Office 365 Proposal Creation Add-in

Getting Started Guide

Version 1.0

Contents

[Overview 1](#_Toc518156580)

[Setup Azure App Service 1](#_Toc518156581)

[Azure App Registration 1](#_Toc518156582)

[Deploy Solution 6](#_Toc518156583)

[Update App Settings 6](#_Toc518156584)

[Update Client App Settings 6](#_Toc518156585)

[Update the Proposal Creation Addin manifest 6](#_Toc518156586)

[Build the application 7](#_Toc518156587)

[Publish to Azure 7](#_Toc518156588)

[Load Proposal Creation Addin 8](#_Toc518156589)

## 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

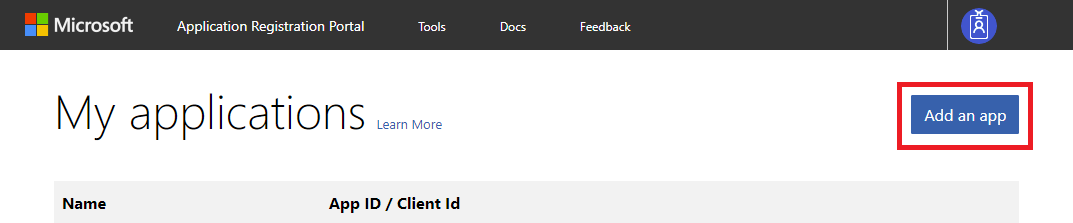
## Setup Azure App Service

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.

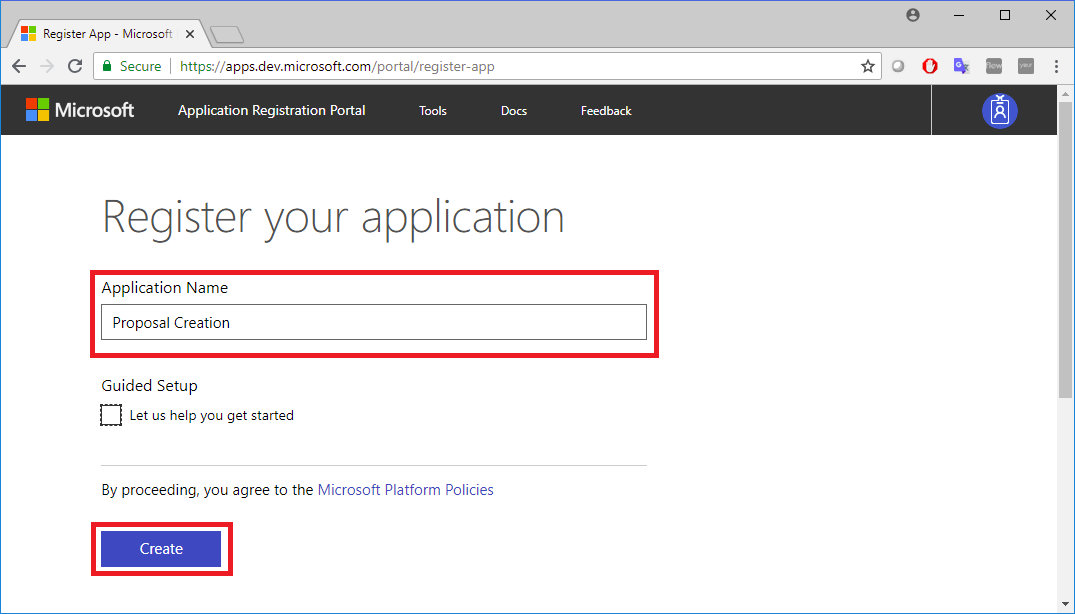
### Azure App Registration

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

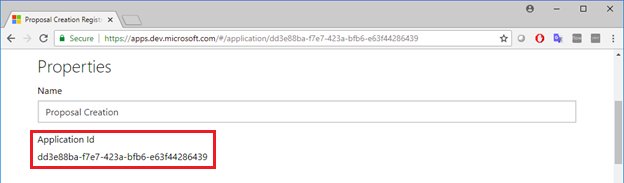
Sign into [Microsoft Application Registration portal](https://apps.dev.microsoft.com).



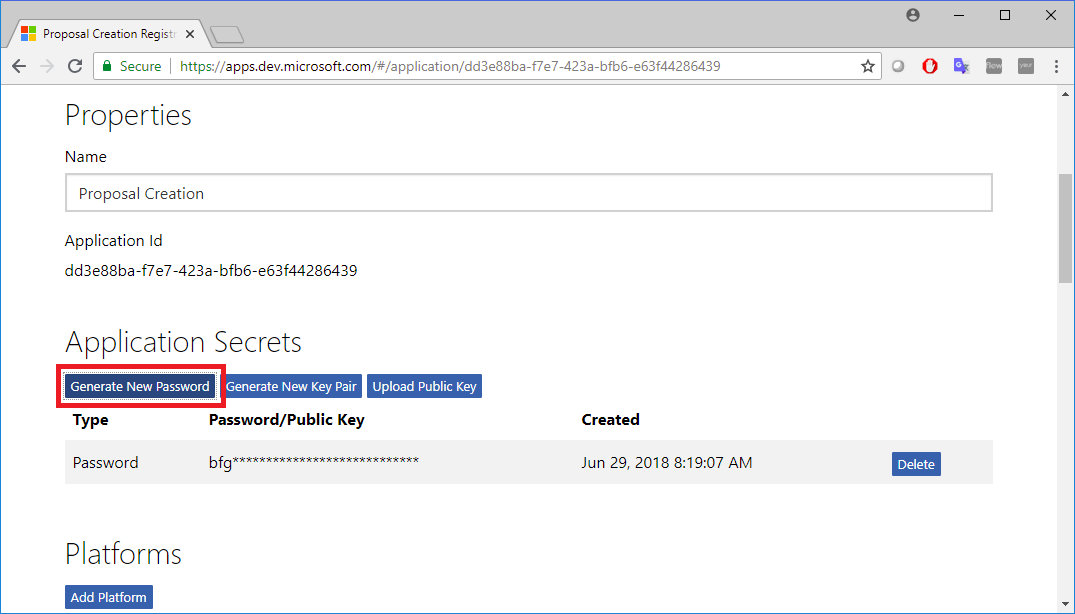
Click on the top left button **App ad app**.



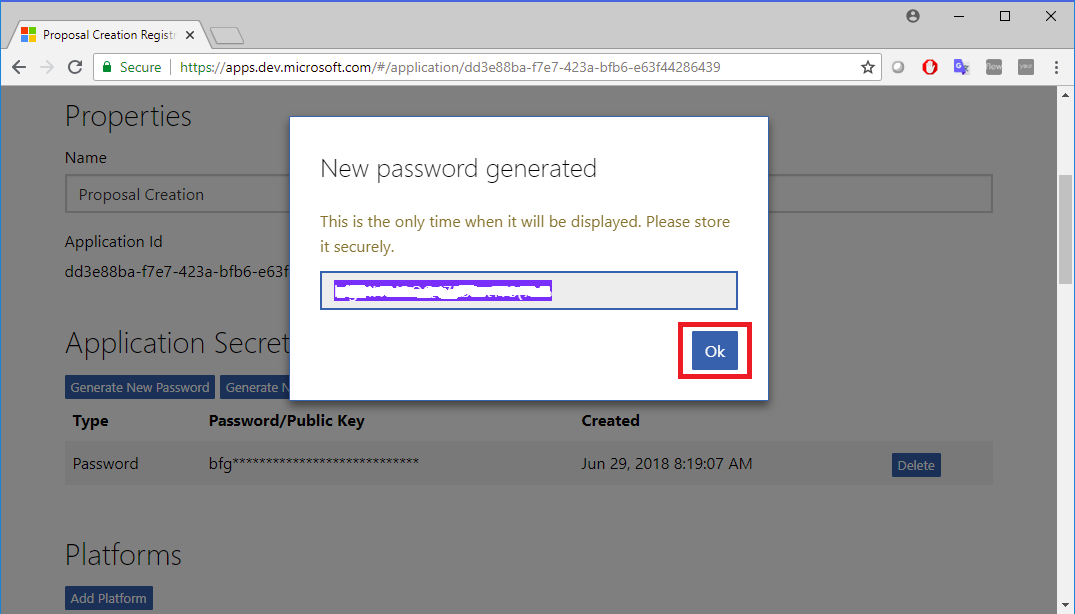
Enter a name for the new application. Then click on **Create**.



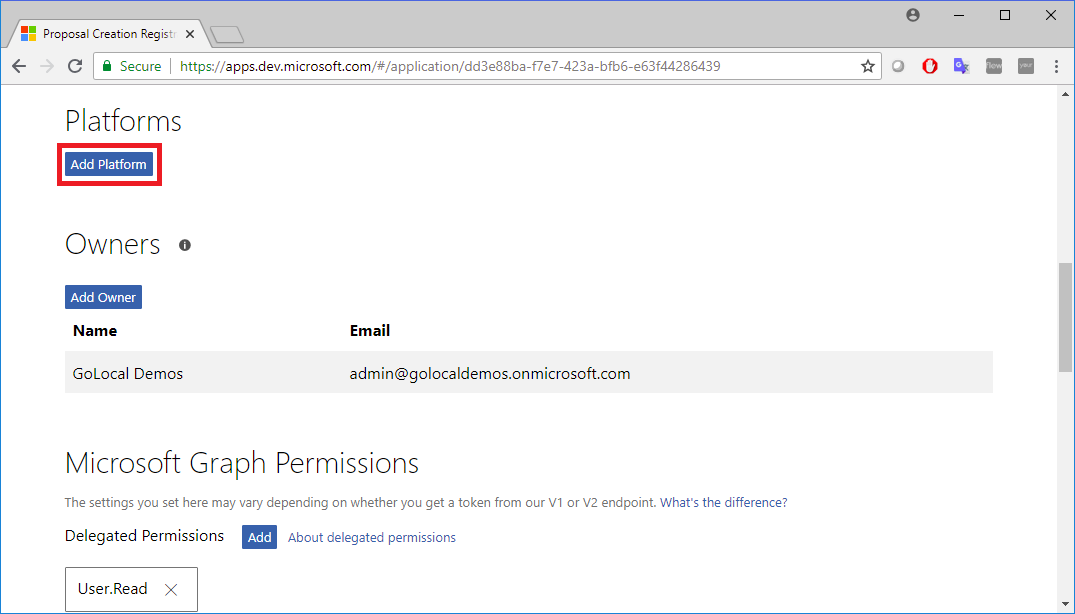
Store the **Application Id** as it is required in the configuration section.



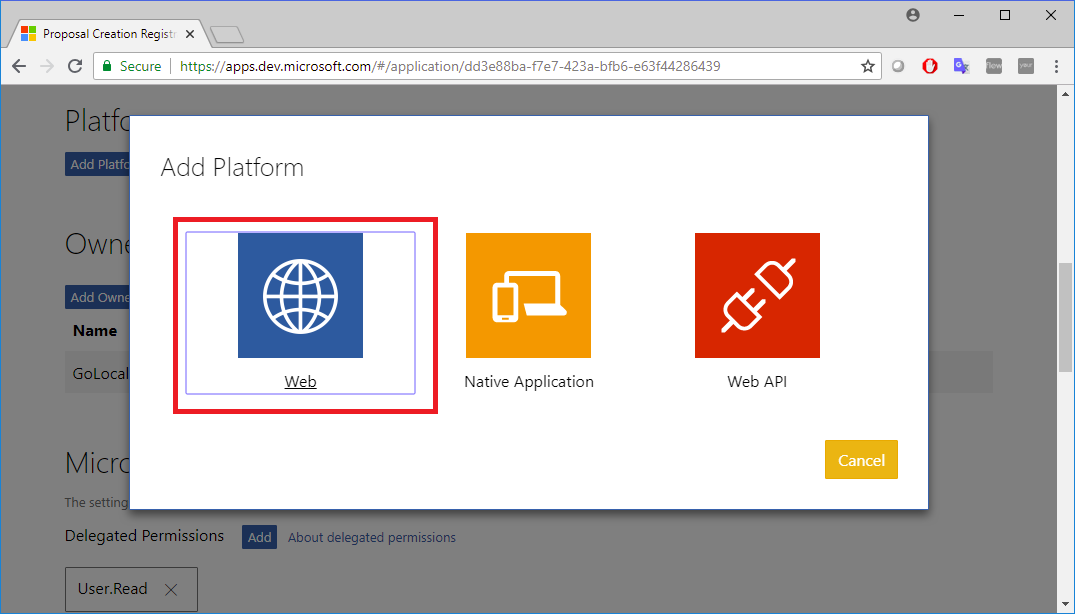
Click on **Generate New Password**.



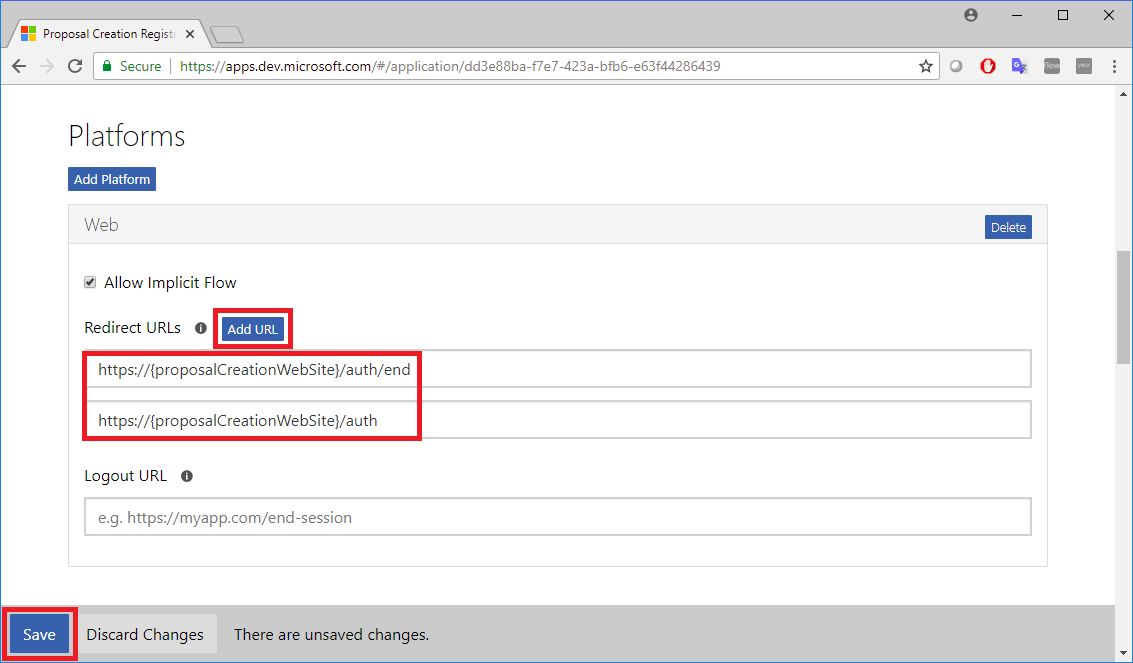
Copy the generated password and store it as it will be required. Then click on **Ok**.



Then click on **Add Platform.**



Choose **Web.**



Add the redirect URLS. Click on **Add URL.**The Proposal Creation Addin requires 2 redirect URLs:

* https://{proposalCreationWebSite}/auth/end
* https://{proposalCreationWebSite}/auth

Identify your Home Page URL, Terms of Service URL and Privacy URL as needed. In the Platform section, add Web API Platform and leave the default values

In the Profile selection, choose the logo that you would like to use and specify the home page URL as the web application URL that you plan to deploy the time tracker solution to



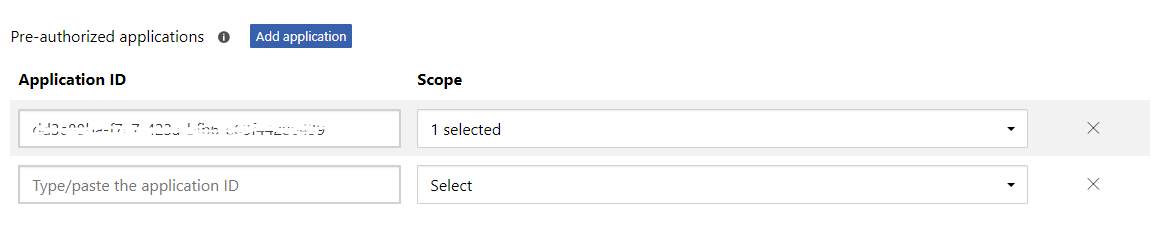
Save Settings and you’re done.

### Update Proposal Manager Registration

Sign into [Microsoft Application Registration portal](https://apps.dev.microsoft.com).

Choose the Proposal Manager Application.

Then pre-authorize the Proposal Creation Addin Application by adding the Application Id previously stored.



## 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 |
| **AzureAd** |
| ClientId | This is the unique ID application of the registered application. | <Application Id stored from the Application Registration portal - apps.dev.microsoft.com> |
| ClientSecret | Client secret for the app registered in tenant | <Secret stored from the Application Registration portal - apps.dev.microsoft.com> |
| Instance | Public Instance name for AAD | "https://login.microsoftonline.com/" |
|  |  |  |
| TenantId | ID of the tenant where the solution is deployed | "<tenant\_id>" |
| GraphScopes | Scope required by the application for accessing Microsoft Graph | “User.Read Sites.Read.All” |
| ProposalManagerApiId | The application id of the Proposal Manager application | <Application Id of the Proposal Manager application registered at apps.dev.microsoft.com> |
| **AppOptions** |
| SiteId | SharePoint site where the application is hosted | “<tenant>.sharepoint.com” |
| ProposalManagerApiUrl | Proposal Manager application url. | "https://proposalmanager.azurewebsites.net” |

### 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

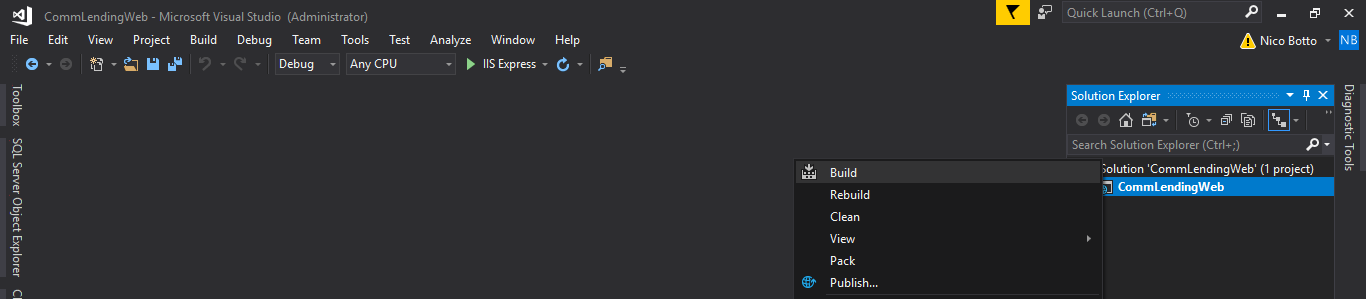
Open the comm-lending-manifest.xml located at .\Manifest and update the following settings:

|  |
| --- |
| <!-- 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.

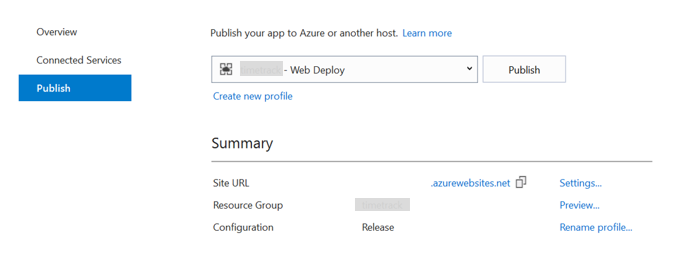
* Select the New Publish profile option and choose Azure App



* 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



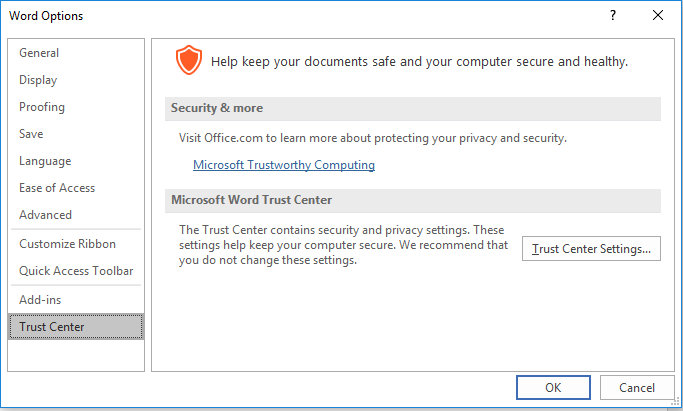
* Click on Publish to deploy the application to the Azure app service

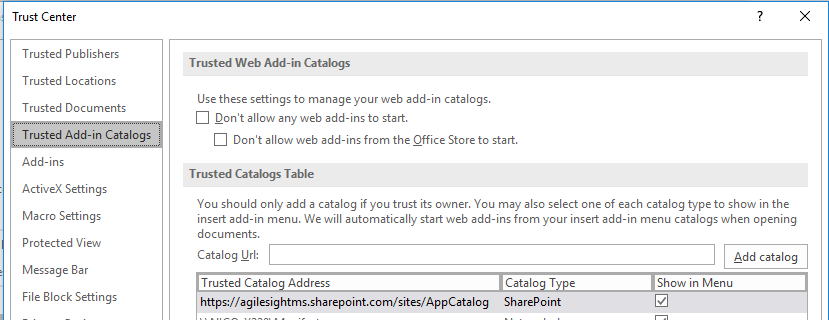
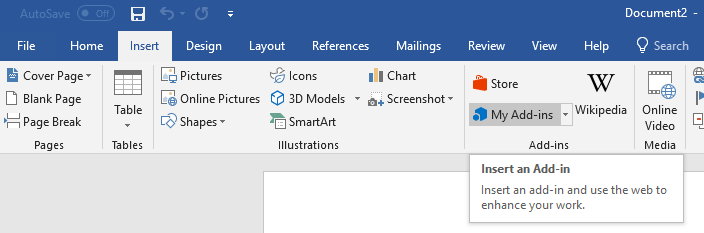
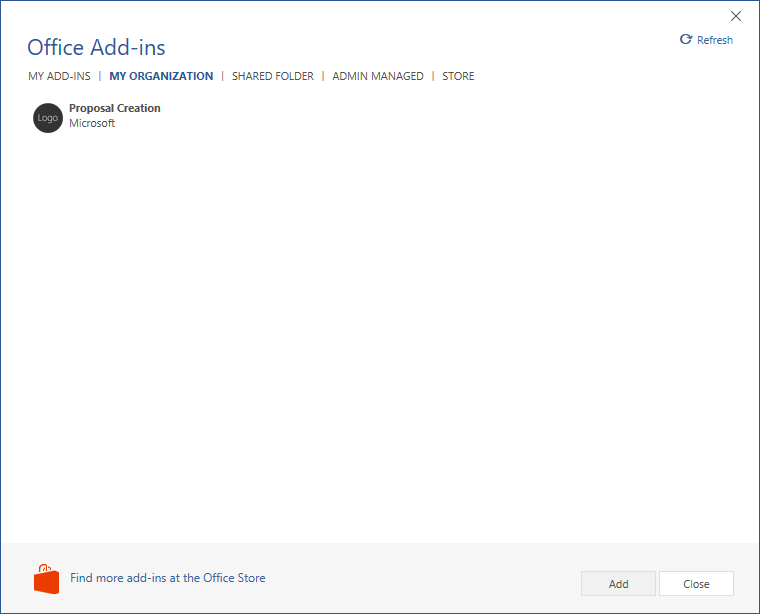


Once publish is completed, the web application is automatically launched on the default browser.

## 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