Office 365 Proposal Creation Add-in

Getting Started Guide

Version 1.0

Contents

[Overview 1](#_Toc4490624)

[Setup Azure App Service 1](#_Toc4490625)

[Azure App Registration 1](#_Toc4490626)

[Update Proposal Manager Registration 8](#_Toc4490627)

[Deploy Solution 8](#_Toc4490628)

[Update App Settings 8](#_Toc4490629)

[Update Client App Settings 10](#_Toc4490630)

[Update the Proposal Creation Addin manifest 10](#_Toc4490631)

[Build the application 11](#_Toc4490632)

[Publish to Azure 12](#_Toc4490633)

[Load Proposal Creation Addin 13](#_Toc4490634)

## Overview

The purpose of this document is to provide guidance on setup and deployment to an ISV partner who is familiar with Office 365, Microsoft Azure and web application implementation.

**Prerequisites**

To setup the solution, make sure that you have:

* Office 365 Tenant (Business or Enterprise SKU)
* Microsoft Azure subscription
* Visual Studio 2017 (Enterprise, recommended)
* Ensure that you have global admin access on the Office 365 tenant where the solution is being deployed and proceed as follows.
* Node.js
* NPM
* Office 2016: The Proposal Creation Add-in uses the Office JavaScript SDK and Word JavaScript API which require Office 2016.
* Ensure that the SharePoint root site is accessible by all the users of the organization. Verify it is shared to “Everyone except external users”.

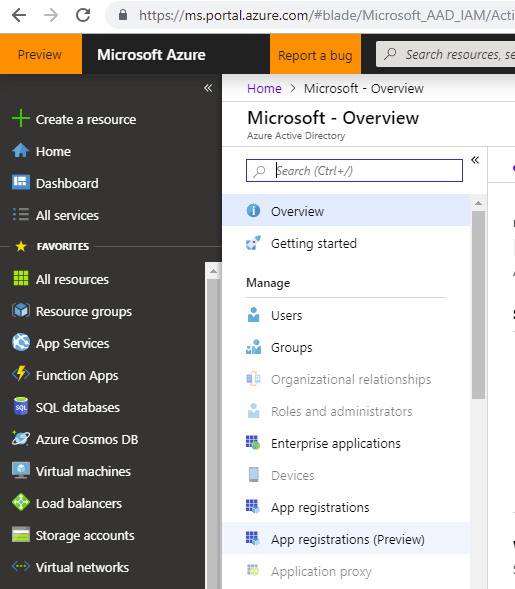
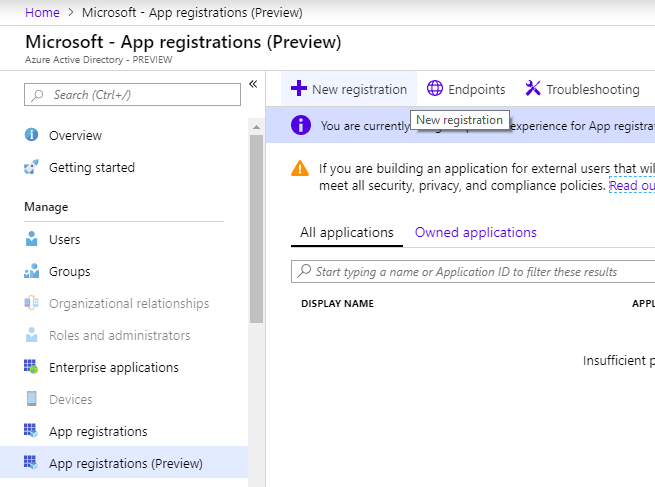
## Setup Azure App Service

Make sure that you have owner access on the Microsoft Azure subscription where the web application is planned to be deployed and made available to the users.

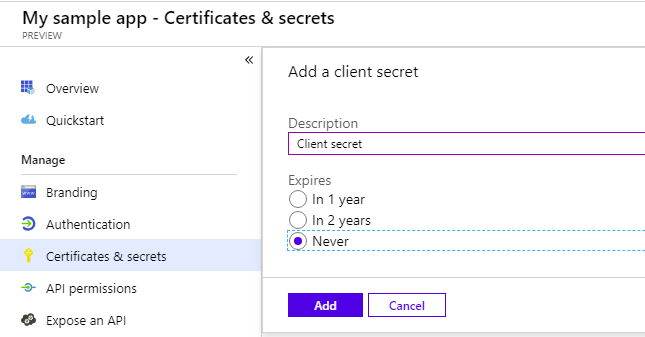
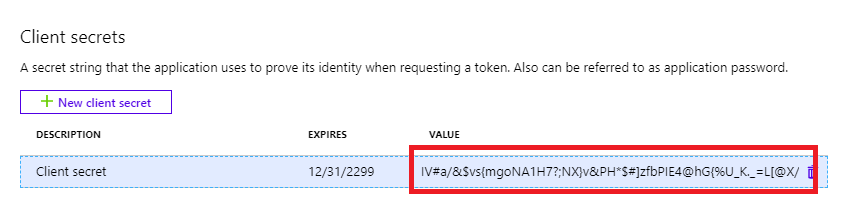
### Azure App Registration

First step is to create the Proposal Creation Add-in web application. Make sure that you have Owner access on the Azure subscription before proceeding.

Sign into [Microsoft Azure portal](https://portal.azure.com/).

1. Go to Azure Active Directory and select the App registrations (Preview) blade:  
   
2. Click on the **New registration** button:  
   
3. Add the name of the application being registered.
4. Select the supported account type: “**Accounts in any organizational directory and Microsoft accounts (e.g. Skype, Xbox, Outlook.com)**”
5. Add the Redirect URI and click on the “**Register**” button

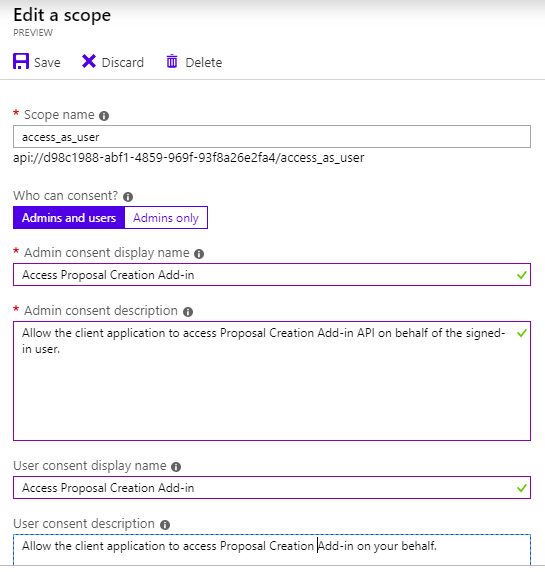
**Create the Client secret:**

1. Click on the Certificates & secrets menu item
2. Click on the **New client secret** button
3. Add a **description** and choose the **Never** expires option as shown below:  
   
4. Click on the **Add** button
5. Copy and store the secret value as it is required later:  
   

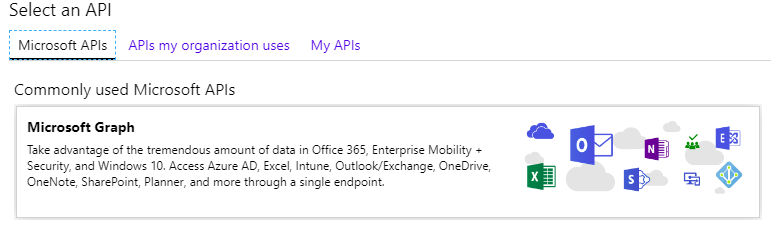
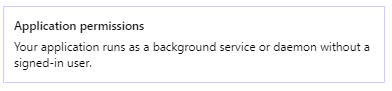
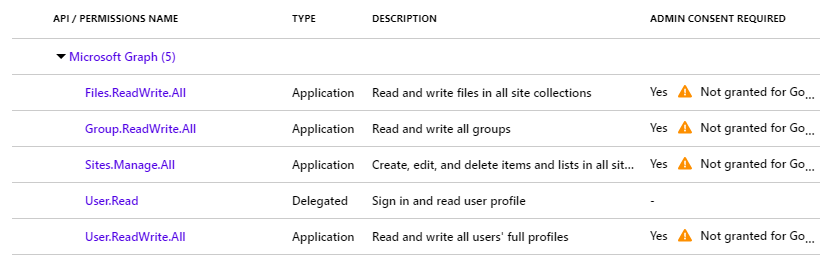
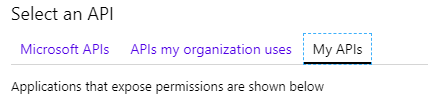
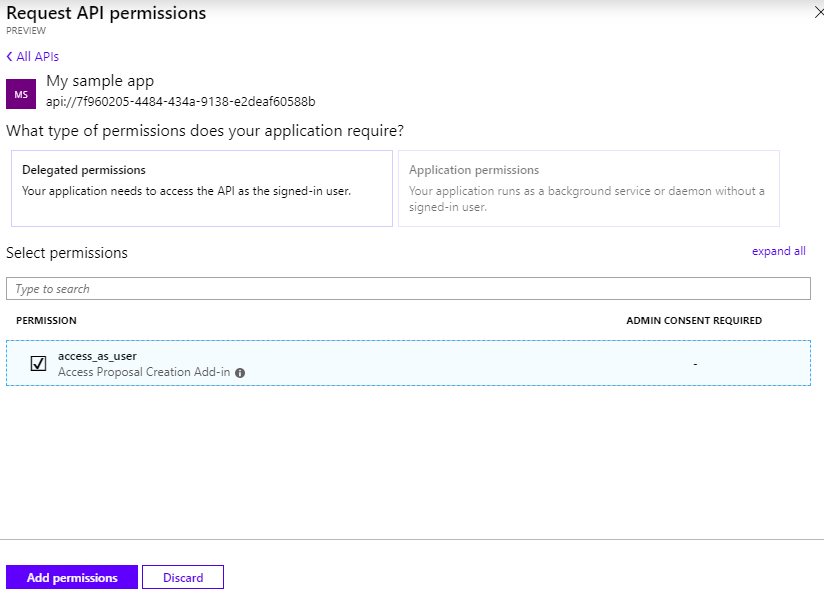
**Add the Redirect URI list**

1. Click on the **Authentication** menu item
2. The Proposal Creation Add-in requires 3 redirect URLs:
   1. https://{proposalCreationWebSite}/auth/end
   2. https://{proposalCreationWebSite}/auth
   3. https://{proposalCreationWebSite}
3. Click on the **Save** button.

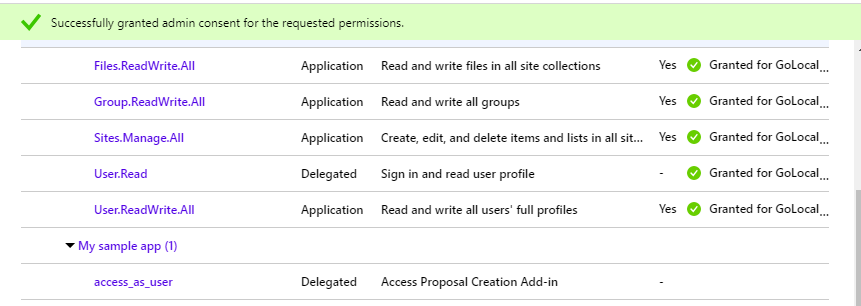
**Expose the Proposal Creation API**

1. Click on the **Expose an API** menu item
2. Click on the **Add scope** button
3. Enter the Scope name: **access\_as\_user**
4. Choose **Admins and users**
5. Enter a consent name: **Access Proposal Creation Add-in**
6. Enter an admin consent description: **Allow the client application to access Proposal Creation Add-in API on behalf of the signed-in user.**
7. Enter a user consent display name: **Access Proposal Creation Add-in**
8. Enter a user consent description: **Allow the client application to access Proposal Creation Add-in on your behalf.**   
   
9. Click the **Add scope** button

**Add the API permissions**

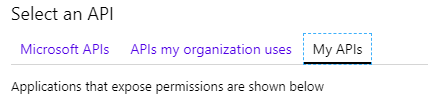
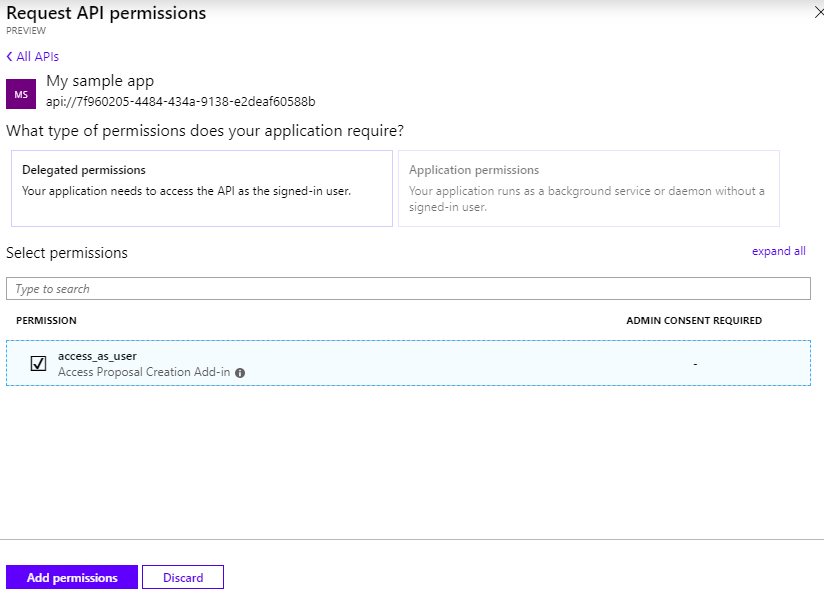
1. Click on the **API permissions** menu item
2. Click on the **Add a permission** button
3. Choose **Microsoft Graph**  
   
4. Select Application permissions:  
   
5. Add the following permissions:  
   
6. Finally add the Proposal Creation add-in API permission, click on the **Add a permission** button
7. Select My APIs tab:  
   
8. Select the application from the list below
9. Check the **access\_as\_user** permission  
   
10. Click on the **Add permissions** button

**Grant consent**

1. Click on the **Grant admin consent for tenant\_name** button
2. A popup window will be displayed, click on the **Yes** button
3. Once the consent is granted the permissions list should look like the image below:  
   

### Update Proposal Manager Registration

Sign into [Microsoft Azure portal](https://portal.azure.com/).

1. Choose the Proposal Manager Application.
2. Select the **API Permissions** menu item
3. Click on the **Add a permission** button
4. Select My APIs tab:  
   
5. Select the Proposal Creation add-in application that you registered in the previous steps from the list
6. Check the **access\_as\_user** permission  
   
7. Click on the **Add permissions** button

## Deploy Solution

Get a local copy of the latest version of the source code and open the solution in Visual Studio.

### Update App Settings

Open appsettings.json located at .\CommLendingWeb and update as follows

|  |  |  |
| --- | --- | --- |
| Property | Description | Example Value |
| **WebHooks:DynamicsCrm:SecretKey** |
| opportunity | Secret shared with Dynamics 365 for opportunity creation | “01234567890123456789012345678901” |
| connection | Secret shared with Dynamics 365 for connection creation | “01234567890123456789012345678901” |
| **Dynamics365** |
| OrganizationUri | Dynamics 365 organizartion instance api url | "https://<org>.api.crm<region\_number>.dynamics.com/" |
| ProposalManagerCategoryId | Id of the Proposal Manager role category in Dynamics 365 | 741530000 |
| ProposalManagerSite | Fully qualified url of the root sharepoint site of the tenant | "https://<tenant>.sharepoint.com" |
| ProposalManagerBaseSite | Relative path from the root site to the Proposal Manager site | "sites/proposalmanager" |
| RootDrive | Name of the root of the proposal manager site’s drive | "Shared Documents" |
|  |  |  |
| TemporaryFolder | Name of the folder where opportunity content is stored temporarily before an administrator creates the corresponding team. | “TempFolder” |
| OpportunityMapping | Mapping configuration for opportunities | See opportunity mapping schema |
| **General** |
| SiteId | SharePoint site where the application is hosted | “<tenant>.sharepoint.com” |
| ProposalManagerApiUrl | Proposal Manager application url. | "https://proposalmanager.azurewebsites.net” |
| **OneDrive** |  |  |
| WebhookSecret | Secret shared with OneDrive to secure webhook calls | "8E0836F5524D4307893D929B8EBE98CF" |
| FormalProposalCallbackUrl | Url where OneDrive will call the middleware whenever a Formal Proposal is updated | "https://<api\_url>/api/document/FormalProposal" |
| ProposalManagerBaseSiteId | Proposal Manager site id | "<tenant>.sharepoint.com,<guid>,<guid>", |
| RootTempFolderName | Relative path from the drive’s root to the temporary folder | "/TempFolder" |
| AttachmentCallbackUrl | Name of the root of the proposal manager site’s drive | "https://<api\_url>/api/document/Attachment" |
| **Proposal Manager** |  |  |
| AppUrl | Proposal Manager’s url | "https://<proposal\_manager\_url>" |
| ApiUrl | Proposal Manager’s api url | "https://<proposal\_manager\_url>/api" |
| CreatorRole | Details of the creator role in Proposal Manager (usually the relationship manager) | See role schema |

### Update Client App Settings

Open appconfig.ts located at .\UI\src\config and update the applicationId indicated below.

|  |
| --- |
| private static ApplicationId: string = "<applicationId>"; |

### Update the Proposal Creation Addin manifest

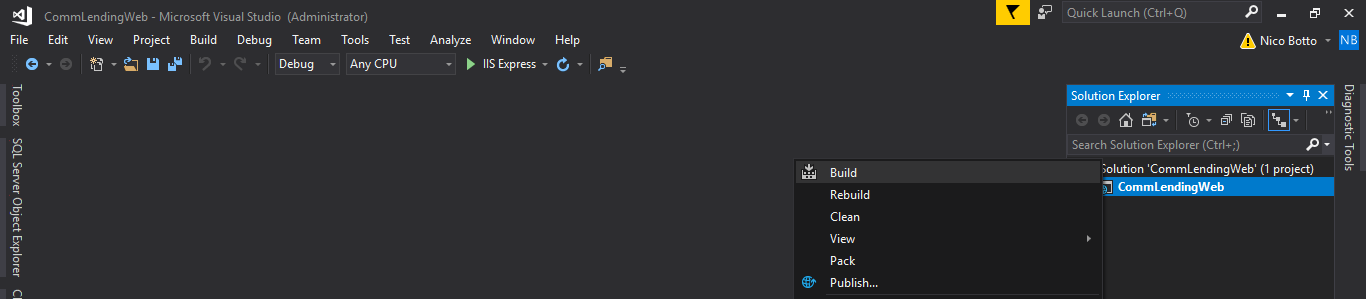
Open the comm-lending-manifest.xml located at .\Manifest and update the following settings:

|  |
| --- |
| <!-- IMPORTANT! Id must be unique for your add-in, if you reuse this manifest ensure that you change this id to a new GUID. --> <Id>**ENTER NEW GUID**</Id>  <!-- Icon for your add-in. Used on installation screens and the add-ins dialog. -->  <IconUrl DefaultValue="https://**<proposal\_creation\_web\_url>**/dist/assets/icon-32.png" />  <HighResolutionIconUrl DefaultValue="https://**<proposal\_creation\_web\_url>/**dist/assets/icon-80.png"/> <!-- Domains that will be allowed when navigating. For example, if you use ShowTaskpane and then have an href link, navigation will only be allowed if the domain is on this list. -->  <AppDomains>  <AppDomain>https://login.windows.net</AppDomain>  <AppDomain>https://login.microsoftonline.com</AppDomain>  <AppDomain>https://**<proposal\_creation\_web\_url>**</AppDomain>  </AppDomains> <DefaultSettings>  <SourceLocation DefaultValue="https://**<proposal\_creation\_web\_url>**" />  </DefaultSettings> <bt:Images>  <bt:Image id="Contoso.tpicon\_16x16" DefaultValue="https://**<proposal\_creation\_web\_url>**/dist/assets/icon-16.png" />  <bt:Image id="Contoso.tpicon\_32x32" DefaultValue="https://**<proposal\_creation\_web\_url>**/dist/assets/icon-32.png" />  <bt:Image id="Contoso.tpicon\_80x80" DefaultValue="https://**<proposal\_creation\_web\_url>**/dist/assets/icon-80.png" />  </bt:Images>  <bt:Urls>  <bt:Url id="Contoso.TaskPane.Url" DefaultValue="https://**<proposal\_creation\_web\_url>**" />  <bt:Url id="SmartLink.TaskPane.Url" DefaultValue="https://**<smart\_link\_web\_url>**/word/point" />  <bt:Url id="Contoso.DesktopFunctionFile.Url" DefaultValue="https://**<proposal\_creation\_web\_url>**/function-file/function-file.html" />  </bt:Urls> |

### Build the application

Before building the solution for the first time, take care to navigate to .\UI from the command line, and run **npm install** to install all dependencies. After installing the dependencies, run **npm run build** to build the UI.

Open CommLendingWeb.sln located at .\Web with Visual Studio. Right click on the CommLendingWeb node and choose **Build**:



## Publish to Azure

Choose Build 🡪 Publish Solution to build and deploy the solution to Azure from the Publish UI.

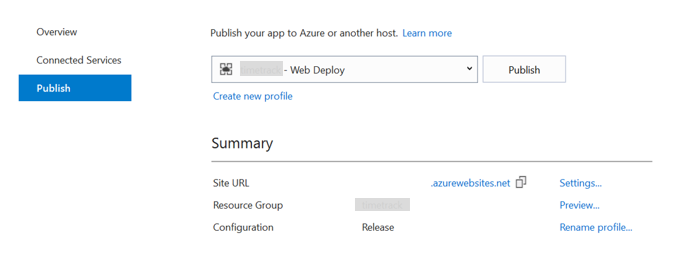
* Select the New Publish profile option and choose Azure App



* Logon using the Azure account that you used to setup the application
* Select the Resource Group where the application is setup and expand, then choose the application



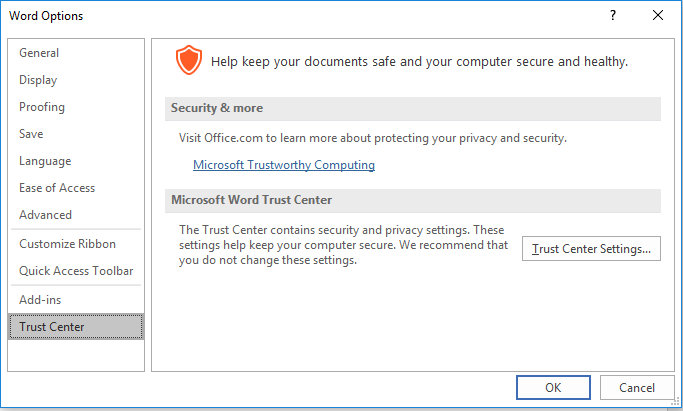
* Click on Publish to deploy the application to the Azure app service

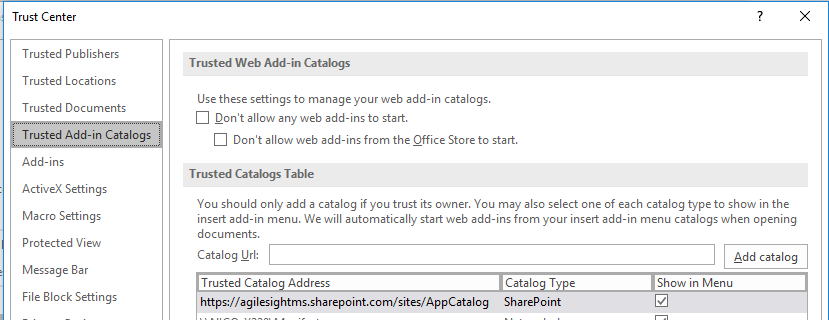
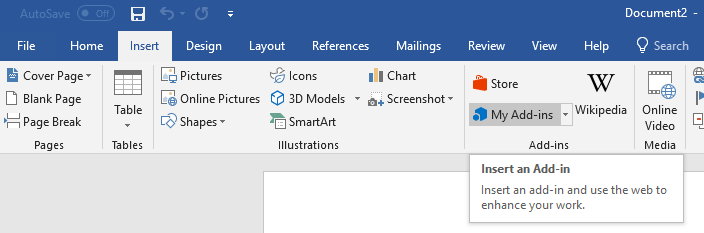
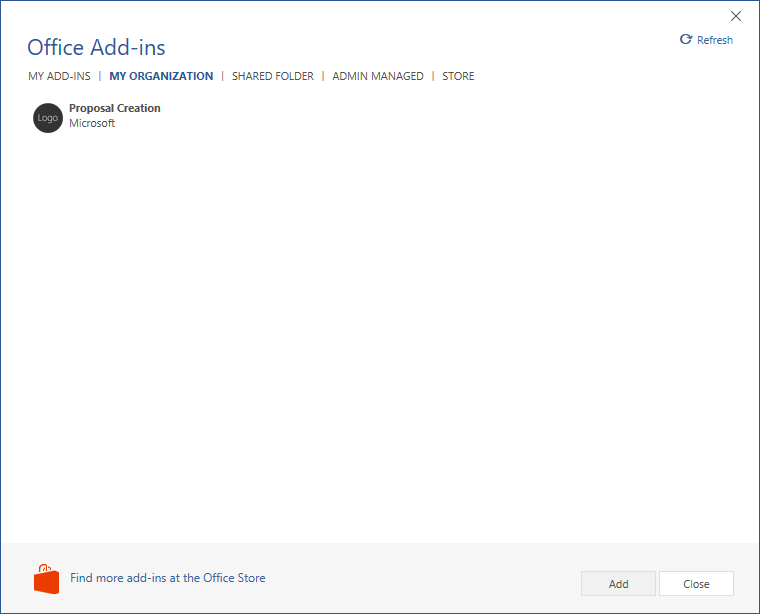


Once publish is completed, the web application is automatically launched on the default browser.

## Load Proposal Creation Addin

1. Open MS Word.
2. Add sharepoint application catalog to Office Trust Center
3. Go to File -> Options -> Trust Center:



1. Click on Trust Center Settings…
2. Select Trusted Add-in Catalogs
3. In Catalog Url add: https://<tenant>.sharepoint.com/sites/AppCatalog   
   
4. Verify that the Show in Menu option is checked
5. Go to the Insert menu item and click on My Add-ins ribbon button   
   
6. Select My Organization tab   
   
7. Select the Add-in and click on Add button

NOTE: If the Word document is opened for the first in MS Teams and the tab MY ORGANIZATION is missing, you need to open the Word document using Word Online and follow the previous mentioned steps for adding the Proposal Creation Add-in. Then you’ll be able to load the Proposal Creation Add-in in MS Teams. Also ensure that the SharePoint root site is accessible by all the users of the organization. Verify it is shared to “Everyone except external users”.