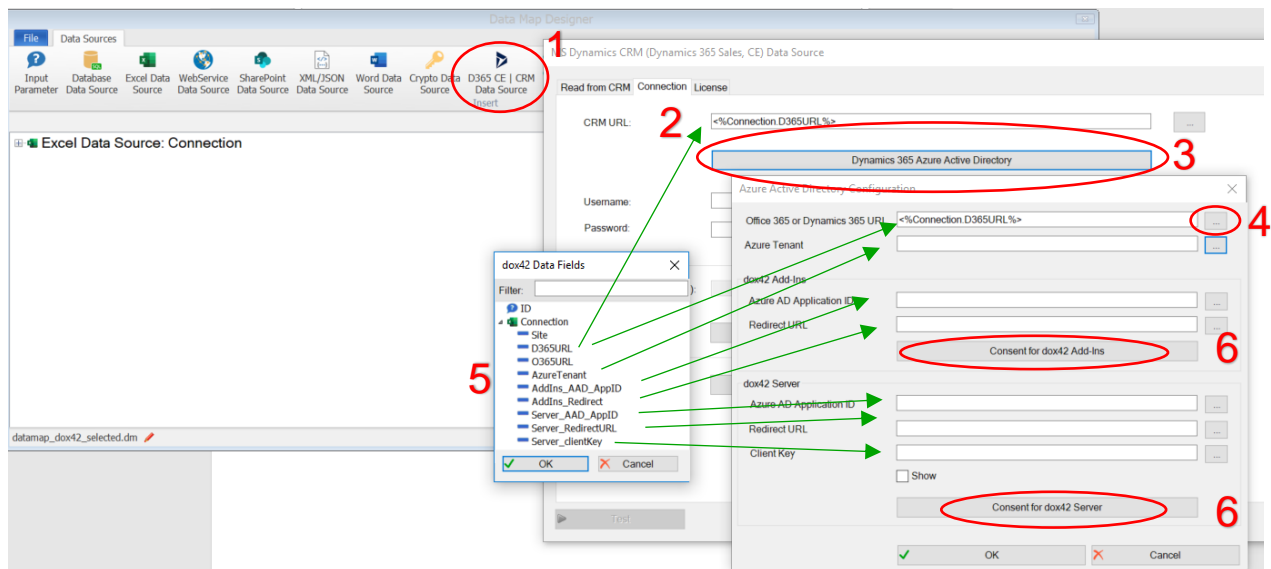
6.4.2. Option 2: Using Copy Paste

After step 2 on Option 1, you also can copy the information directly into the AAD Configuration without using the Excel Config file:
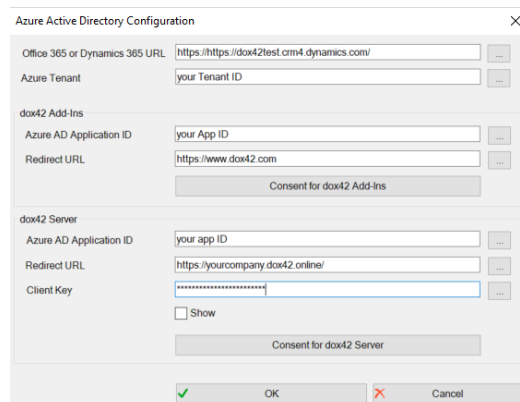
We recommend to encrypt your Client Key with the dox42 Crypto Data source! (see tutorial "My first dox42 document")

After completing the fields, don't forget to click on Consent for both the Add-in and Server. **Dynamics admin rights are required**. **This step can be skipped, if you clicked on "grant permissions" within the AAD admin center already (see Chatper 6.1.5)**.

6.4.3. Option 3: Using Copy Paste

Available from dox42 Add-in Version 4.4.1.3. you can also import your connection information via an XML file.

6.5. Configure Azure AD Access for dox42 On-Premise Server in Web.config

This chapter only applies to dox42 on-premise customers. If you use AAD with a dox42 on-premise Server, you need to adjust the web.config of your Server.

ConfigSection:

<configSections>

. . .

  <section name ="azureAD" type ="dox42.Core.Utils.AzureADSectionHandler,dox42Core"/>
</configSections>

Azure AD Section:

```
<azureAD>

    <add     resource365="https://mytenant.sharepoint.com"
             appID="…"
             tenant="…"
             clientKey="********************************" />
</azureAD>
```

7. Connection to D365 CE | Dynamics CRM with AAD Application User & Client Secret/Certificate

Microsoft has deprecated the WS-Trust authentication type security protocol and hence a connection to D365 CE with Username/Password, may not be possible going forward.

We recommend to work with Azure AD impersonation for your D365 CE connection (see chapter 6), however in some scenarios, for example when using Microsoft Power Automate or PowerApps, the connection via an AAD application user, as well as a Client Secret or Certificate is needed.

Please note that for your dox42 Online or Server On-Premise calls with Azure Active Directory, you will still need to follow the instructions in chapter 6. This chapter only refers on how to connect with an Azure Active Directory App Registration user with Client Secret or Certificate within the dox42 template.

To keep the process simple, we've created a second application within Azure Active Directory, you could however also combine the app you registered in chapter 6 with the one described below.

7.1. Register an Application in Azure Active Directory

First, register a new application in your Azure Active Directory Admin Center (https://aad.portal.azure.com/)

Go to **App Registrations** and Click on **New Registration**.

Enter a Name for your Application and Click on Register. For this Authentication method, no Redirect URI is needed.

Register an application ...

* Name

The user-facing display name for this application (this can be changed later).

dox42 CRM Application Certificate/Client Secret

Supported account types

Who can use this application or access this API?

◉ Accounts in this organizational directory only (dox42 GmbH only - Single tenant)

◯ Accounts in any organizational directory (Any Azure AD directory - Multitenant)

◯ Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

◯ Personal Microsoft accounts only

Help me choose...

By proceeding, you agree to the Microsoft Platform Policies ⤤

Register

Once you've registered the application, your app will look like this:

Copy your **Application ID and Tenant ID** from here and paste it in an Excel or XML Config file. Working with a config file will allow you to manage your CRM credentials centrally. See Chapter 6.1.6 Excel Config File for more information.

Next, you need to upload your Certificate and/or add your Client secrets. Change to the menu option "**Certificates & secrets**" and do so accordingly.

Please note that Azure Active Directory only allows certificates to be uploaded in .cer, .pem or .crt format, which are not password protected formats. For the import into the dox42 D365 CE | Dynamics CRM data source your certificate needs to be in .pfx format and hence, password protected! Please make sure to export your certificate in the right formats within your certmgr.

If you're working with a Client Secret it is important **that you copy the Value (not the ID) immediately.** Please paste the client secret to the Excel Config file too and encrypt it (watch our tutorial video "my first dox42 document" to learn how to use the dox42 Crypto data source). **Also, copy the value immediately after you've added the client secret, as later on you will not be able to retrieve it again.**

Next, you need to add API permissions to your application.

Change to the **API Permissions Tab** – click on **Add a Permission** and **add delegated user impersonation rights for Dynamics CRM.**
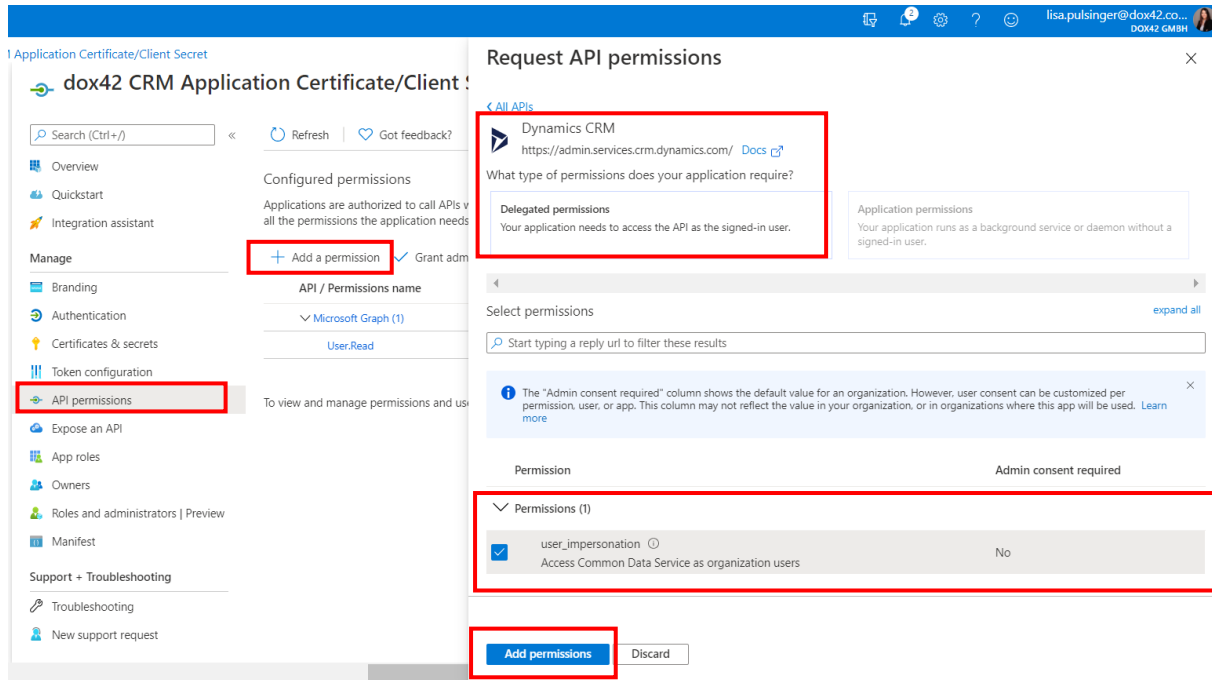
Now grant the admin consent for the delegated Dynamics CRM user_impersonation, by clicking "Grant admin consent for dox42 GmbH":
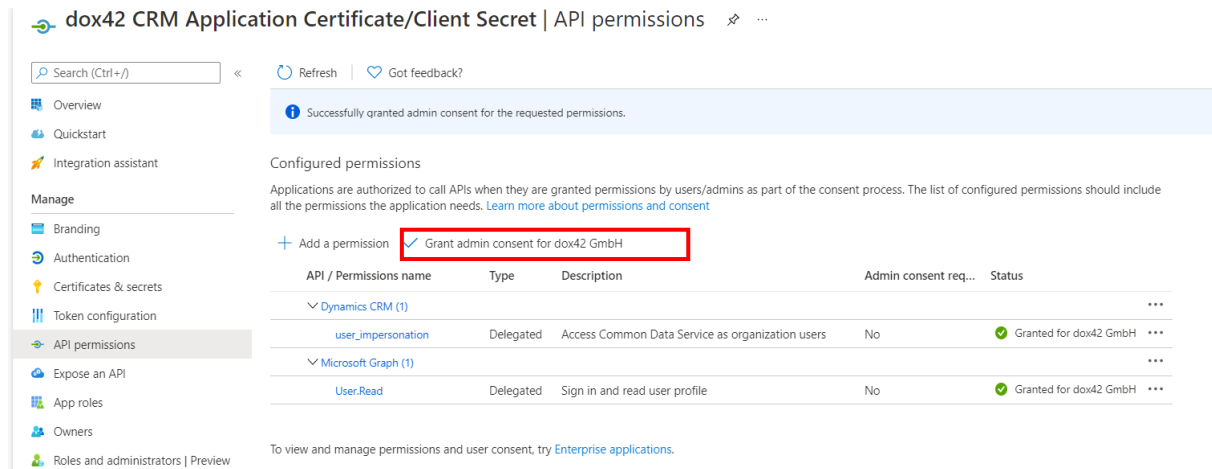
7.2. Add an application user to D365 CE | Dynamics CRM

Next you need to add an application user for your newly registered AAD application in your D365 CE | Dynamics CRM System.

When working with the Power Platform admin center, please follow the steps documented in this article: https://docs.microsoft.com/en-us/power-platform/admin/manage-application-users.

Do not forget to add System administrator rights to the newly added Application user.

In the classic Dynamics CRM settings area, you can follow the screenshots below:

Go to your **D365 CE | Dynamics CRM** and change to **Settings > Security > Users**. In the User View Dropdown select **"Application Users"** and click **New**.
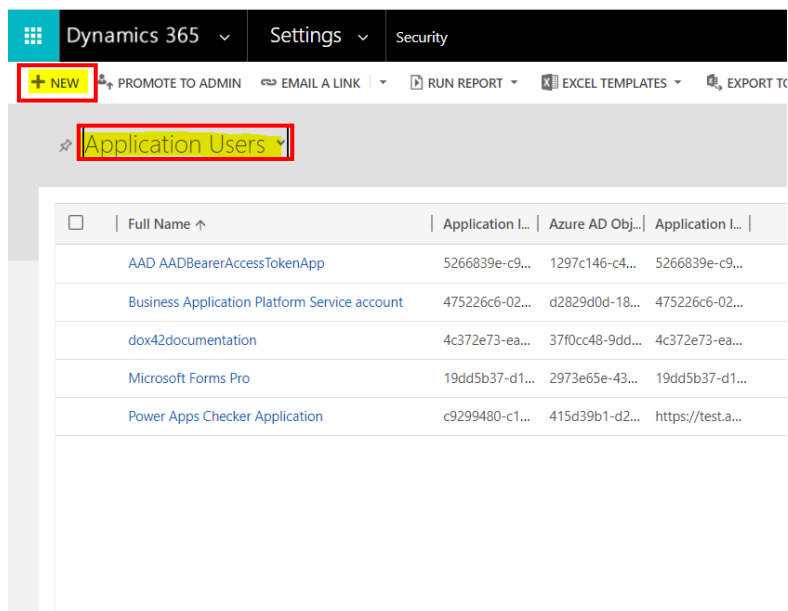
For your new application user, enter the **Application ID of the app you registered** beforehand in Azure Active Directory and click **Save.** The rest of the information will be populated automatically and your application user gets registered. Make sure that the status of the application user is set to **"Enabled".**
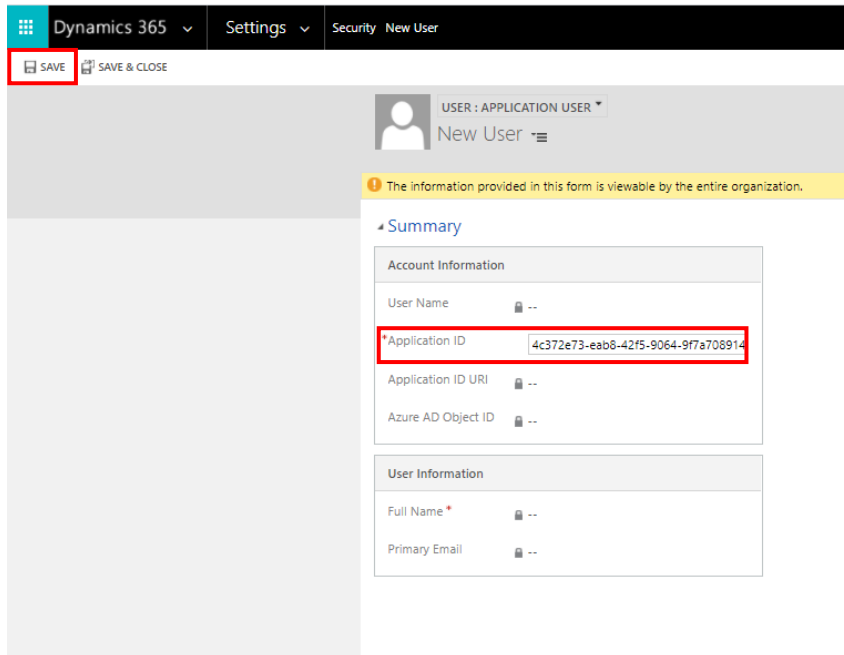
After your Click on SAVE the application will look like this:
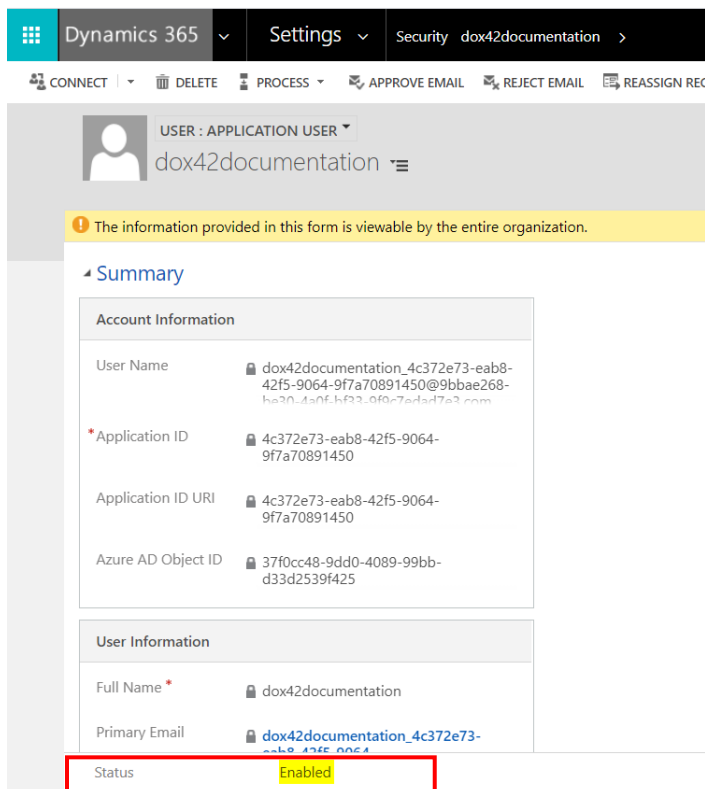
Next, you need to assign **the System Admin role to your Application User**. It needs to have sufficient Read rights to read the data, but also Write rights in case you want to attach your generated documents to an entity. For that click on "**Manage Roles**" Select the **System Admin** and Click "**Ok**".
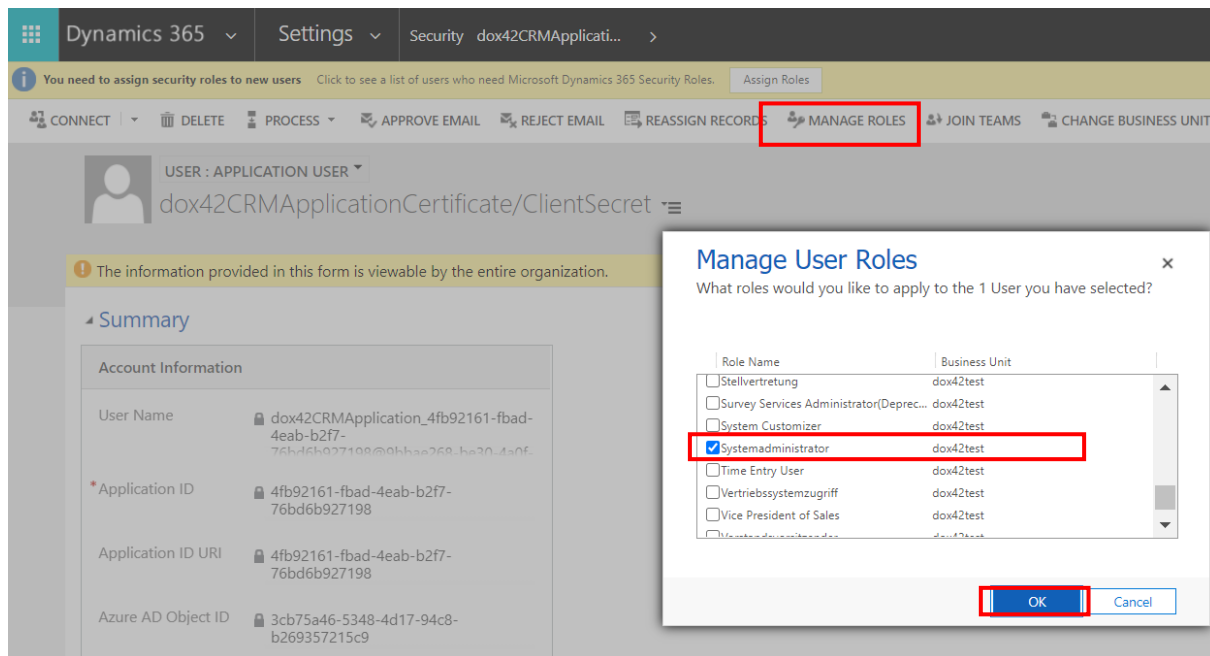
Now your set-up is complete and you can connect to your D365 CE | Dynamics CRM within your dox42 Add-in.

7.3. Connect to D365 CE|CRM in dox42 Add-in w. AAD App, Certificate/Client Secret

Open the Data Map Designer of your dox42 Add-in in Word/Excel/PowerPoint, click on **D365 CE | CRM data source and navigate to the Connection tab**.

On the connection screen choose **"AzureAD App Registration with Client Secret/Certificate"** (1) as Authentication method. Now enter **your Dynamics URL, your AAD App ID and Tenant ID (2). Enter your Client Secret (3) OR import your Certificate which you uploaded to your AAD app registration beforehand (3).**

When using a certificate, make sure to import a certificate in .pfx format, and enter the certificate password.

We recommend to enter your connection data dynamically via your Excel config file, rather than copy/pasting the absolute values here. *See Chapter 6.4.1 Option 1: Using an Excel config file filling in the information dynamically* for further information (4).

Next click on **"Connect"** (5) and continue in the "Read from CRM" tab to select your entity and data you want to read.
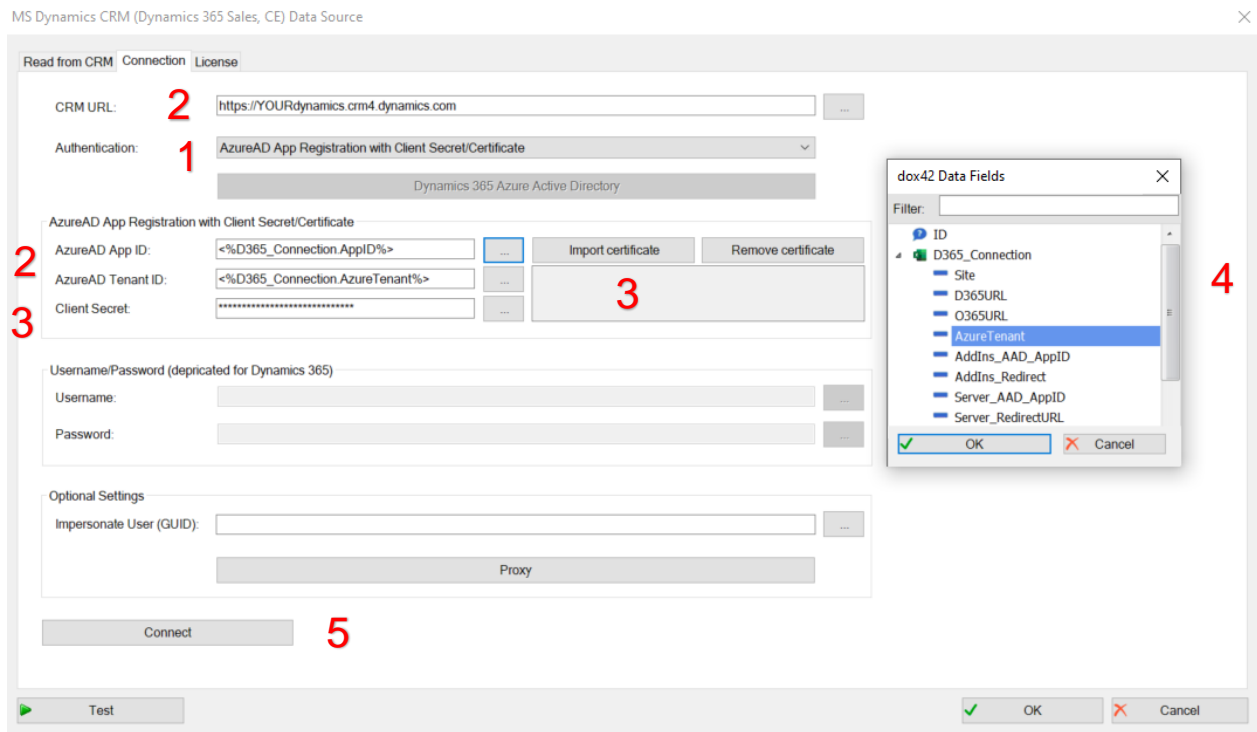
8. Generating D365 CE | Dynamics CRM Documents from Microsoft Power Automate Flows and Power Apps

To use dox42 with a Power Automate Flow or from Power Apps, you can work with the dox42 Power Automate Premium Connector or alternatively with a custom connector in Microsoft Power Automate. Please follow our PowerAutomate Documentation or our dox42 Class Tutorial Webinar for further details on the set-up.

Make sure that the D365 CE data sources in the templates you use within a PowerAutomate Flow scenario, connect via AzureAD App Registration with Client Secret/Certificate or Username/Password. This is needed to obtain the necessary tokens from Azure Active Directory. Follow Chapter 7 for more information.

9. Integrating dox42 into your D365 CE or Dynamics CRM (via dox42 Solution)

dox42 has developed a solution for Dynamics CRM (D365 CE) that allows for an easy integration of your dox42 calls into your CRM interface. From dox42 Solution Version 1_5 upwards there are two options on how to display your dox42 calls. The first one is to add one button for each dox42 template configuration (see chapter 9.6), the second option it to add a pop-up button that allows you to select your templates from a pop-up window and to include further input at run-time, e.g. to dynamically choose the file format (see chapter 9.8).

Option 1:
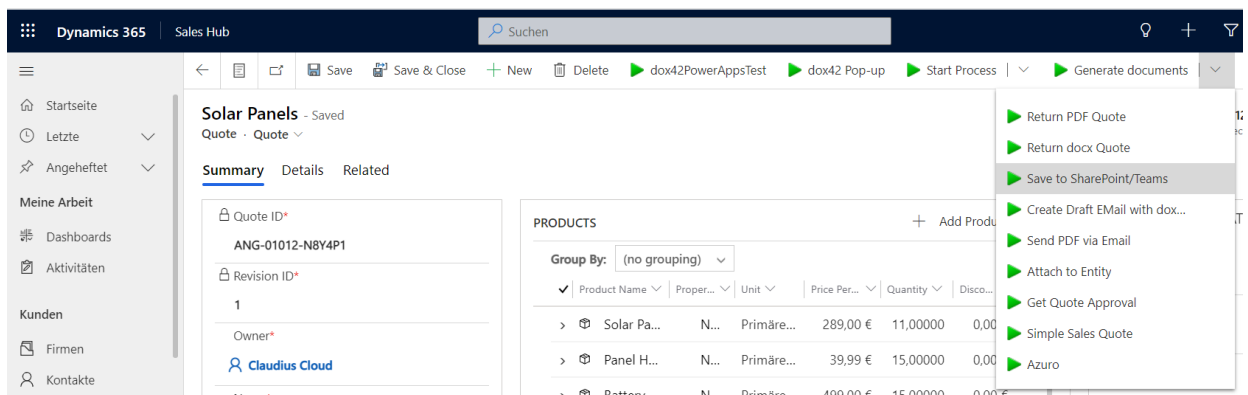
Option 2:
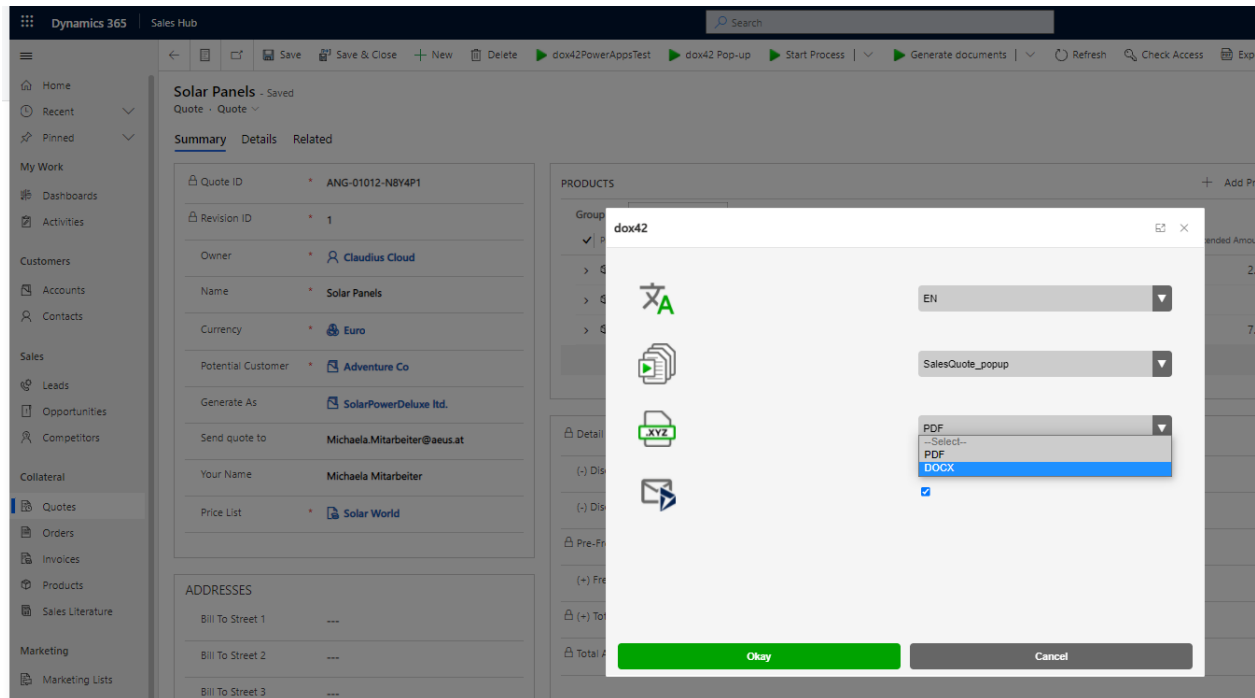
9.1. Input Parameter for your dox42 templates

> The dox42 solution has specified **"Id" as standard name for the input parameter**, and pulls the GUID of your selected entity. When setting up templates, **name your input parameter accordingly**.
>
> If you generate a document from a **list selection with more Ids or when using the dox42 popup to generate your documents and you would like to include input at runtime, e.g. a dynamic file format, specify "RequestBody"** as input parameter name in your template.

Per default, the dox42 solution pulls the user Id of the logged in user and the CRM environment as additional input parameters. You can use them to impersonate the CRM user in your data sources (see 4.1) and dynamically select the CRM environment you are connecting with. You can change this configuration in the dox42 client libraries of your dox42 solution.

Furthermore, in the "RequestBody" input parameter you can include any values from your dox42 pop-up in JSON format. How to easily configure your RequestBody data sources, is explained in detail chapter 9.8.2.

9.2. Import the dox42 Solution

The dox42 solution is available as managed or unmanaged solution. Choose the version you want to work with and import the dox42 Solution (Zip Folder dox42_1_Versionnumber.zip) into your D365 CE or Dynamics CRM:

Choose the dox42 zip file and click next.

Solution Information

⚠ Changes applied by importing an unmanaged solution cannot be uninstalled. Do not install this solution if you want to roll back these changes.

Solution Information

Name:	dox42 Client Libraries
Publisher:	dox42 GmbH(dox42)
Package Type:	Unmanaged

View solution package details

ℹ By enabling this command, you consent to share your data with an external system. Data imported from external systems into Microsoft Dynamics 365 are subject to our privacy statement that can be accessed here. Please consult the feature technical documentation for more information.

Back   Import   Cancel

Click Import to import the solution.

## Importing Solution

ℹ The import of solution: dox42 Client Libraries completed successfully.

Date Time ↓	Type	Display Name...	Name	Status	[
19:31:05.12	Dependencie...			📝	
19:30:43.62	Security Role	dox42 User	dox42 User	📝	Se
19:30:42.90	Security Role	dox42 Templ...	dox42 Templ...	📝	Se
19:30:42.27	Security Role	dox42 Global...	dox42 Global...	📝	Se
19:30:38.68	Chart		dox42_servic...	📝	En

◄◄ ◄   Page 1   ►

Download Log File   Publish All Customizations   Close

After the import is finished click "Publish All Customizations".
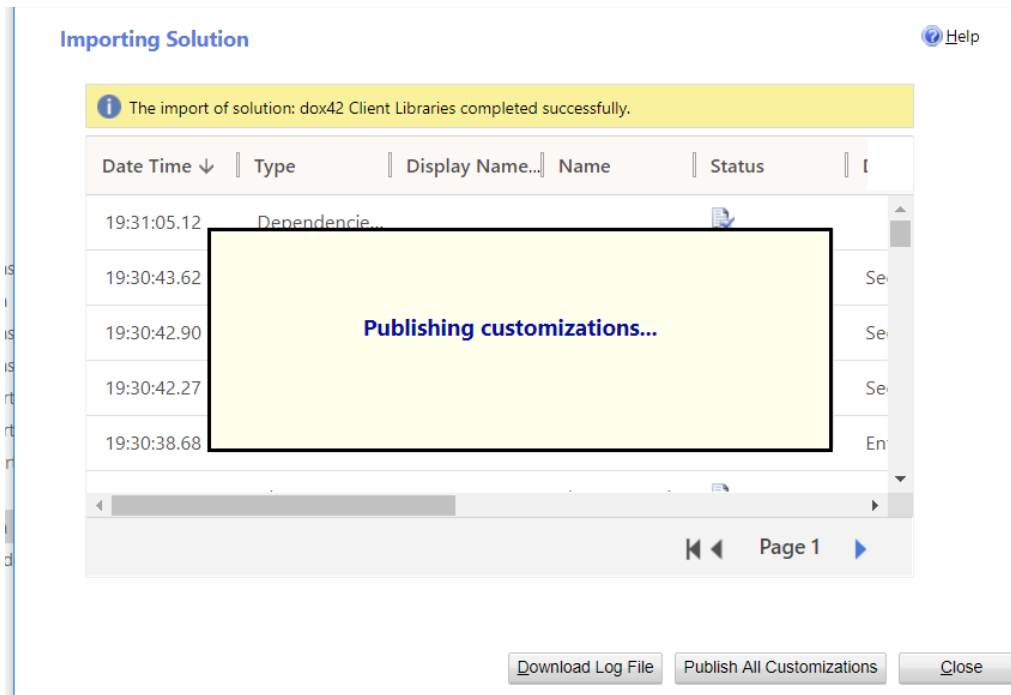
After the publishing has finished (the overlay "Publishing customizations…" disappears), click close to close the import wizard.

If your system is in English, your entities "dox42 Service Connection" and "dox42 Template Connection" will automatically appear in the Settings area.

If your system is in German (or any other language), please click on your entities "dox42 Service Connection" and "dox42 Template Connection" and select in which areas those entities should appear (e.g. in the Settings menu). (see German screenshot below).
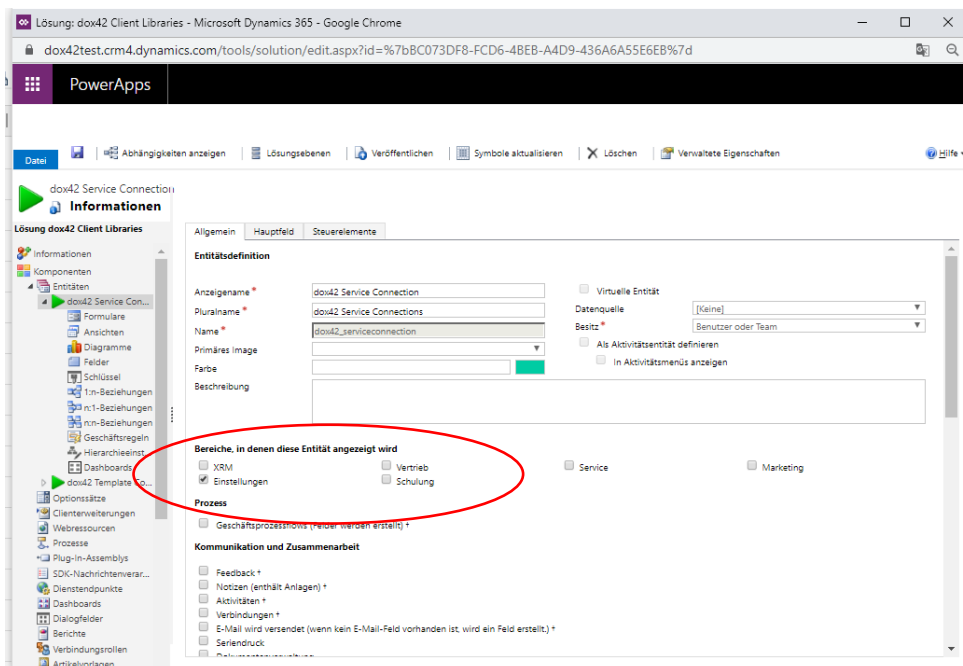
If you are importing the dox42 Dynamics CRM | D365 CE solution as a managed solution and the entities are not shown in the settings area, you need to enable them via Settings > Customize the System > Entities. Follow Resolution 2 described in this document.

Solution version 1_6 and above includes a **dox42 model driven app**, with which you can access the dox42 Service Connection and Template Configuration via your Dynamics Apps. **We recommend to use the model driven app, when working with the dox42 pop up integration (option 2 described above).**
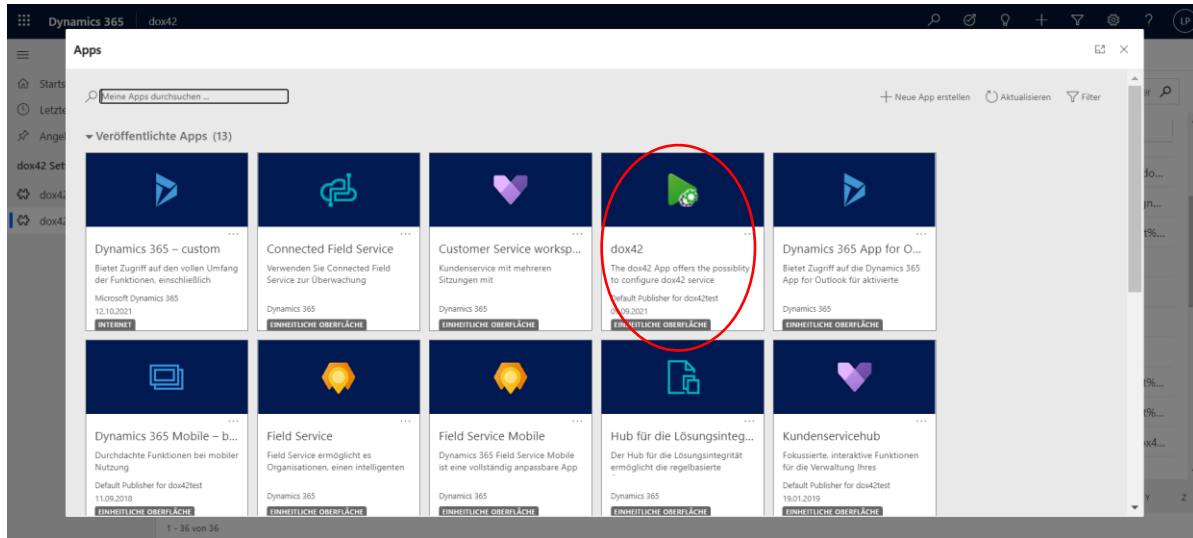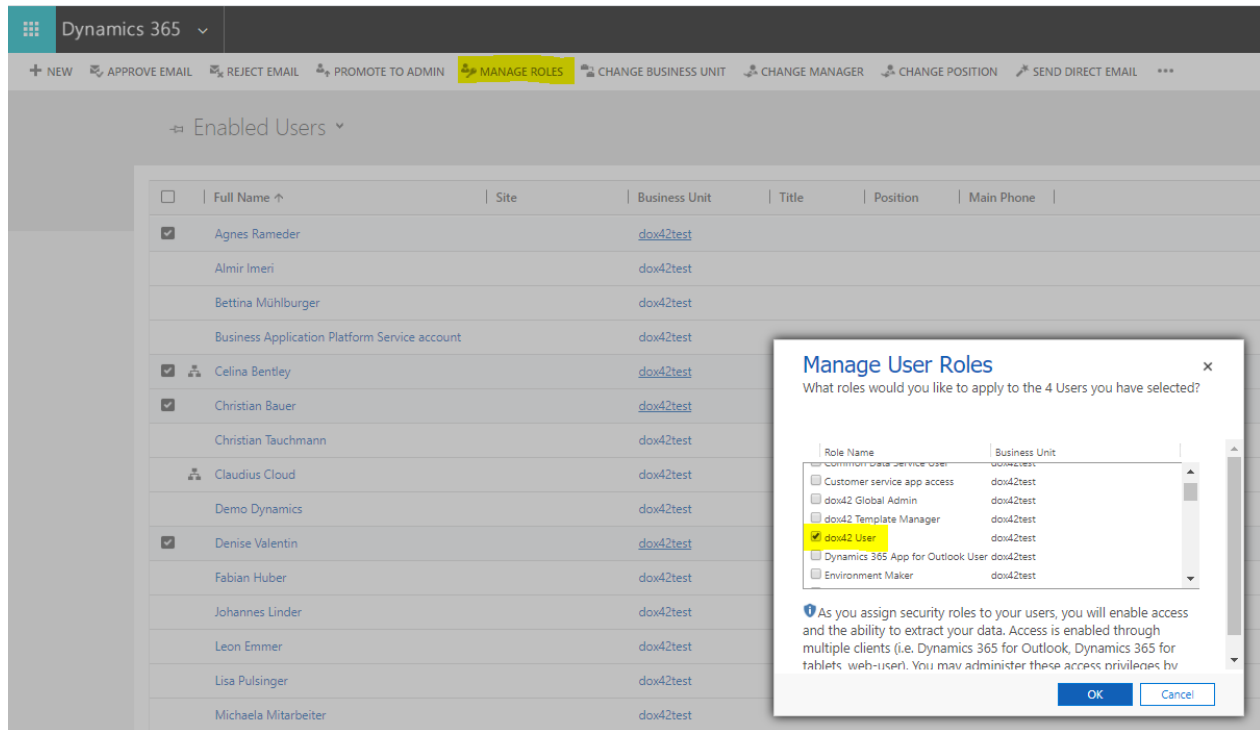

9.3. Security Roles

The solution also contains security roles which regulate the access to the two dox42 entities "dox42 Service Connection" and "dox42 Template Connection". The dox42 Global Admin has full rights for both entities, whereas the dox42 template manager can edit dox42 template configurations, but can only read service connections. **The dox42 User role only has read only permissions on both entities, and needs to be assigned to all users generating documents.**

Manage your security roles by switching to the menu **Settings > Security > Users**. Then select your desired users, click on Manage Roles and add the dox42 Security Roles to the users.
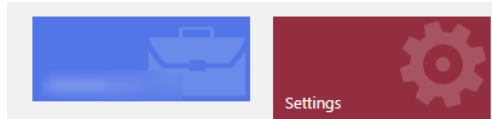

By default the rights for the "dox42 User" and "dox42 Template Manager" are set on business unit level. Should you require the rights to be set on organization level, please switch to Settings > Security > Security Roles, select the desired security role, move to the "Custom Entities" Tab and click a few times on the rights until they are set on the appropriate level. A green dot indicates rights on organizational level.
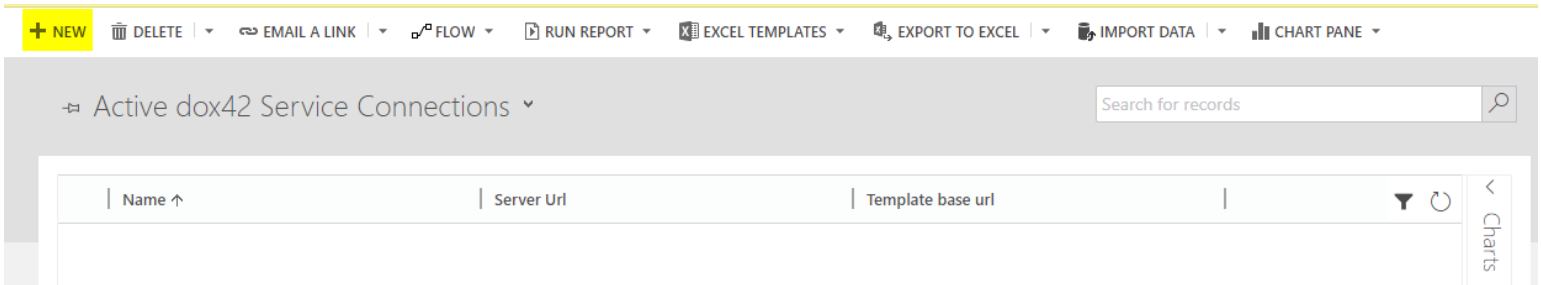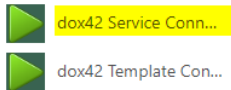
 logo
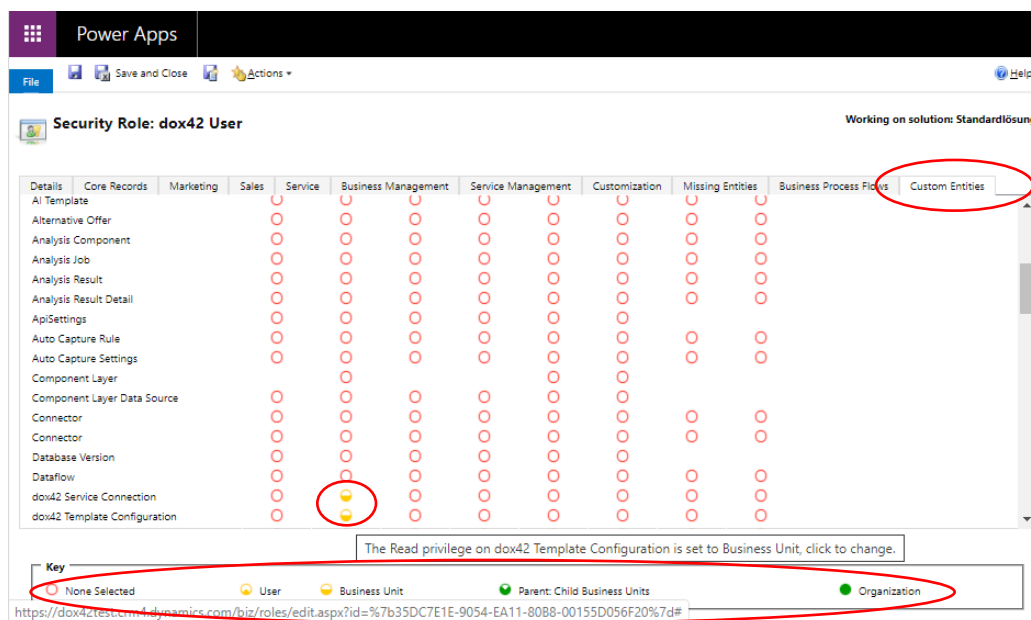
9.4. Adding dox42 Server configurations to Dynamics 365

Add a new dox42 Service Connection either via the dox42 custom entities in the settings area, or via the dox42 model driven app:

For AAD authentication with adal paste the "Application (client) ID" and the "Directory (tenant) ID" from the Application Registration into "App Id (for AAD)" and "Tenant Id (for AAD)".
You can skip this step, if you do not need AAD authentication. For instructions on your dox42 app registration with AAD, please follow the steps mentioned in Chapter 6.

This is the interface when using the dox42 model driven app: (Go to your Dynamics Apps > dox42 > dox42 Service Connections and click on New.
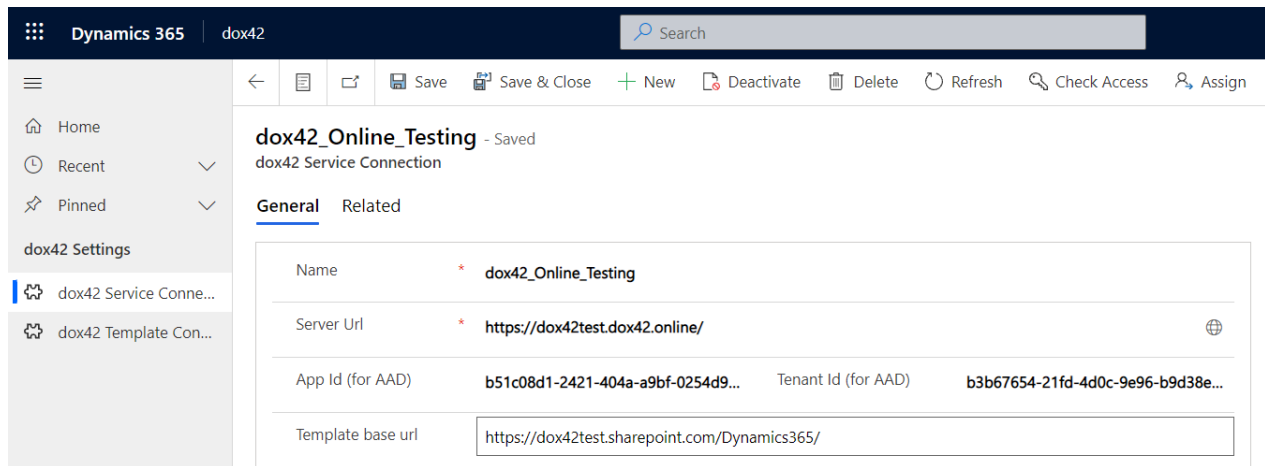
9.5. Adding template configurations to Dynamics 365

Add a new dox42 Template Configuration either via the dox42 custom entities in the settings area, or via the dox42 model driven app:

Enter the values for your dox42 Service, Template, Http Method, Operation and Query String. When working with the dox42 pop up (integration option 2) also add the primary entity, language and other parameters.

- **Name:** When using option 1, the name of the template configuration has to match the name you will use in the ribbon workbench button command action (see 9.6). When using option 2 (dox42

pop-up), the name you put here, is the name that will be displayed for this template in the dox42 pop-up

- Location of the template: The location of the template is build out of the service connection (Template base url) and the Template Url of the template configuration. These two values are concatenated. If you are generating Excel spreadsheets, please specify the path of the data map.

- Http Method: Choose if your dox42 call should use the method "GET" or "POST". **If you generate a document from a list with several Ids or when using the dox42 pop-up and a Requestbody as input parameter, specify "POST"**. Selecting "Auto" ensures that whenever you pass a list of Ids, the request method will be POST, otherwise GET is used automatically.

- Operation: Use "Generate Document" for Word templates, "Generate Slides" for PowerPoint templates and "Generate SpreadSheet" for Excel templates.

- Additional Query String: You need to define your dox42 Output actions here. E.g. if you want to return your document as pdf, the Additonal Query string looks like this: ReturnAction.Format=pdf. Find all dox42 parameters in the dox42 Server documentation. In Chapter 5 of this documentation you can find the dox42 CRM output action for attaching documents as an annotation to an entity.
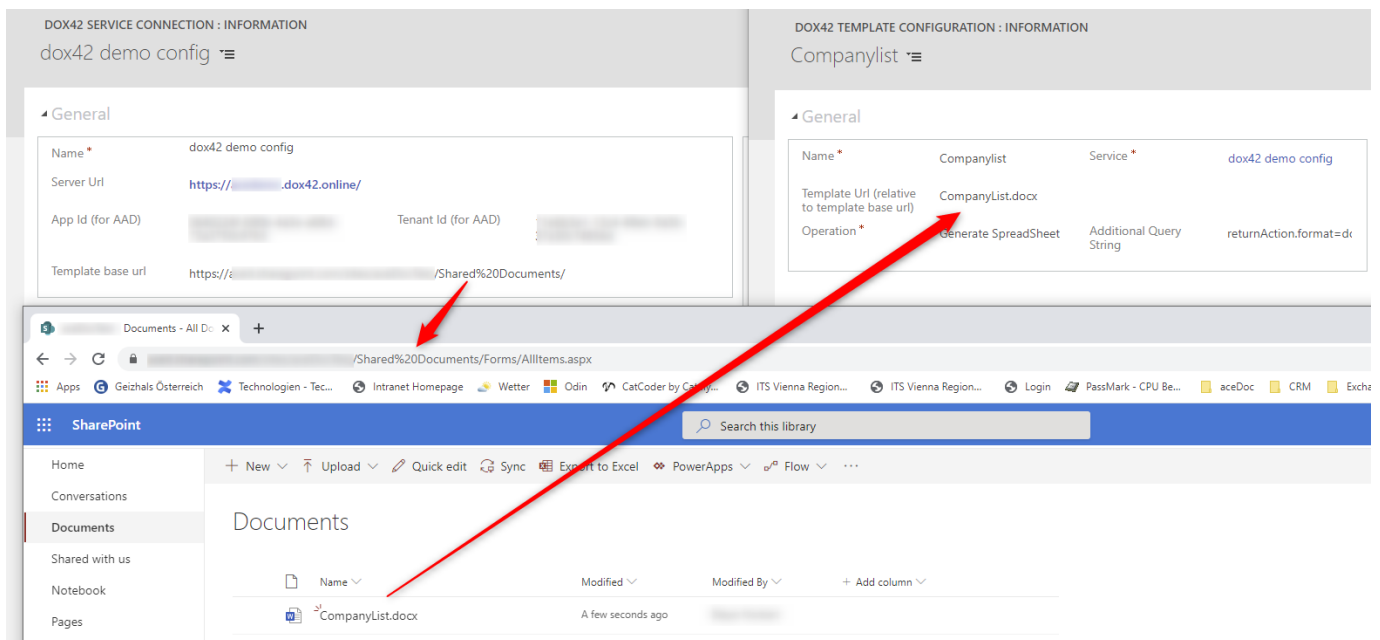
Download the dox42 Server Designer from www.dox42.com/modules/download for an easy set-up of the query string. When using the server designer to set-up your query string, you can just copy and paste the string with the correct format to your Additional Query string field.

dox42 Server Designer:
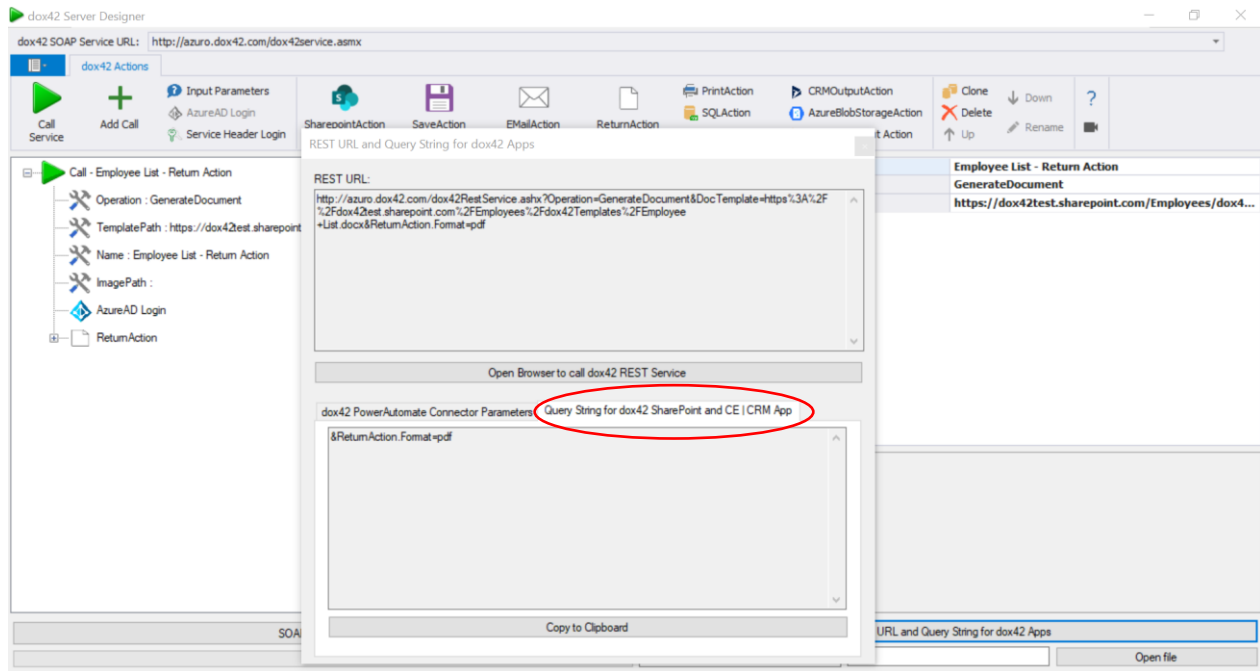
When working with the dox42 pop-up, you should configure your dox42 templates via the model driven app:
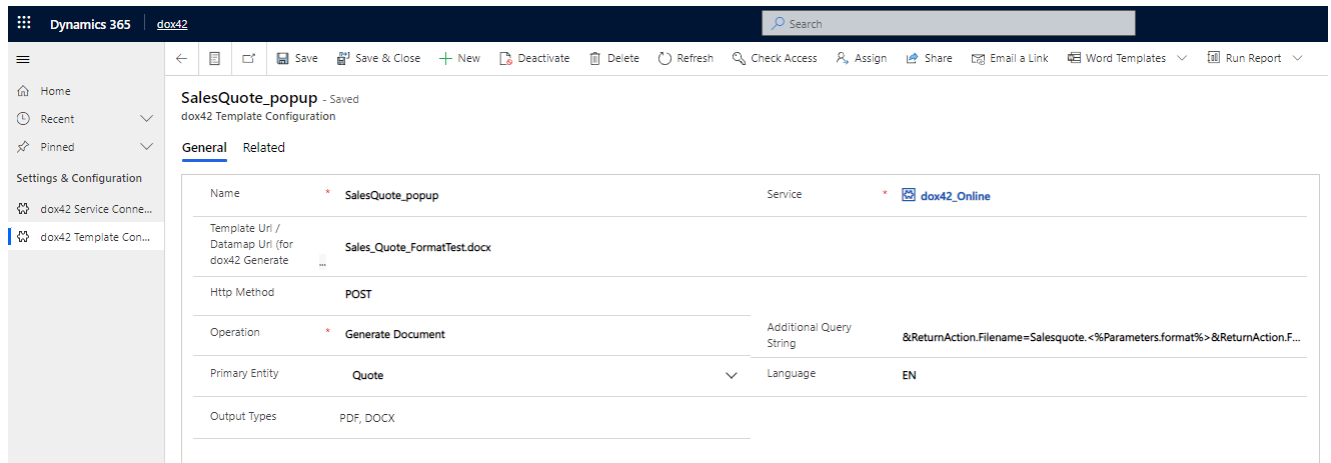
Access the dox42 model driven app through your Dynamics apps and click on dox42 Template Configuration. Click on New. Additionally to all the parameters already mentioned above **you need to choose the primary entity the template should be displayed on – this is obligatory if you work with the dox42 pop-up (integration option 2).** You can also select the language of the template, the format extensions available or any other parameter you decide to provide for your templates.

The Language parameter serves as a categorizer for your dox42 pop-up. If you select English here, then this template configuration will be shown when you select EN in the popup, if you selected DE it will be shown alongside all DE templates in your popup. You can add further languages within the dox42 Dynamics CRM solution (Adavanced Settings > solutions> dox42 client libraries > languages.)

If you use the "Output Types" to dynamically choose the format the document should be generated as from your popup, you will need to specify your input parameter in your dox42 template as "RequestBody" and include "format" as one parameter. See chapter 9.8.2 for a detailed description of the RequestBody input parameter. In this case, the additional query string then needs to have variables for the format. E.g. ReturnAction.Format=<%parameters.format%>

9.6. Adding Buttons to your entity forms, grids and subgrids

To display your dox42 document generation calls on your entity forms and grids you need to integrate them via buttons on the desired position. **You can do this with Ribbon Workbench or Power Apps (see chapter 9.8.5.)**

If you decide to integrate your dox42 buttons via Ribbon Workbench, follow the steps in this chapter. First, set up a new solution in your Dynamics settings (Settings > Solutions > New) and add the entity to which you want to add your dox42 button. You can also upload the custom icons for your ribbon buttons here.
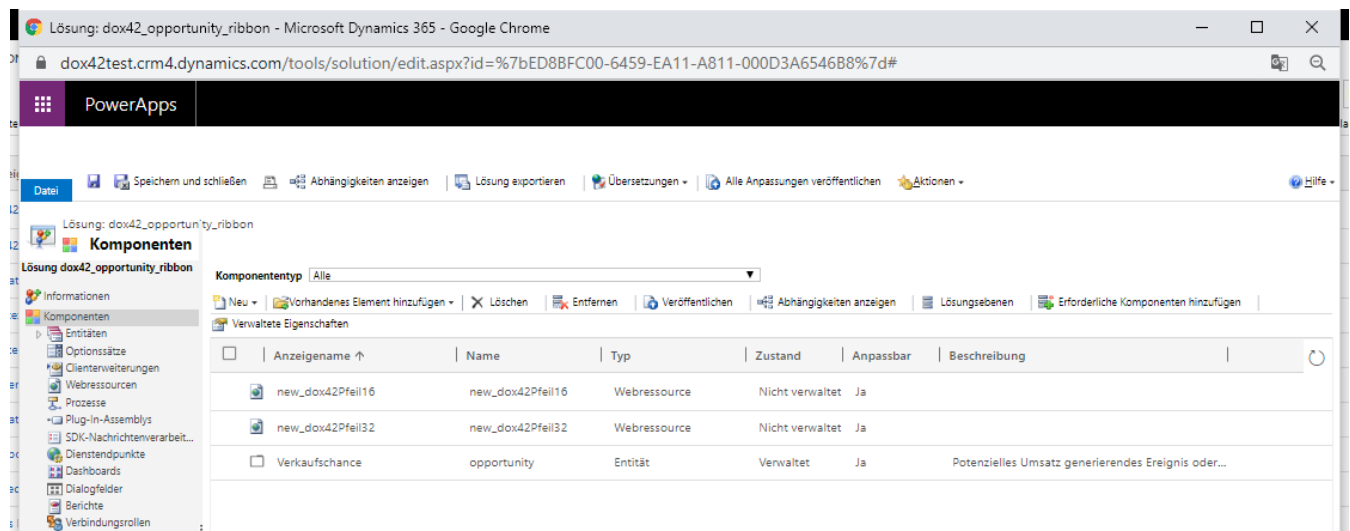

Next, open Ribbon Workbench from your Settings>Customizations page in Dynamics (You need to install it separately to your CRM.)

Open your newly added solution.

First, add a button (1) by dragging it to the desired position on your ribbon (2) and insert information like an ID, label, description and image (3). **Please fill in the "Modern Image" section with the same value you added for "Image 32", if you want your icon to appear in Dynamics CRM Modern UI.**

If you want to add a drop down button, choose "Split Button" (1) and drag it to the desired position on you ribbon (2) , then drag a "Menu Section" (1) to the Split Button. Next, you drag more buttons (1) to the drop down area of your Split Button (2) and configure them as described below (3).
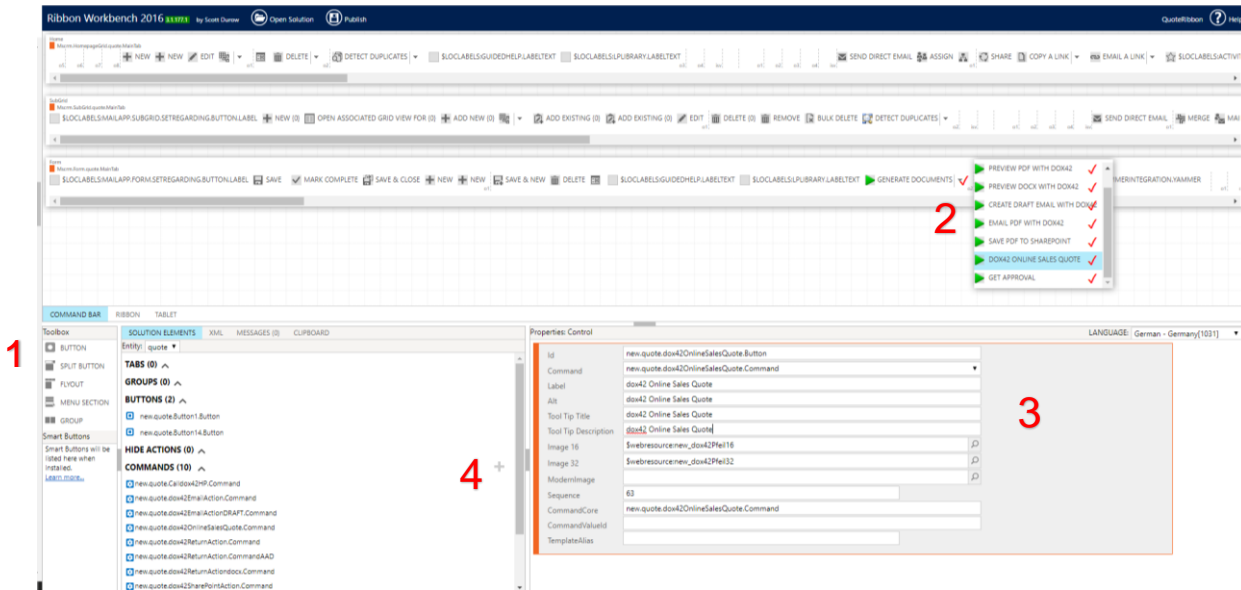
Next you need to add your commands to the button. Click the + sign in the menu Command Bar > Solution Elements in the lower left corner of your screen (4).

9.7. dox42 integration option 1: use one button for each dox42 template

In the example below the CommandAction for Option 1 is defined. By following this approach you will add a separate button for each document template you want to call. **How to add only one button for a list of templates in the dox42 Popup (Option 2)** will be explained in chapter 9.8.4

For the CommandAction define:
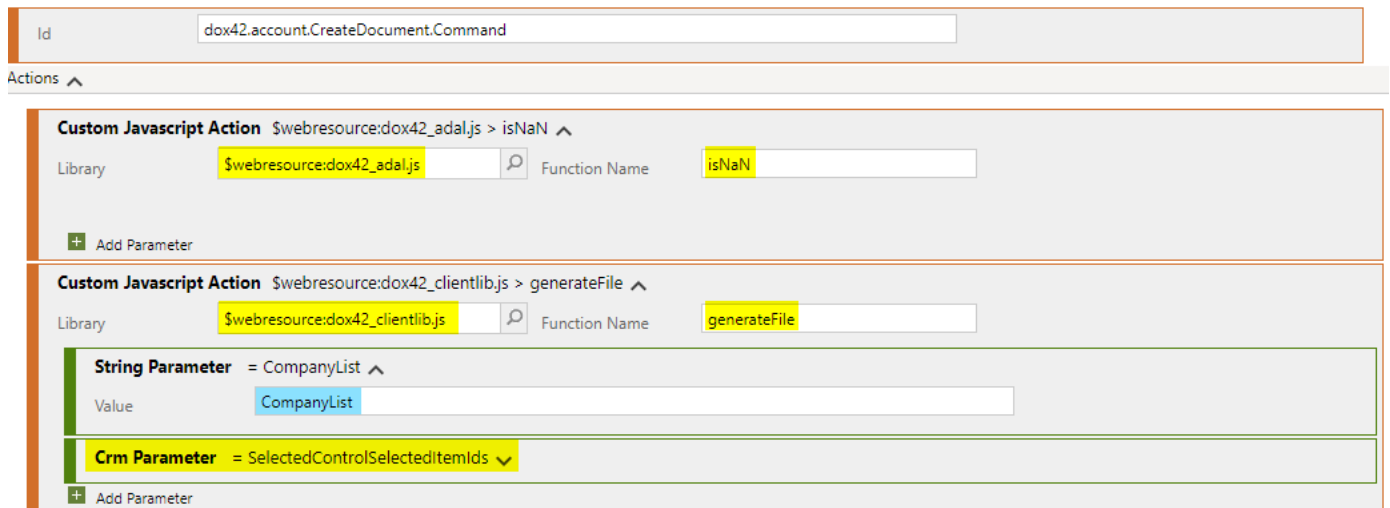
The first JavaScript Action is isNaN from the Library dox42_adal.js.

This loads the ADAL script.

The second JavaScript Actions is generateFile from the Library dox42_clientlib.js.

Pass the name of the Template configuration (configured as dox42 Template – see 9.5) as first parameter and the SelectedControlSelectedItemIds (Crm Parameter) as second parameter to the generateFile method.

You can continue this with as many dox42 commands as you may need. Next, publish your solution (click on Publish in the upper left corner of your Ribbon Workbench solution).

If you switch back to your CRM page, your newly added dox42 buttons will appear in the command ribbon.
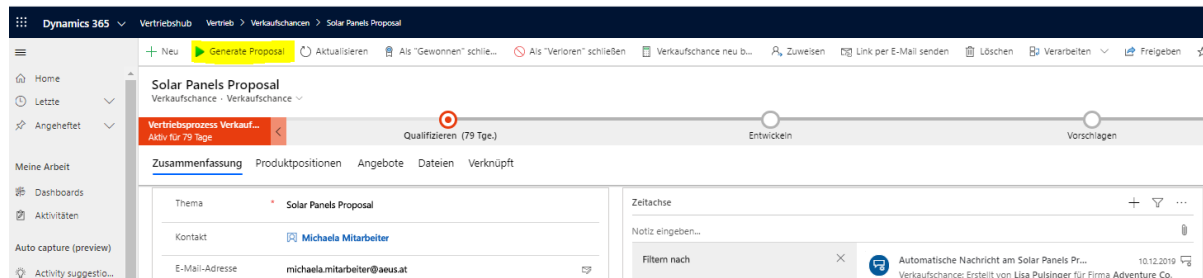
9.7.1.   Adding Ribbon buttons to lists

When adding dox42 buttons to lists, which allows you to select one or several items for your document generation, use the **input parameter "RequestBody"** instead of "id" in your document template. Furthermore, you will need to add a JSON data source to your template. You can find a sample template in your download package (Folder CRM_ProductList_JSON_sampletemplate). In your dox42 template configuration in CRM, specify "Auto" or "POST" as Http Method (see Chapter 9.5).

In Ribbon workbench, instead of placing the button on the form, you need to place it on the homepageGrid.MainTab of your entity (the top ribbon displayed in your Ribbon Worbench set-up).

If you insert the button to a Classic UI CRM environment, please follow the approach described above – see 9.6. For your button to be shown in a Unified UI CRM environment, you need to add a **Display enable rule** (SelectionCountRule with a Minimum of 1) to your button command. This will have the effect that your button in Unified UI environment will be shown as soon as you select at least one item:

Again, after you publish your solution, your button will be shown in the ribbon of your selected form.

9.7.2.   More Information on Ribbon Workbench

If you are looking for a more detailed introduction to Ribbon Workbench, please follow this link.

9.7.3.   Video Guide for D365 CE | CRM dox42 integration (Option 1)

For a detailed video guide on how to integrate dox42 with your Dynamics CRM | D365 CE , watch our dox42 Class tutorials:
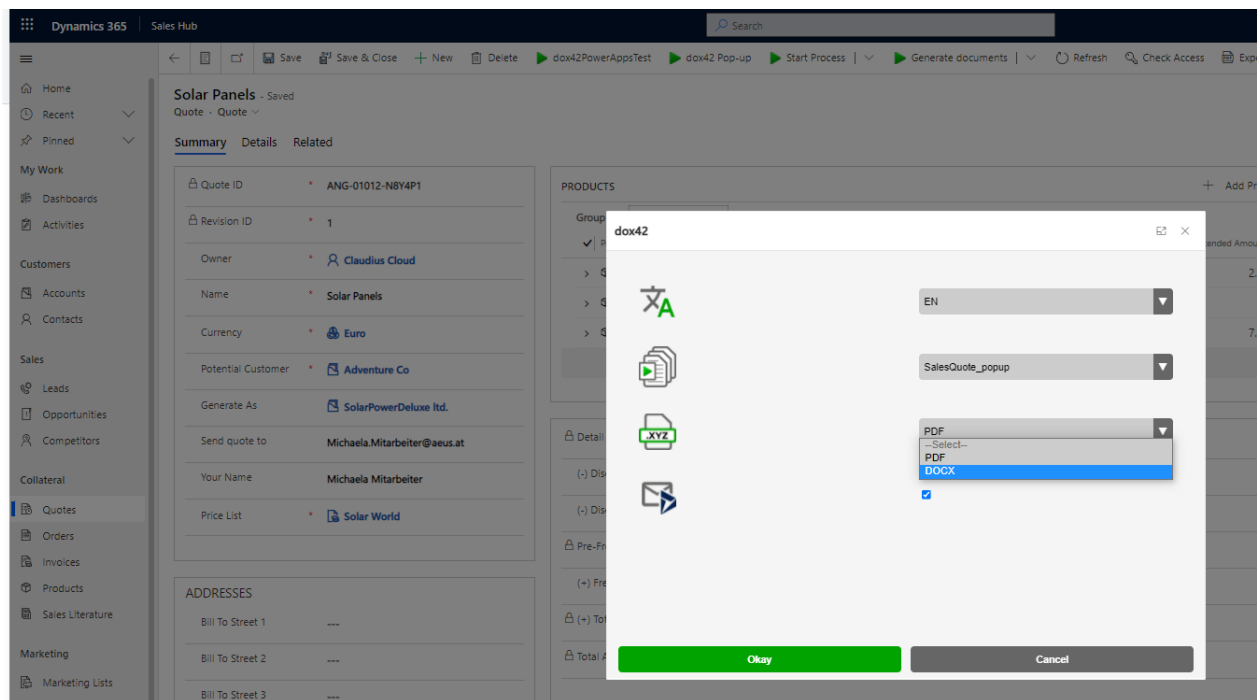
English Class:

https://www.dox42.com/Resources?filter[]=.tutorialvideo.CE-CRM&search=/class/i&

German Class:

https://www.dox42.com/de/Resources?filter[]=.tutorialvideo.CE-CRM&search=/class/i&

9.8. dox42 integration option 2: dox42 popup

This chapter explains how you can work with the dox42 pop-up web resource and how you can customize your dox42 pop-up web resources to your own liking. **(Available from dox42 solution version 1_6 and above).** The dox42 pop-up allows you to choose which template to generate in one simple overview and also gives you the opportunity to pass on further input parameters like a dynamic format, a language switcher, additional text or choosing to attach the document to a CRM email, at runtime. You can specify yourself which parameters should be available on the pop-up. Here is an example of how the dox42 pop-up can look like:

The dox42 solution version 1_6 and above delivers three sample popup web resources.

1. **dox42_apps/dox42_Popup** → includes the template selector, a language switcher and the function to attach your generated document to a CRM e-mail.

2. **dox42_apps/dox42_Popup_Format.htm** → includes the template selector, a language switcher, a file format selector as well as the function to attach your generated document to a CRM e-mail. (shown in screenshot above)

3. **dox42_apps/dox42_Popup_Example.htm** → next to all functions provided in the popups mentioned above, this webresource contains further parameters like an additional comment field or check-boxes. We've included it in the solution to provide you with examples how you can customize the pop-up windows yourself.

When using popups 2 and 3, be aware that your templates need to have their input parameter structured as a Requestbody JSON. This is needed in order to include the dynamic format or other information provided through the popup. You will also need to set the http method to "POST" in your dox42 template configuration and your Additional query string needs to contain dynamic values e.g. ReturnAction.Format=<%JSON.format%>. See chapter 9.8.2 for further details.


9.8.1.   Adjusting the dox42 Popup web resource to your needs

You can skip this chapter, if you do not want/need to modify the delivered web resources to your own needs.

First, get yourself familiar with the dox42 Popup web resource. There are three different dox42 pop-up webresources shipped with dox42 solution 1_6 and above. This chapter explains different sections in the HTML code to give you the information required to customize the pop-up to your needs. You can create as many different dox42 pop-up web resources as you wish.

In the most basic sense, the pop-up web resource is an HTML form that is submitted when you press "Generate" and delivers form data to the Javascript code within the dox42 CRM solution, which in turn converts it to a JSON object that is sent via the request body.

Go to Settings > Solutions > dox42 Client libraries and Open the webresource "dox42Popup.html"
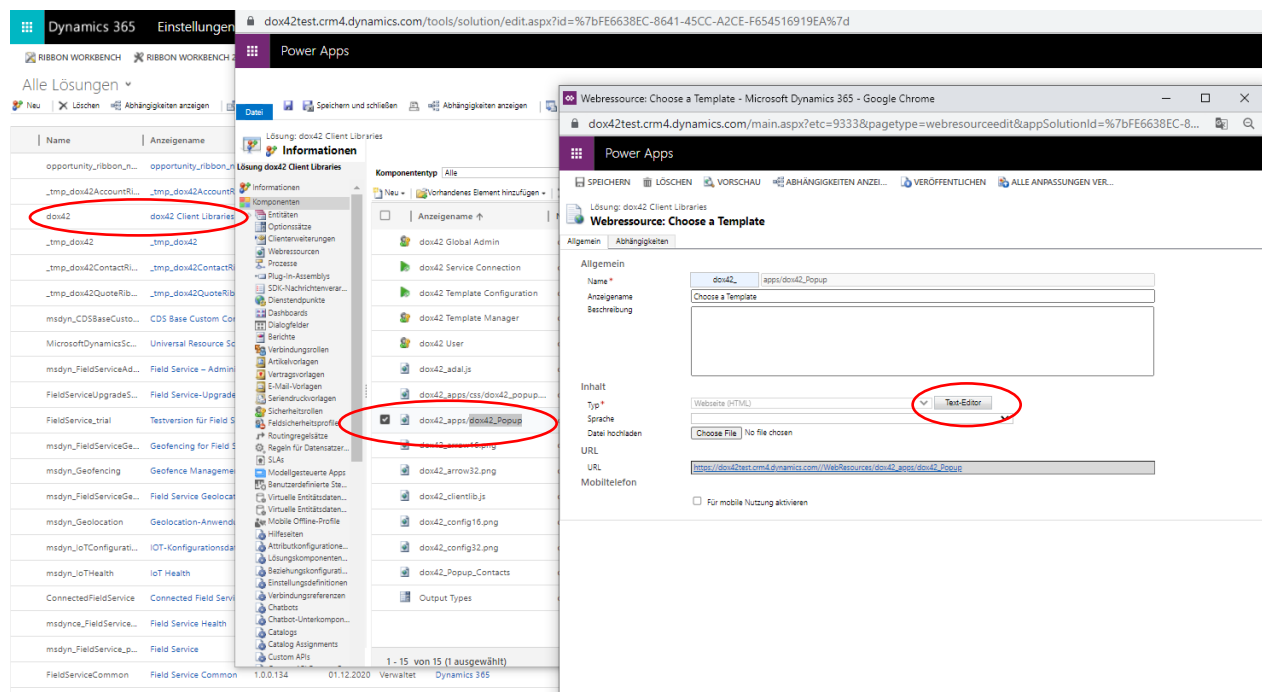
Click on the Text-Editor to get a preview of how the standard pop-up looks like. Change to the Source tab to review and adjust the web resource to your liking.

In the screenshot below you can find the full HTML code that is used within the sample web resource. If you want to further adjust your pop-up, e.g. include different input parameters or remove some, this section will help you.

```
<form id="inputParamForm" class="form-table">
    <div class="form-row">
        <div class="form-col">
            <label>Select a Template: </label>
        </div>
        <div class="form-col">
            <div class="dox42SelectWrapper>
                <select class="dox42Select exclude" name="dox42Template">
                    <option selected="true">Template Name 1</option>
                    <option>Template Name 2</option>
                </select>
            </div>
        </div>
    </div>
    <div class="form-row">
        <div class="form-col">
            <label>Choose a Format: </label>
        </div>
        <div class="form-col">
            <div class="dox42SelectWrapper">
                <select class="dox42Select" name="format">

                    <option selected="true">docx</option>
                    <option>pdf</option>
                </select>
            </div>
        </div>
    </div>
    <div class="form-row">
        <div class="form-col">
            <label>Enter a value: </label>
        </div>
        <div class="form-col">
            <input class="dox42Input" name="TextboxValue1" placeholder="Enter a value">
        </div>
    </div>
    <div class="form-row">
        <div class="form-col">
            <label for="chk1">Checkbox 1</label>
            <input type="hidden" value="off" name="CheckboxValue1">
            <input id="chk1" type="checkbox" class="dox42InputParamCheckbox" name="CheckboxValue1">

        </div>
        <div class="form-col">
            <label for="chk2">Checkbox 2</label>
            <input type="hidden" value="off" name="CheckboxValue2">
            <input id="chk2" type="checkbox" class="dox42InputParamCheckbox" name="CheckboxValue2">

        </div>
    </div>
    <div class="form-row">
        <div class="form-col">
            <label for="chk3">Send CRM Email?</label>
            <input id="chk3" type="checkbox" class="dox42InputParamCheckbox”
exclude" name="dox42CRMEmail">

        </div>

    </div>
</form>
```

The main form for input parameters and other options may contain any kind of form element, for example textboxes, checkboxes, selects, radio buttons etc. The form data is handled in the clientlib.js file. Now you might notice, some form elements have the HTML class "exclude". With this class you can differentiate which form data elements should be saved to the input parameter JSON object and which should be saved to the session storage where they can be retrieved later for another purpose. **You must retrieve the value of these elements yourself; they will not work automatically.** Choosing this approach, it is recommended that you define global variables at the start of the file to avoid any errors due to typos. For the default pop-up it looks like below.

```
//Defined at the start of the file or before method
Let dox42TemplateValue = "dox42Template";
Let dox42InputParamsValue = "dox42InputParams";
Let dox42CRMEmailValue = "dox42CRMEmail";

//Implementation could look like this
var dox42Template = sessionStorage.getItem(dox42TemplateValue);
var dox42CustomData = sessionStorage.getItem(dox42InputParamsValue);

//Example implementation of a custom excluded element "dox42CRMEmail
var generateCRMEmail = sessionStorage.getItem(dox42CRMEmailValue) != null ?
sessionStorage.getItem(dox42CRMEmailValue) : false;
```

Also, it is recommended to handle some form elements with hidden fields with the same name, especially checkboxes and radio buttons, since these elements are "not valid form data" if they are not checked and will therefore not be saved as form data, which could lead to problems in the template.

```
<div class="form-col">
    <label for="chk1">Checkbox 1</label>
    <input type="hidden" value="off" name="CheckboxValue1">
    <input id="chk1" type="checkbox" class="dox42InputParamCheckbox" name="CheckboxValue1">
</div>
```

The value of the checkbox overwrites the value of the hidden input if it returns a value (i. e. "on").

You can add as many dox42 pop-up web resources as you wish, for example if you want to have a separate pop-up per entity. For that simply copy the HTML content of the "dox42Popup" web resource and paste it into a new web resource within the dox42 client libraries solution.

9.8.2. Setting up RequestBody input parameters

The templates you generate via the dox42 pop-up options 2 and 3 need to have one input parameter called "RequestBody", for pop-up option 1 you can also use "RequestBody" or alternatively "Id" to pull the GUID of your primary entity. In the "RequestBody" input parameter you pass on all additional parameters that you include in your pop-up in JSON format, e.g. a format switcher or an additional text field.

When you generate your document, the Requestbody is saved as JSON and looks like the one in the screenshot below. You can also find it in your logfile on your dox42 Online or Server instance.

```
{
    "UserId":"18156D60-911E-E711-8127-E0071B6641E1",
    "Ids":[
        {
            "id":"73A2C21C-4020-E711-8122-E0071B647F41"
        }
    ],
    "Environment":"https://dox42test.crm4.dynamics.com",
    "format":"docx",
    "TextboxValue1":"BlaBla",
    "CheckboxValue1":"on",
    "CheckboxValue2":"on"
}
```

Within your dox42 Add-in (V4321 and above) you can easily insert a RequestBody sample input parameter, by clicking on Input parameter, selecting the input type Text and clicking on "D365 CE Request Body":
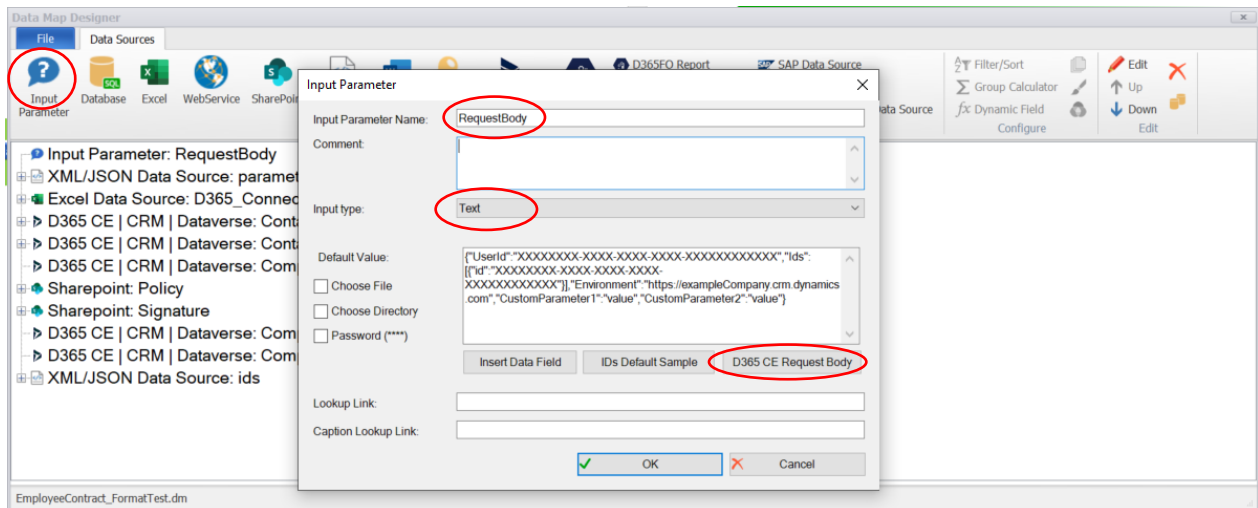

This loads the correct JSON structure for your RequestBody. **Adjust it to your needs and enter one sample entry with valid Id inputs here, to be able to properly test your template. Don't forget to name the input parameter "RequestBody".**

Next the JSON needs to be resolved using an XML/JSON data source in your template. Add a new XML/JSON data source and load your input parameter via the Get functionality. Also here, you can quickly load the correct structure for your JSON by selecting D365 CE Request Body in the SharePoint/MultiValue Field/Apps dropdown:
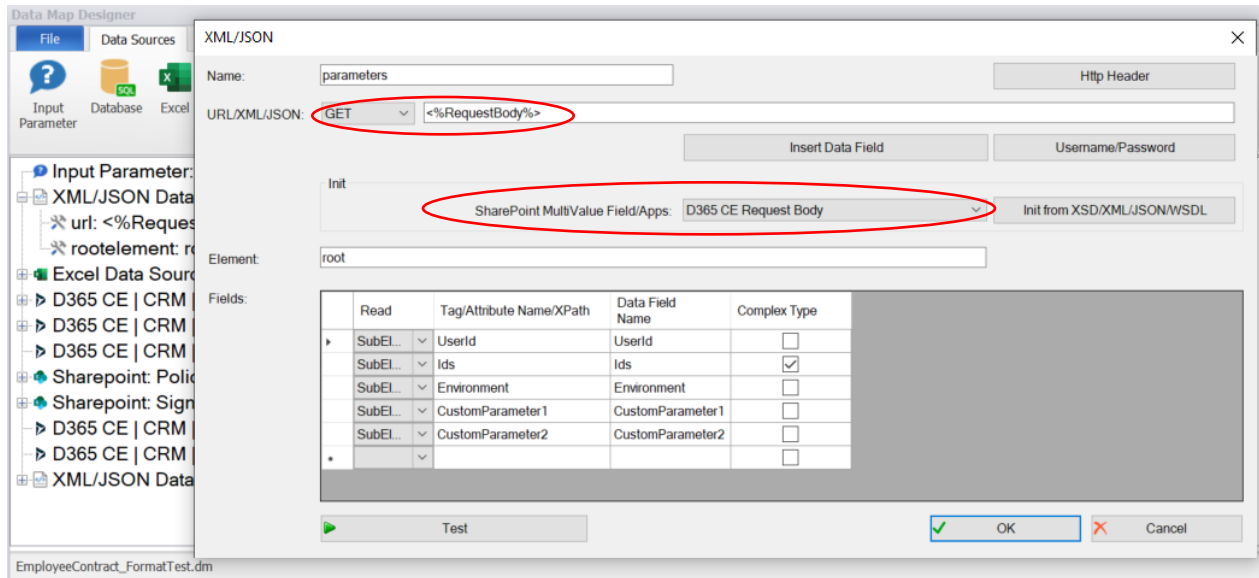
Next, make sure that you resolve the complex type field for Ids. For that, you simply add another XML/JSON data source to separate them into various id entries:
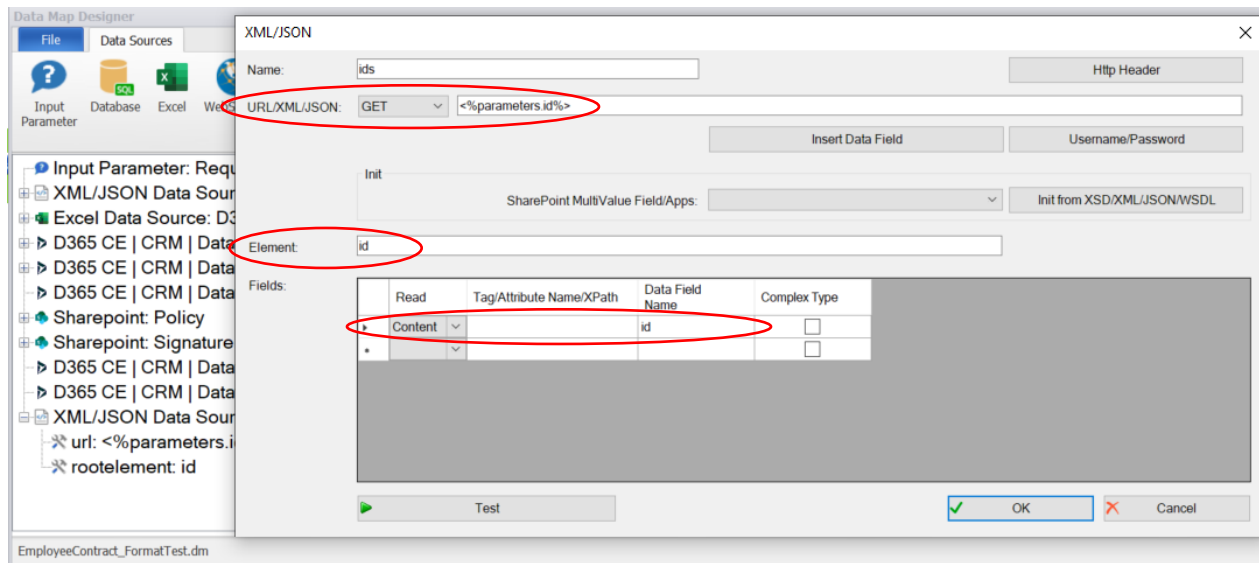

You can now use your two new XML/JSON data sources for anything you want. Use the ids to filter your primary data source, add a text field to your template or define an additional query string in the dox42 Template Configuration in Dynamics (see below) with a dynamic format, for example:

ReturnAction.Format=<%parameters.format%>&ReturnAction.Filename=Contract.<%parameters.format%>

9.8.3. Template Configuration in Dynamics CRM

Next you need to set-up your template configuration via the dox42 model driven app. Make sure that the HTTP method is set to **POST** when using pop-ups options 2 or 3. **Do not forget to set the primary entity for each template configuration you want to show within your popups**. In case you cannot find your desired entity in the primary entity dropdown, access the template configuration via your Dynamics Advanced Settings > Extensions and write the name of the primary entity in the plain text field. All details regarding the template configuration are explained in chapter 9.5.
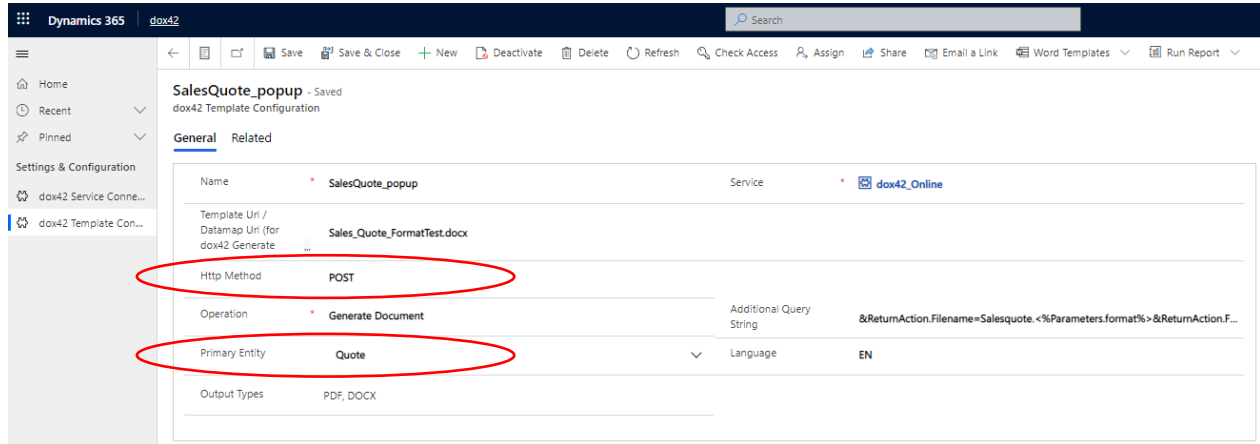
9.8.4. Adding the dox42 pop-up button via Ribbon Workbench

Finally, you need to add one button for your dox42 Popup web resource by using Ribbon Workbench or Power Apps. If you use Ribbon Workbench, the command needs to be structured as in the screenshot below:

You need to call the function "popupWebResource" and the String Parameter you need to supply are the name of the target web resource you want to display in the pop-up and the "SelectedControlSelectedItemIds" CRM parameter.
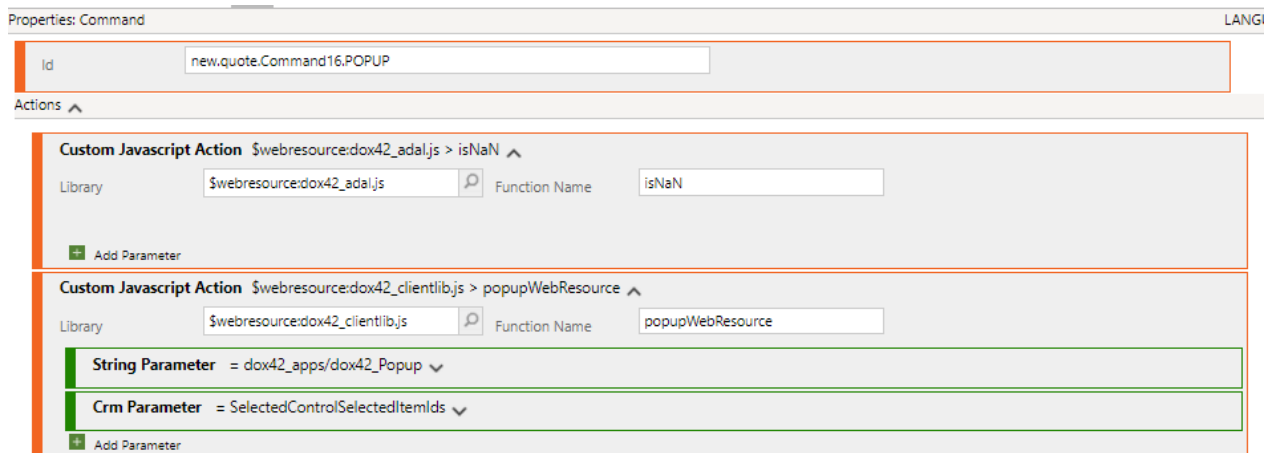
Add the command to your button, click on publish and the button will appear on the location you placed it within Ribbon workbench. A complete description on how to add the button to a form, grid or sub-grid is found in chapter 9.6.
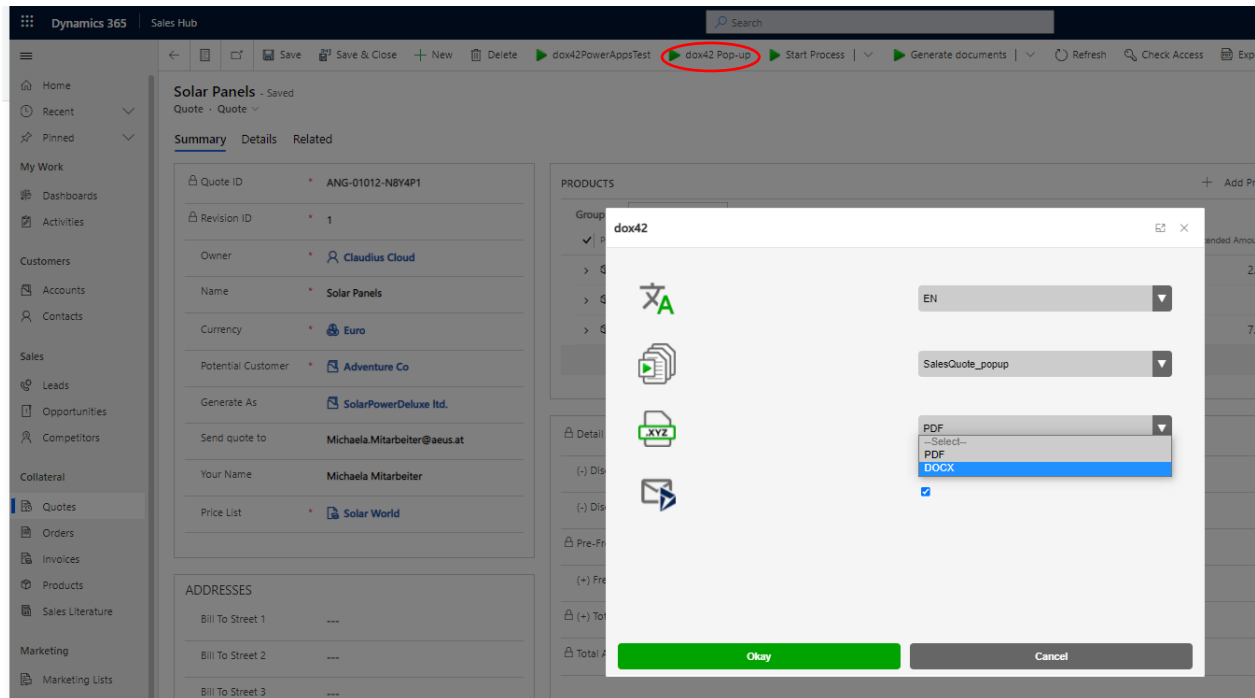

Do not forget to add a button with Ribbon Workbench or Power Apps on every entity you want your pop-up to appear!

9.8.5. Adding the dox42 pop-up button via Power Apps

Another method of how to integrate your dox42 buttons to your entity forms and grids, is to use Power Apps. Microsoft has recently launched this possibility.

Open your D365 CE Solution via Power Apps and click on "edit in preview".
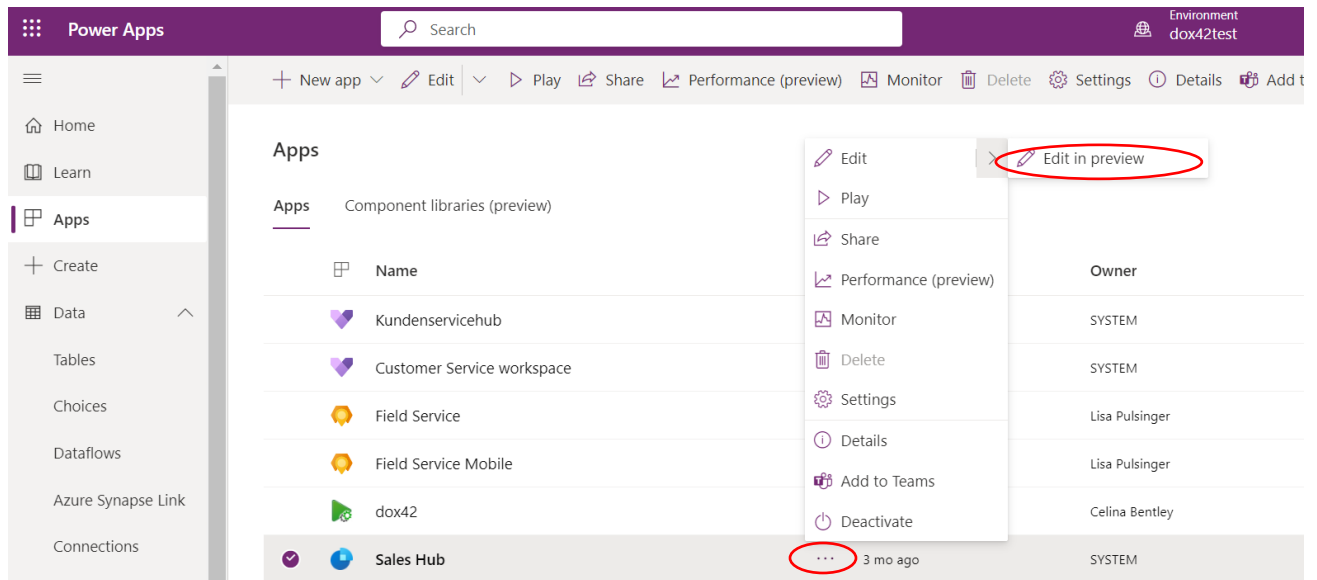
Next, decide on which entity you would like to display your dox42 button on and click on "Edit in command bar".
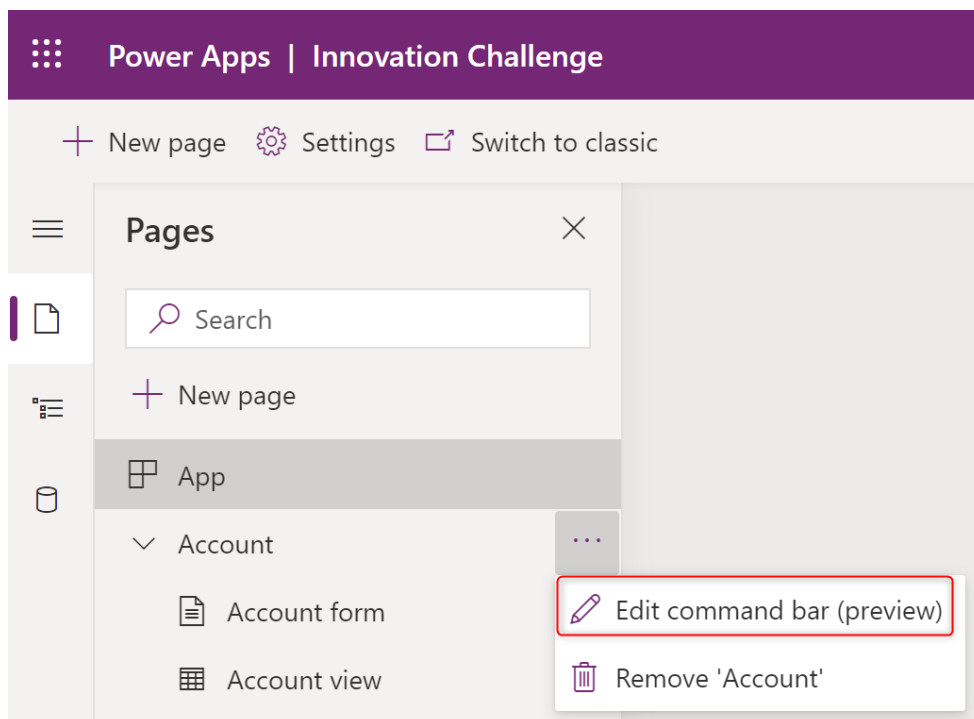
Then select whether you would like to place your dox42 button on the entity form (main form) or on a grid or subgrid.

Now you can add your button via a new command.
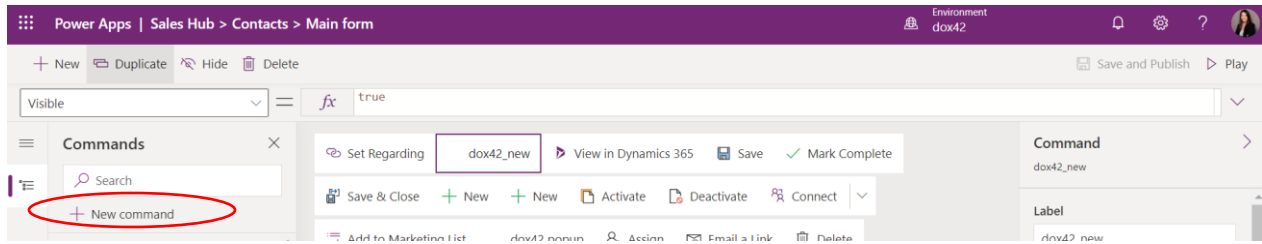
To display the dox42 Popup upon a click on the button you are adding, please add the following details to your command:

To call dox42 via this button, you need to integrate the dox42_clientlib.js Javascript as a "Run Javascript" Action to this command. Add two parameters:

Parameter 1: String Parameter, and the value is the name of the dox42 pop-up web resource you would like to integrate, e.g. dox42_apps/dox42_Popup_Format.htm (for pop-up options 2 and 3 please adjust accordingly).

Parameter 2: SelectedControlSelectedItemIds

To be able to add a Javascript Function to your button, you need to change the visibility from "Always show" to "Show on condition from formula".

See https://docs.microsoft.com/en-us/powerapps/maker/model-driven-apps/use-command-designer#actions for further details.

Once you are done, click on "Save and Publish" and the button with the dox42 Pop-up will appear in your D365 CE:
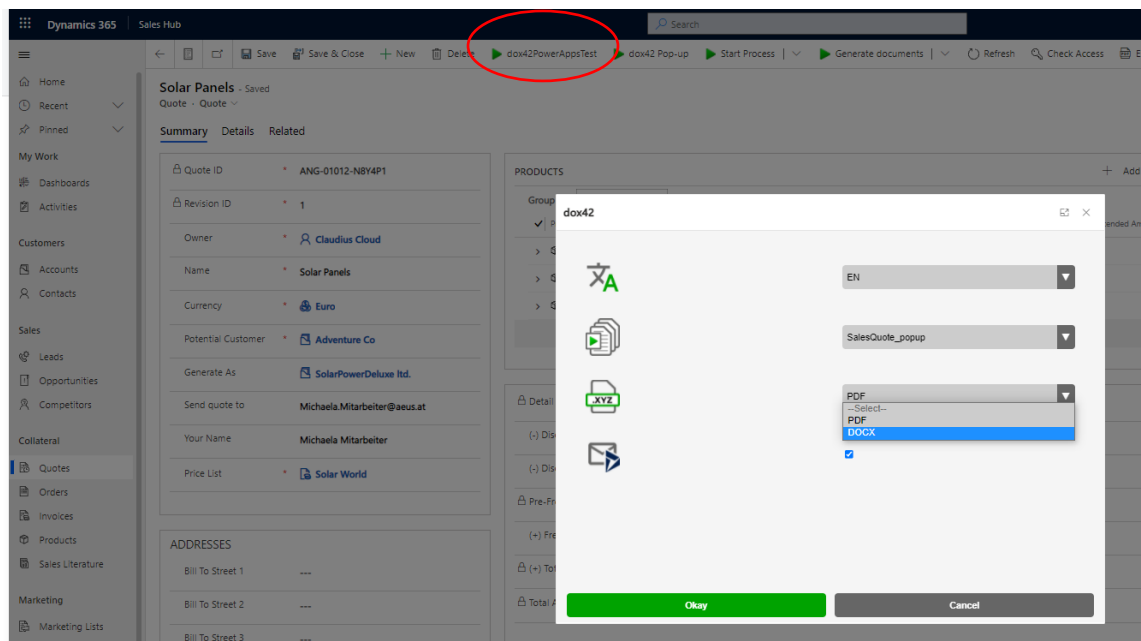
# 10.   Adding a dox42 Call via IFRAME to your Dynamics CE|CRM Forms

As an alternative to chapter 9 and if you wish for your dox42 calls to be added directly to an entity form, rather than via a Ribbon Workbench button in the command bar, you can integrate dox42 calls via an HTML webresource/IFRAME to your entity form.
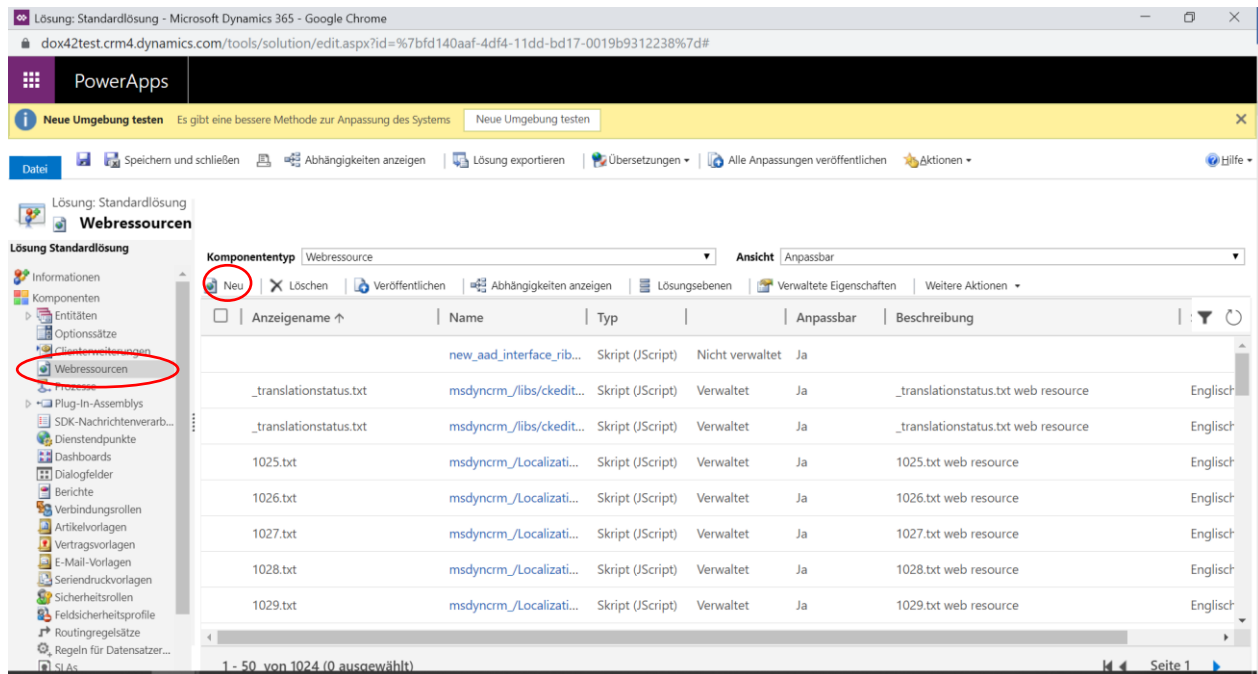
10.1.   Configure the JavaScript Interface

First, configure your JavaScript Interface. For this, please follow the instructions for the AzureAD_dox42_javascript_interface mentioned in the Azure Active directory documentation. You can find the JavaScript in your download package.

10.2.   Set up an HTML webresource for the JavaScript Interface

For this please change to Settings > Customization > Customize System.
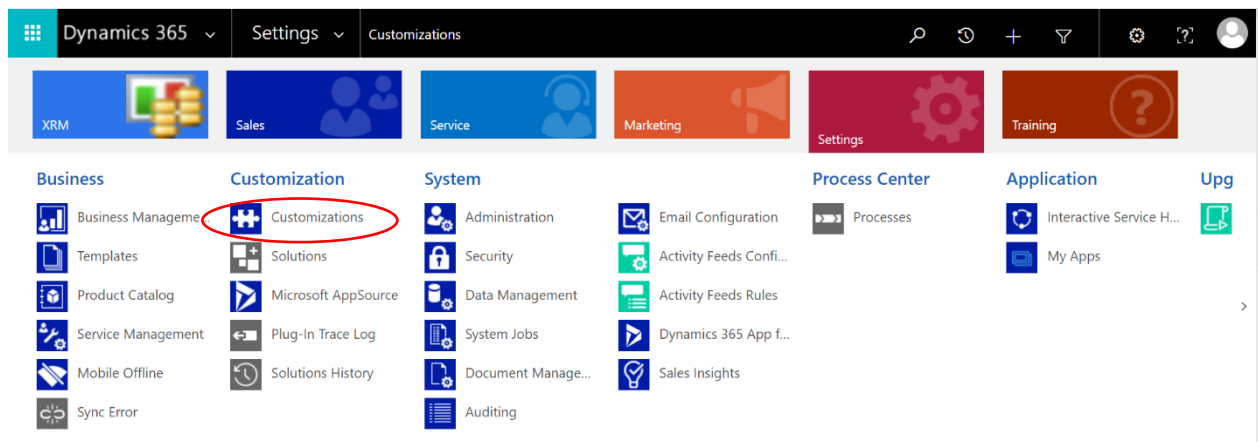
Now a new window opens with all your custom components of your CRM. Click on the web resources button in the left context menu and after that click on New.

Now add your previously configured AzureAD_dox42_javascript_interface JS as **Type HTML** webresource.

Click the button "Text-Editor" to add your JavaScript/HTML Code.

Save and Publish your webresource.

You need to copy the URL of your newly added web resource and add it as reply URL to your Azure Active Directory App Registration.

10.3.   Add the web resource as IFRAME to your CRM form

Next you need to insert your web resource to your CRM page as an IFRAME using the D365 CRM form editor. First open the desired form editor, e.g. from your quote overview.
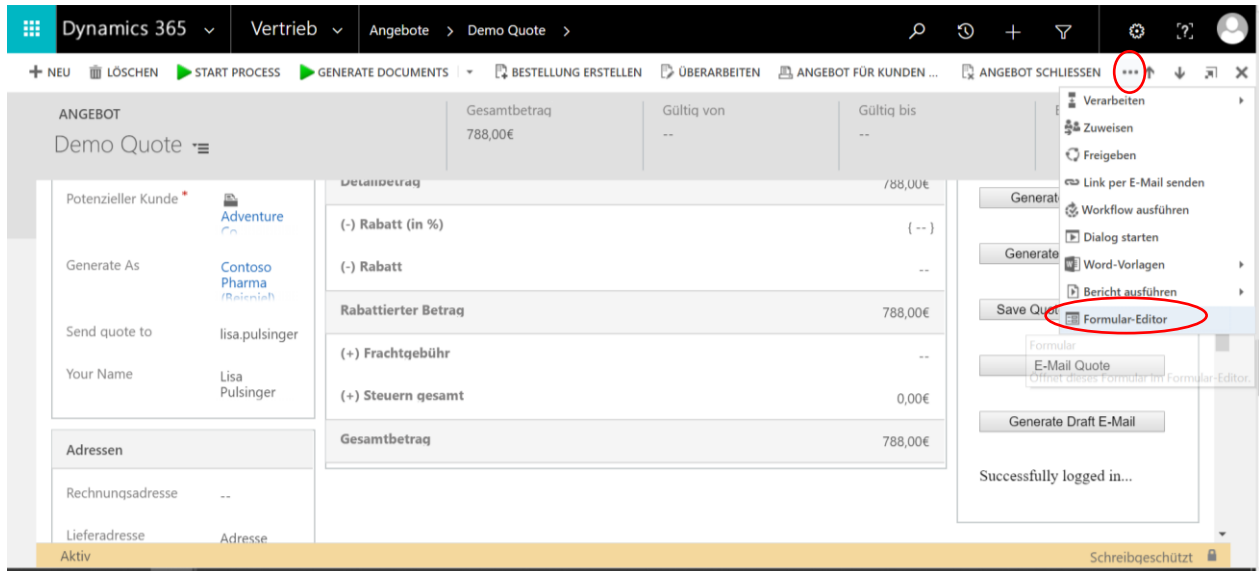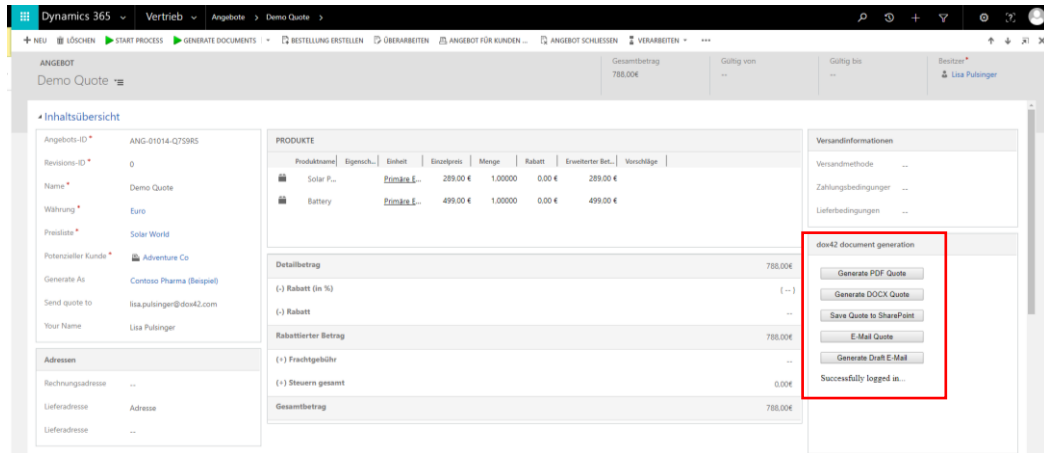
Now select the area where you would like the webresource to appear on your form, click the "Insert" ribbon in the menu bar and select IFRAME.

Now configure your IFRAME settings. For the URL please add the URL of your previously added webresource, as described in chapter 10.2.
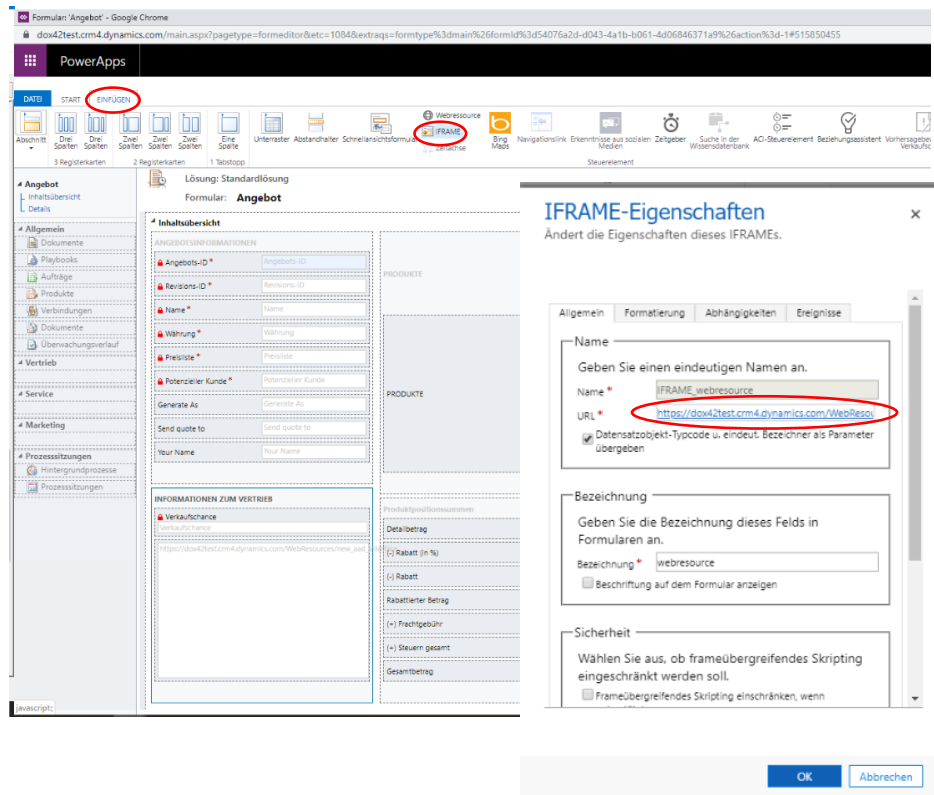
Click OK, Save and Publish in order to have your dox42 calls appear on your form.

11. Support

Should you have any questions, please do not hesitate to contact support@dox42.com. We are happy to help you!

Good luck with dox42!