

# Initial setup - Installation Guide





NTT Data, November 2023



Last updated: November 10<sup>th</sup>, 2023



# CONTENTS

CONTENTS .....	3
TÖGÖ INITIAL SETUP .....	4
Process overview.....	6
Planning.....	8
Preparation.....	11
Deployment.....	14
Preparation.....	14
Before starting.....	14
Get the Azure subscription ID  .....	14
Get the Azure tenant ID  .....	15
Get the Microsoft 365 tenant ID  .....	15
Install additional PowerShell modules .....	16
Set SharePoint term store administrator  .....	18
Install.....	19
Deployment Package Structure.....	19
Configure environment constants.....	20
Advanced usage.....	23
Post deployment actions.....	44
Link luis.ai applications with the Azure Subscription .....	44
User profile custom Properties (Audience Targeting).....	46
Verification .....	50

# TÖGÖ INITIAL SETUP

TÖGÖ is a Social Digital Workplace solution that extends Microsoft 365 capabilities to create an easy-to-use and integrated social experience on top of the Microsoft 365 out-of-the-box suite.

## Goal

The purpose of this document is to provide advice on how to successfully set up a TÖGÖ Fast Track, to help TÖGÖ customers with a streamlined process that defines the required activities and to assist you during the setup process.

## Organization

The structure of this document is based on the following sections:

### TÖGÖ Setup Process Overview

Provides an overview of the NTT DATA setup process, identifying the main activities and the expected results for each activity.

### Planning

This section describes the activities in the planning phase of the setup process. And details the recommended approach for this phase.

### Preparation

Describes the required environment preparation for a TÖGÖ setup, including the required subscription acquisition in Microsoft 365 and Azure, and the required Microsoft Entra ID setup to support access to the required tenant.

### Deployment

Describes the step-by-step process required to deploy a TÖGÖ Fast Track.

# Intended Audience

This document is intended to support the following audiences:

## Project Managers

Assists the project manager in comprehending the necessary steps for a successful deployment of TÖGÖ, including all preparation procedures.

## Operations Managers

Enables the Operations Manager to plan a TÖGÖ deployment and determine the necessary cloud infrastructure platform requirements.

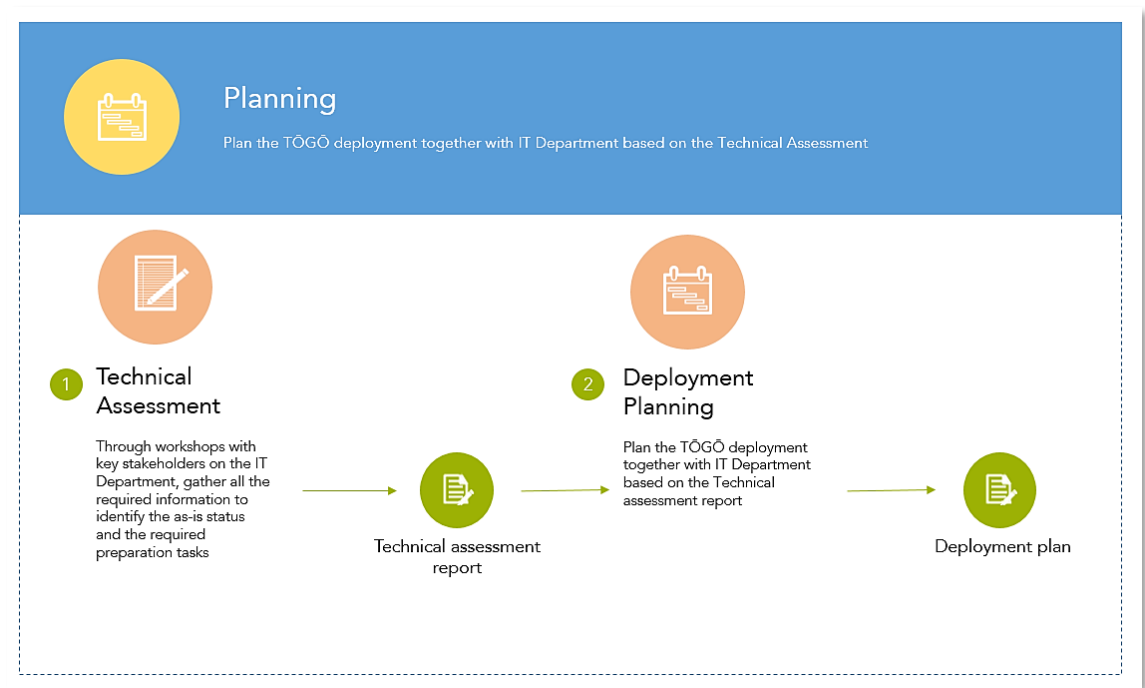
## Operations Team Members

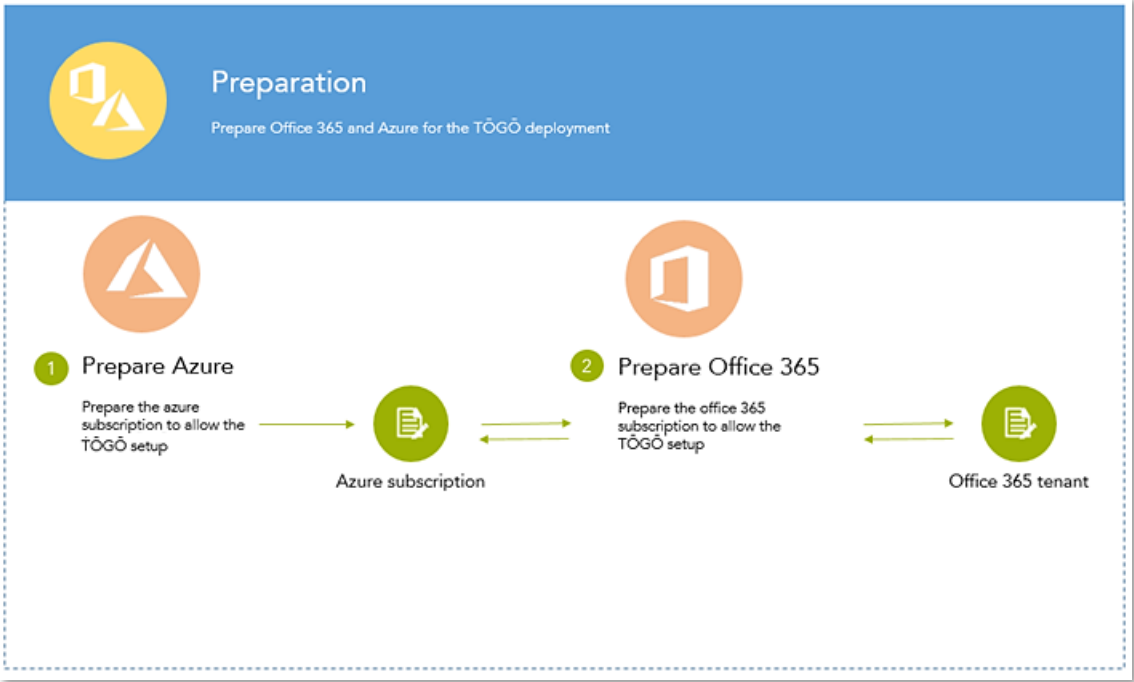
Enables operations team members to set up a TÖGÖ by following the necessary steps described in this document, after obtaining an overview of the mandatory deployment process.

# Process overview

TÖGÖ is a system based on multiple interconnected containers. Each container has its own deployment process and necessary infrastructure. NTT DATA has created a process based on a series of activities to help our customers successfully set up a TÖGÖ Fast Track.

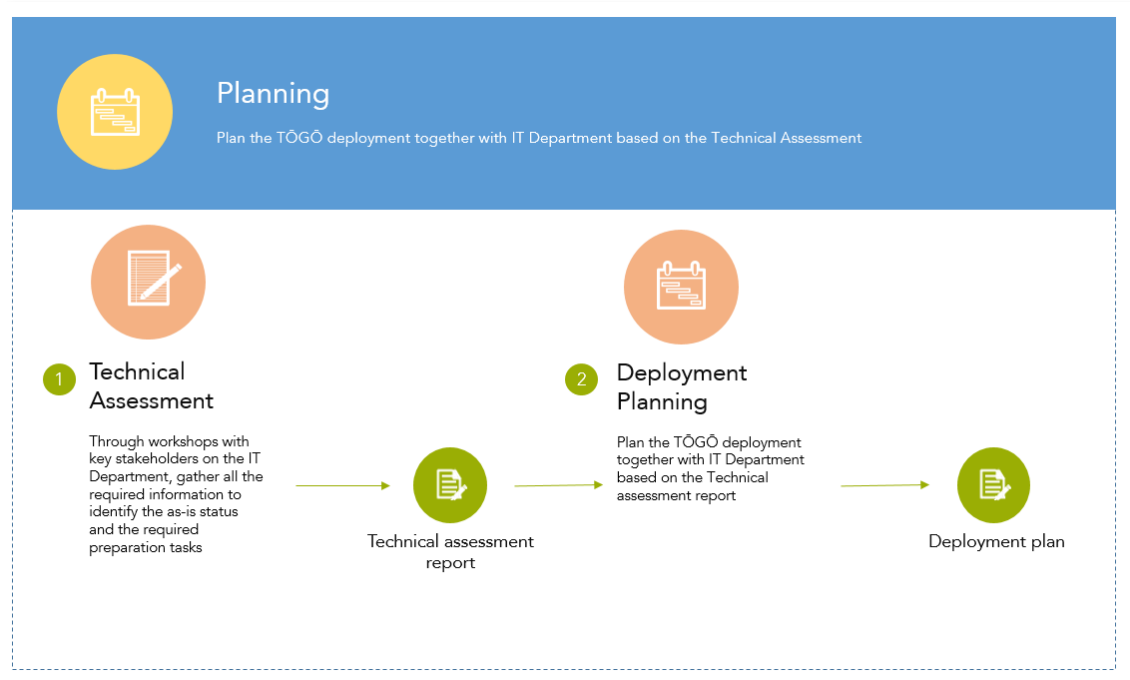
The following diagram outlines the activities required for a successful TÖGÖ setup.





## Planning

Work with the IT department to perform the Technical Assessment and Plan the TÖGÖ deployment.







### Technical Assessment





This activity allows the current as-is situation to be identified from a technical perspective prior to the deployment of TÖGÖ and will identify the preparatory actions required to set up TÖGÖ on the current as-is technical environment.

### Deployment Planning

Using the technical assessment as input, you will need to start creating a deployment plan that includes the required deployment actions. Each action will include the due date and responsible.

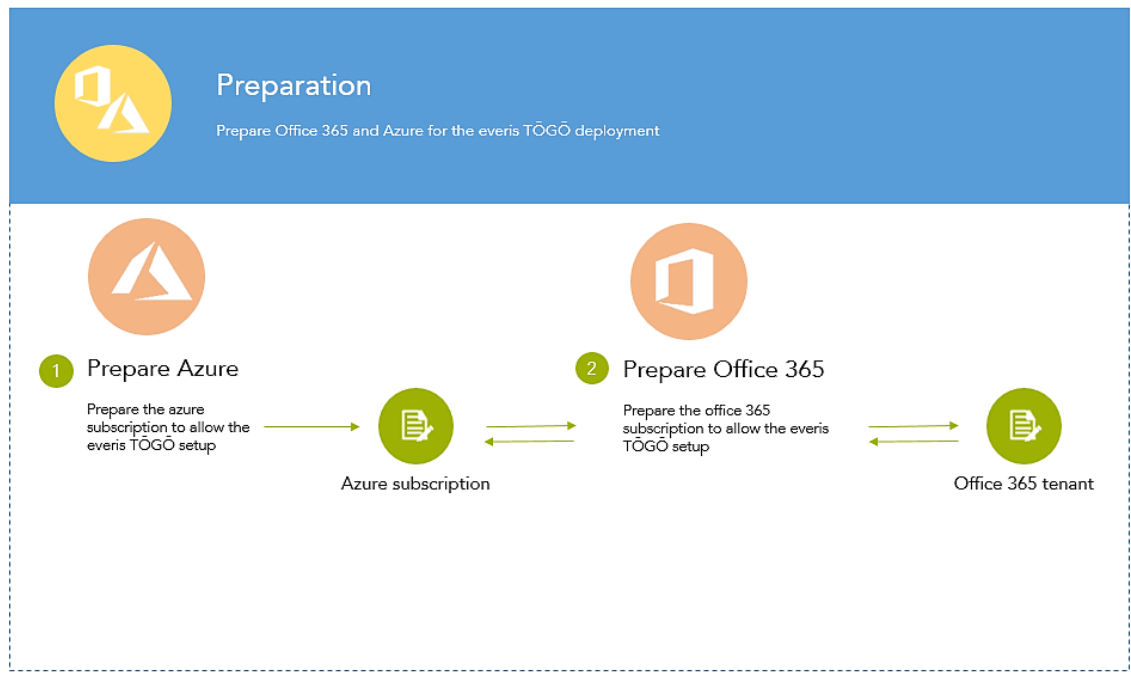


TECHNICAL ASSESMENT	
	<p><b>GOAL</b></p> <p>Identify the organizational, environmental, and technical information that will be required for a successful TÖGÖ setup process.</p>
	<p><b>INPUT</b></p> <p><b>Technical Assessment Questionnaire</b> A questionnaire designed to help you to identify what technical infrastructure and support will be required for the TÖGÖ setup.</p>
	<p><b>TASKS</b></p> <p><b>Technical Assessment</b> Conduct a series of meetings and workshops to gather all the necessary information for the TÖGÖ deployment and assess the current status. This information can be used as input to create a deployment plan.</p> <p><b>Technical Assessment Report Review &amp; Approval</b> Once the technical assessment report is completed, the IT department must review and iterate it as needed, and once it meets the IT department's needs, approve it.</p>
	<p><b>DELIVERABLES</b></p> <p><b>Technical Assessment Report</b> A document that contains the Technical Assessment Report. This report contains the as-is situation before the TÖGÖ deployment and identifies the necessary preparatory actions that will be required before the TÖGÖ deployment.</p>

DEPLOYMENT PLANNING	
	<p><b>GOAL</b></p> <p>Create a deployment plan with an agreed upon timeline, stakeholders responsible for executing the TÖGÖ deployment, including preparation and deployment tasks.</p>
	<p><b>INPUT</b></p> <p><b>Technical Assessment Report</b> A document that contains the Technical Assessment Report. This report contains the as-is situation prior to the TÖGÖ deployment and identifies the required preparatory actions that will be required prior to the TÖGÖ deployment.</p>
	<p><b>TASKS</b></p> <p><b>Create the Deployment Plan</b> Once you have identified all the required tasks, you will need to identify task dependencies, task owners, and stakeholders. Meet with key people to get estimates and agree on a timeline based on their team assignments to start building the plan.</p> <p><b>Review &amp; Approve the Deployment Plan</b> Once the deployment plan is complete, share it with all stakeholders and owners for review and approval if no further iterations of the document are required.</p>
	<p><b>DELIVERABLES</b></p> <p><b>Deployment plan</b> A document containing the deployment plan agreed upon by all stakeholders that will guide the next two phases.</p>

## Preparation

Perform the preparation actions as per the deployment plan schedule. These actions are specific to each environment and will be handled as per the deployment plan schedule.











### Prepare Azure

Execute the necessary steps to create an Azure subscription that permits deployment of TÖGÖ Azure PaaS services.

### Prepare Microsoft 365

Execute the necessary steps to configure a Microsoft 365 tenant that enables TÖGÖ SharePoint deployment.

PREPARE AZURE	
	<p>GOAL</p> <p>Ensure a properly configured Azure subscription, including synchronization of Microsoft 365 Tenant Microsoft Entra IDs.</p>
	<p>INPUT</p> <p><b>Deployment Plan</b> A document that lists the required Azure preparation tasks and the schedule for performing these tasks.</p>
	<p>TASKS</p> <p><b>Execute deployment plan Azure preparation tasks</b> Perform the Azure preparation tasks identified in the deployment plan.</p> <p><b>Control and monitor</b> Depending on the Deployment Plan, complexity control and monitoring are key tasks for a successful deployment. During this phase, we plan to identify technical and coordination issues that may need to be addressed during the deployment and adjust the Deployment Plan schedule accordingly.</p>
	<p>DELIVERABLES</p> <p><b>Azure Subscription</b> An Azure Subscription setup as required for the TÖGÖ Fast Track deployment.</p>

PREPARE MICROSOFT 365	
	<p><b>GOAL</b></p> <p>Create the minimal infrastructure in Microsoft 365 SharePoint Online to enable TÖGÖ deployment.</p>
	<p><b>INPUT</b></p> <p><b>Deployment Plan</b> A document that outlines the required Microsoft 365 preparation tasks and the timeline for completing those tasks.</p>
	<p><b>TASKS</b></p> <p><b>Execute the Microsoft 365 preparation tasks in the deployment plan</b> Perform the Microsoft 365 preparation tasks identified in the deployment plan.</p> <p><b>Control and monitor</b> Depending on the deployment plan, complexity control and monitoring is a key task for a successful deployment. During this phase, we plan to identify technical and coordination issues that can be adjusted as we go along and adjust the Deployment Plan schedule accordingly.</p>
	<p><b>DELIVERABLES</b></p> <p><b>Microsoft 365 Tenant</b> A Microsoft 365 tenant with the required service subscriptions and environment ready to set up TÖGÖ.</p>

# Deployment


The following section describes the steps required to complete a TÖGÖ Fast Track installation.

## Preparation

### Before starting

Before you begin, you will need:

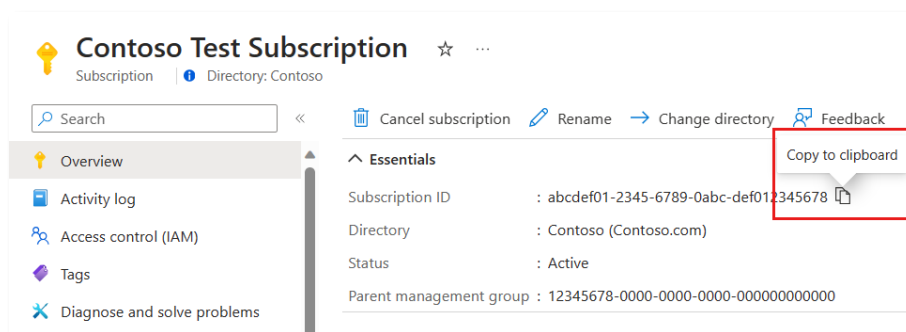
1. The TÖGÖ Deployment Package.
2. A computer with:
  - a. Internet access to the Azure subscription and Microsoft 365 tenant subscription.
  - b. PowerShell installed.
  - c. A Microsoft 365 tenant with an app catalog site created to allow you to deploy the TÖGÖ SPFx solutions.

 **Note:** the recommended name for the application catalog site is **AppCatalog**.

### Get the Azure subscription ID

Follow these steps to retrieve the ID for a subscription in the Azure portal.

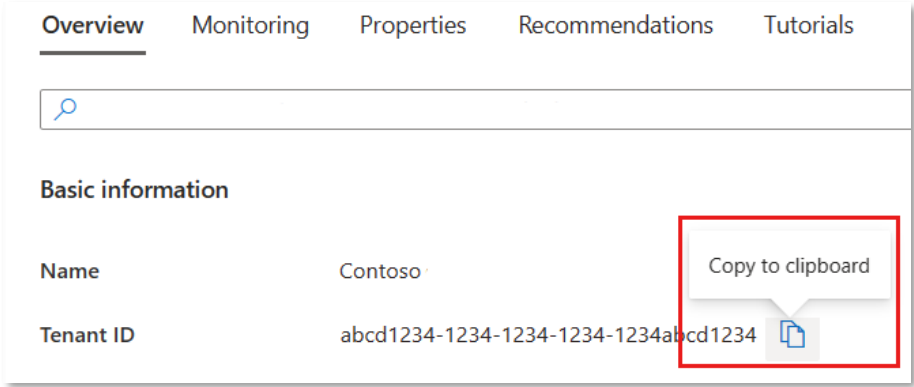
1. Login to [Azure portal](#) with an Azure subscription administration user.
2. Click on the **Subscriptions** menu item.
3. Find the **Subscription ID** in the **Overview** section.



## Get the Azure tenant ID

Follow these steps to retrieve the ID for a Microsoft Entra tenant in the Azure portal.

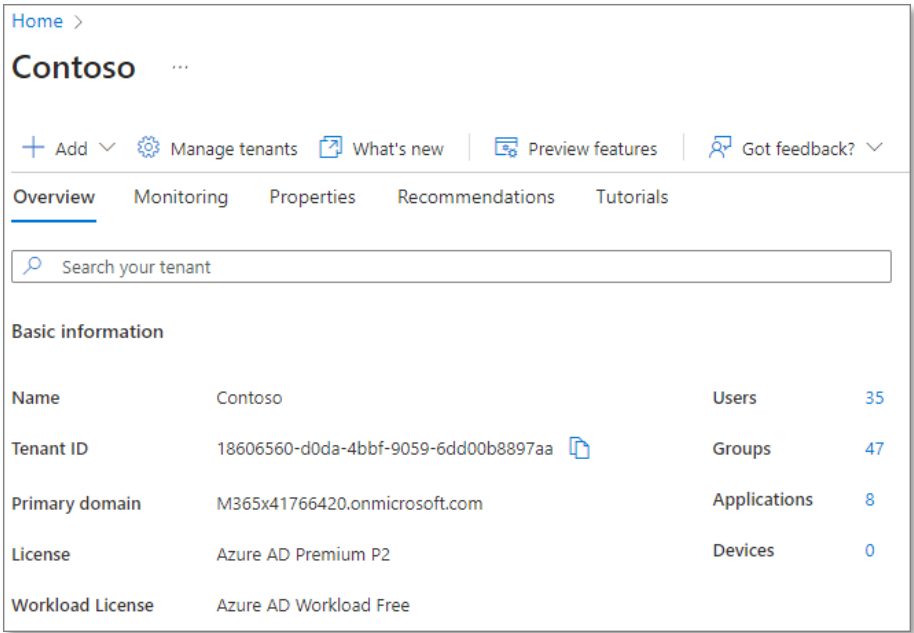
- 1. Login to [Azure portal](#) with a subscription administration user.
- 2. Click on the **Microsoft Entra ID** menu item.
- 3. Find the **Tenant ID** in the **Overview** section.



## Get the Microsoft 365 tenant ID

Following are the steps required to obtain the Azure subscription id:

- 1. Login to [Microsoft Entra admin center](#) with a Microsoft 365 subscription administration user.
- 2. Find the **Tenant ID** in the **Overview** section.



## Install additional PowerShell modules

All scripts must be run with PowerShell 5.0 version on the local machine.

The following are the PowerShell modules required during the installation process:

- Node.js
- Bicep
- PnP PowerShell
- Azure Az
- Azure CLI
- Azure AD
- Azure Bot Service
- M365 CLI
- Moline
- Open SSL

Here is how to check that the modules are present, and how to install them on the local machine that will be used during the installation. Although the installation will check that everything is correct, it is recommended that you carry out these steps first.

### Node.js

To check if it exists, we will use


```
node -v
```

If it does not exist, we need to download and install it from here [Node.js](#). We recommend that you download the LTS version

### Bicep

To check if it exists, we will use

```
az bicep version
```

 **Note:** the minimum required version is 0.14.46

If it does not exist, we need to download and install it from here [Bicep](#). If you are not using the Windows x64 version, you can find all versions [here](#).

### PnP PowerShell

To check if it exists, we will use

```
Get-Module -ListAvailable -Name PnP.PowerShell
```

 **Note:** the required version is 1.12.0

If it does not exist, we will use to install it

```
Install-Module PnP.PowerShell -RequiredVersion 1.12.0 -Force
```



### Azure Az

To check if it exists, we will use

```
Get-InstalledModule -Name Az
```

 **Note:** the required version is 9.3.0


If it does not exist, we will use to install it

```
Install-Module -Name Az -RequiredVersion 9.3.0 -Force
```

### Azure CLI

To check if it exists, we will use

```
az version
```

 **Note:** the minimum required version is 2.40.0

If it does not exist, we need to download and install it from here [Azure CLI](#). If you are not using the Windows version or for more information on how to install it, you can access [here](#).

### Azure AD

To check if it exists, we will use

```
Get-Module -ListAvailable -Name AzureAD
```


If it does not exist, we will use to install it

```
Install-Module AzureAD -AllowClobber
```

### Azure Bot Service

To check if it exists, we will use

```
Get-InstalledModule -Name Az.BotService
```

 **Note:** the minimum required version is 0.30.0


If it does not exist, we will use to install it

```
Install-Module -Name Az.BotService -RequiredVersion 0.3.0 -Force
```

### M365 CLI

To check if it exists, we will use

```
m365 version
```

 **Note:** the minimum required version is 5.8.0

If it does not exist, we will use to install it

```
npm i -g @pnp/cli-microsoft365
```

### MSOnline

To check if it exists, we will use

```
Get-Module -ListAvailable -Name MSOnline
```

If it does not exist, we will use to install it

```
Install-Module MSOnline
```

### Open SSL

To check if it exists, we will use

```
openssl version
```

If it does not exist, we will use to install it

```
Install-Module -Name OpenSSL
```

## Set SharePoint term store administrator







Follow these steps to add the user who will be used during the installation as the term store administrator.

1. In the SharePoint admin center, under **Content services**, select [Term store](#).
2. In the tree view pane on the left, select the taxonomy.
3. In the **Term store** page, for **Admins**, select **Edit**. The **Edit term store admin** panel appears. Enter the names or email addresses of the people who you want to add as term store admins. Select **Save**.

# Install

## Deployment Package Structure

The TÖGÖ Deployment Package contains the following files and folders.

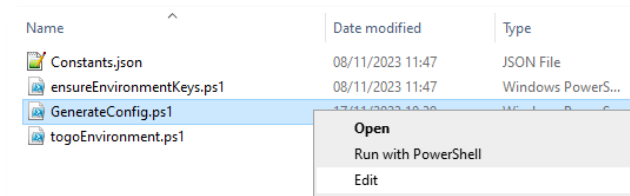
Name	Date modified	Type
 <b>aad</b>	08/11/2023 11:47	File folder
 <b>backend</b>	08/11/2023 11:47	File folder
 <b>env</b>	08/11/2023 11:47	File folder
 <b>iac</b>	08/11/2023 11:47	File folder
 <b>scripts</b>	08/11/2023 11:47	File folder
 <b>sp</b>	08/11/2023 11:47	File folder

- **aad**, folder containing the support scripts and files required during the deployment process to create the app registrations in the Microsoft Entra ID.
- **backend**, folder containing the support scripts and files required during the deployment process to set the binaries for the TÖGÖ app services.
- **env**, folder containing the support scripts and files required during the deployment process to manage the environment constants.
- **iac**, folder containing the support scripts and files required during the deployment process to provision the TÖGÖ resources.
- **scripts**, folder containing the support scripts and files required to start the deployment process.
- **sp**, folder containing the support scripts and files required during the deployment process to configure the SharePoint resources and upload the TÖGÖ solution to the app catalog.

## Configure environment constants

In this first step, you will configure the environment constants associated with your Microsoft 365 and Azure subscription.

We will do this with the help of the **GenerateConfig.ps1** file, located in the **env** folder, using the Windows PowerShell ISE, as shown next. Once the data is entered, it is necessary to save the file and run the script.



### GenerateConfig.ps1

```

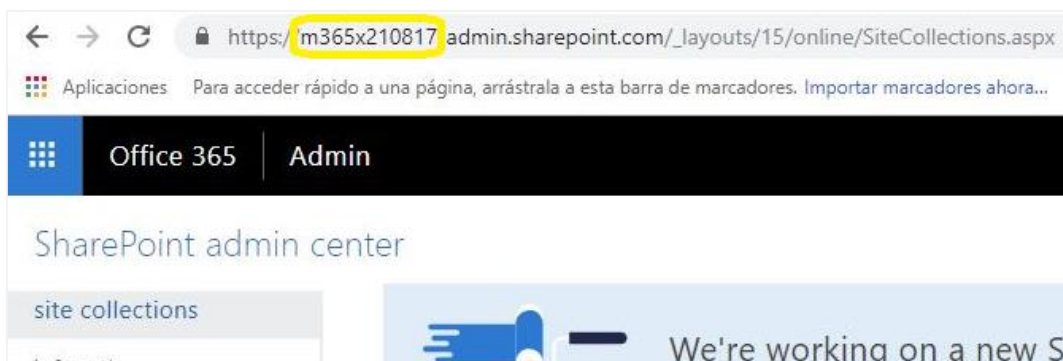
1 param (
2     [Parameter(Mandatory = $true, HelpMessage = "Tenant name, which corresponds to the prefix of the SharePoint URL. Eg: For 'https://everitogopre.s
3     [string]$tenantName,
4     [Parameter(Mandatory = $true, HelpMessage = "Azure Subscription identifier")]
5     [string]$azSubscriptionId,
6     [Parameter(Mandatory = $true, HelpMessage = "O365 Tenant identifier")]
7     [string]$o365TenantId,
8     [Parameter(Mandatory = $true, HelpMessage = "Azure Tenant identifier")]
9     [string]$azTenantId,
10    [Parameter(Mandatory = $true, HelpMessage = "O365 Administrator's email")]
11    [string]$o365AdminMail,
12    [Parameter(Mandatory = $true, HelpMessage = "Azure Administrator's email")]
13    [string]$azureAdminMail,
14    [Parameter(Mandatory = $true, HelpMessage = "Managed path. Possible values: sites, teams")]
15    [string]$managedPath,
16    [Parameter(Mandatory = $true, HelpMessage = "Resource group name")]
17    [string]$rgName,
18    [Parameter(Mandatory = $true, HelpMessage = "Resource group location")]
19    [string]$rgLocation,
20    [Parameter(Mandatory = $true, HelpMessage = "Open Api Resource group name")]
21    [string]$openApiRgName,
22    [Parameter(Mandatory = $true, HelpMessage = "Open Api Resource group location")]
23    [string]$openApiRgLocation,
24    [Parameter(Mandatory = $true, HelpMessage = "Custom audience targeting. If true, Togo audience targeting will be applied, otherwise SharePoint
25    [string]$customAudience,
26    [Parameter(Mandatory = $true, HelpMessage = "Overwrite addons. If true, Togo addons will be overwritten, otherwise addons will not be modified")
27    [string]$overwriteAddonDefinitions,
28    [Parameter(Mandatory = $true, HelpMessage = "Togo User Profile Synchronization. If true, Togo user profile linked with external system, otherwis
29    [bool]$togoUPEnabled,
30    [Parameter(Mandatory = $true, HelpMessage = "Togo User Profile Synchronization. If true, Togo user profile will be sync in SharePoint user profi
31    [bool]$togoSharePointUPSyncEnabled,
32    [Parameter(Mandatory = $true, HelpMessage = "Generate self-signed certificate. If true, the process will generate self-signed certificate, other
33    [string]$generateSelfSignedCertificate,
34    [Parameter(Mandatory = $true, HelpMessage = "Togo version you want to install")]
35    [string]$togoVersion
36 )
37
38
39 $environmentConfig = [ordered]@{
40     tenantPropTogoVersionValue = $togoVersion;
41     tenantPropTogoLocalesValue = ["en-US", "es-ES"];
42     managedPath = $managedPath;
43     envM365TenantId = $o365TenantId;
44     envM365TenantName = $tenantName;

```

When this script is run, you will be prompted for the following parameters:

**tenantName**

Specify the **name of the tenant** that matches the prefix of the SharePoint URL.



**azSubscriptionId**

Specify the **Subscription ID** obtained in the *Get the Azure subscription ID* section.

**o365TenantId**

Specify the **Tenant ID** obtained in the *Get the Microsoft 365 tenant ID* section.

**azTenantId**

Specify the **Tenant ID** obtained in the *Get the Azure tenant ID* section.

**o365AdminMail**

The email address of the Microsoft 365 tenant administrator. It is recommended that this user be the Global Admin in Microsoft 365, because this user must have administrator permissions to the Term Store, edit permissions to the SharePoint App Catalog, and permissions to the Microsoft Entra ID to create enterprise registrations.

**azureAdminMail**

The email address of the Azure tenant administrator. This user must have permissions to create resources and register providers within the subscription.

**managedPath**

This managed path is used to create the TÖGÖ site. The possible values are:

**sites**, <https://<tenantname>.sharepoint.com/sites/togo>

**teams**, <https://<tenantname>.sharepoint.com/teams/togo>

which are the managed paths that SharePoint Online provides by default.

**rgName**

This is the name of the resource group where all TÖGÖ resources will be installed. For example, *RG\_TOGO\_PRO*.

**rgLocation**

This is the name of the location where all TÖGÖ resources will be installed. For example, *West Europe*.

**openApiRgName**

This is the name of the resource group where TÖGÖ Open API resources will be installed. For example, *RG\_TOGO\_PRO*.

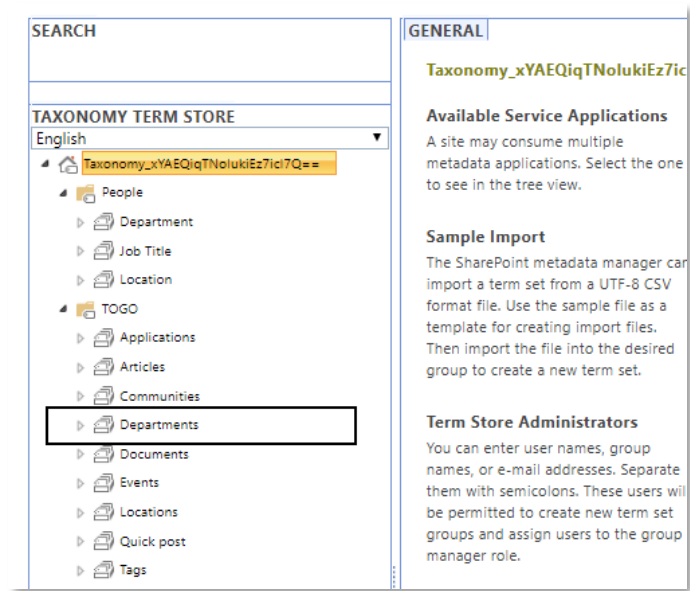
**openApiRgLocation**

This is the name of the location where TÖGÖ Open API resources will be installed. For example, *West Europe*.

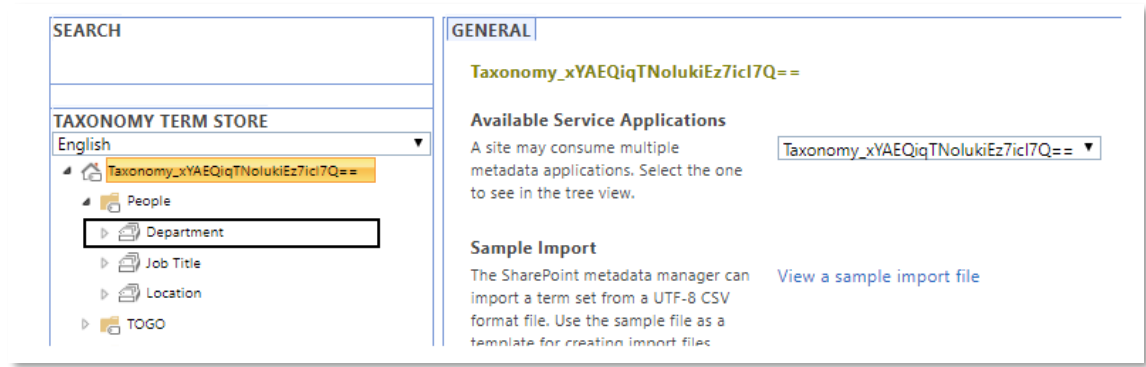
customAudience

This parameter indicates whether you want to use custom audience targeting. Possible values are true or false.

If **\$true**, TÖGÖ audience targeting is used. The installation process creates a custom term set to manage the departments and locations used in the Audience Targeting and Departments and Locations filter values.



If **\$false**, it uses SharePoint OOTB.



overwriteAddonDefinitions

This parameter indicates whether the addon configuration should be overwritten. Possible values are true or false. If **\$true**, TÖGÖ addons are overwritten. If **\$false**, TÖGÖ addons are not changed.

togoUPEnabled

As part of TÖGÖ Audience, we allow you to decide whether to synchronize the user's profile with Microsoft Entra ID or with an external system. Possible values are true or false. If **\$true**, TÖGÖ user profile will be linked with external system. If **\$false**, we will use synchronization with Microsoft Entra ID.

### togoSharePointUPSyncEnabled

As part of TÖGÖ Audiences, we allow you to decide whether you want to synchronize the user's profile with the SharePoint Online user profile. Possible values are true or false. If **\$true**, TÖGÖ user profile will be sync in SharePoint User Profile. If **\$false**, it will not.

### generateSelfSignedCertificate

During the installation process, we need a certificate for some Azure resources. Possible values are true or false. If **\$true**, the process will generate self-signed certificate. If **\$false**, it will be required to upload one to Key Vault.

### togoVersion

Specify the version of TÖGÖ you want to install. For example, *1.9.0*

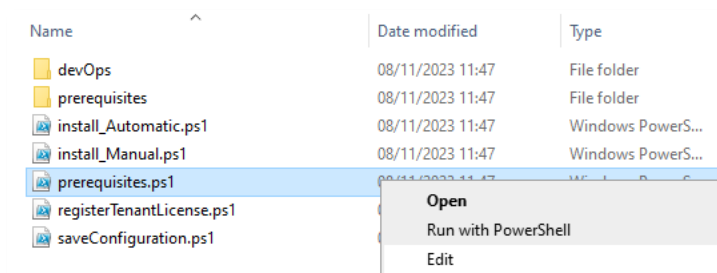
## Prerequisites

In the next step, we will install all the prerequisites needed to deploy TÖGÖ. This step requires the presence of the Azure and Microsoft 365 administrators that were specified in the generation of the constant file during the previous step, as they will be asked for their credentials during the process.

**Note:** it is recommended that this user be a Global Admin in Microsoft365. Review section **o365AdminMail** of the previous step.

**Note:** it is recommended that this user be an Admin in the Azure subscription. Review section **azureAdminMail** of the previous step.

We will do this by using the **prerequisites.ps1** file, which is in the **scripts** folder, and by using the Windows PowerShell ISE, as shown next.



### prerequisites.ps1

```
prerequisites.ps1 X
18 Write-Host -nonewline "Importing Azure Powershell modules..." -ForegroundColor Yellow
19 Import-Module -Name Az
20 Disable-AzDataCollection
21 Write-Host -nonewline "Done!" -ForegroundColor Green
22 $ErrorActionPreference = "Stop"
23 try {
24     Write-Host ""
25     Write-Host -nonewline "Check and execute installation prerequisites, Continue? (Y/N)" -ForegroundColor Yellow
26     $response = read-host
27     if ($response -ne "Y") { exit }
28
29     Write-Host -nonewline "Please, provide your M365 Tenant Name: " -ForegroundColor Yellow
30     $m365TenantName = read-host
31     if (($null -eq $m365TenantName) -or ($m365TenantName -eq '')) { exit }
32
33     $scriptPath = $PSScriptRoot
34     $directorySeparator = [System.IO.Path]::DirectorySeparatorChar
35     $spScriptPath = "$scriptPath$directorySeparator..$directorySeparator$sp$directorySeparator$scripts$directorySeparator"
36     $envPath = "$scriptPath$directorySeparator..$directorySeparator$env$directorySeparator"
37     $spPath = "$scriptPath$directorySeparator..$directorySeparator$sp$directorySeparator"
38     $spScriptsPath = "$spPath$scripts$directorySeparator"
39     $prerequisitesPath = "$scriptPath$directorySeparator$prerequisites$directorySeparator"
40     $keyVaultPath = "$scriptPath$directorySeparator..$directorySeparator$iac$directorySeparator$scripts$directorySeparator$res"
41     $keyVaultCertificatePath = "$prerequisitesPath$createCertificate.ps1"
```

This is necessary to verify that the execution folder is the same as the folder that contains the prerequisites.ps1 script.

```
PS C:\TÖGÖ - Release - v1.9.0\scripts>
```

When everything is ready and checked, you **can run the script**. You need to follow the steps that appear in the console. There are several cases, such as entering user credentials, accessing a SharePoint configuration page, or simply waiting for the step to complete. The steps are detailed below.

### Tenant name

The first parameter the script asks for is the tenant's name, that matches the prefix of the SharePoint URL.

```
Please, provide your M365 Tenant Name: tenantname
```

### Check PowerShell modules

The first step is to verify that all the PowerShell modules are installed correctly. If any modules are missing, the script tells you what to do to install them. You can also refer to section [Install additional PowerShell modules](#).

```
Checking installed powershell modules...
Check PowerShell version... Done

Check if Node.js and NPM are installed... Done
Check if Az modules are installed...Done
Check if module PnP PowerShell Online is installed... Done

Check if module Azure CLI 2.0 is installed... Done
Check if module AzureAD is installed... Done

Name          Version
----          -
PnP.PowerShell 1.12.0
AzureAD        2.0.2.118
Check if Az.BotService is installed... Done

Check if Bicep is installed... Bicep Version: Bicep CLI version 0.11.1 (030248df55)
Done

Check if module M365 CLI is installed...Module M365 CLI does not exist, execute the install command: npm i -g @pnp/cli-microsoft365
If m365 command is not recognized, add the following route to the path: C:\users\{Your User}\AppData\Roaming\npm
Check if module MSOnline is installed...Done

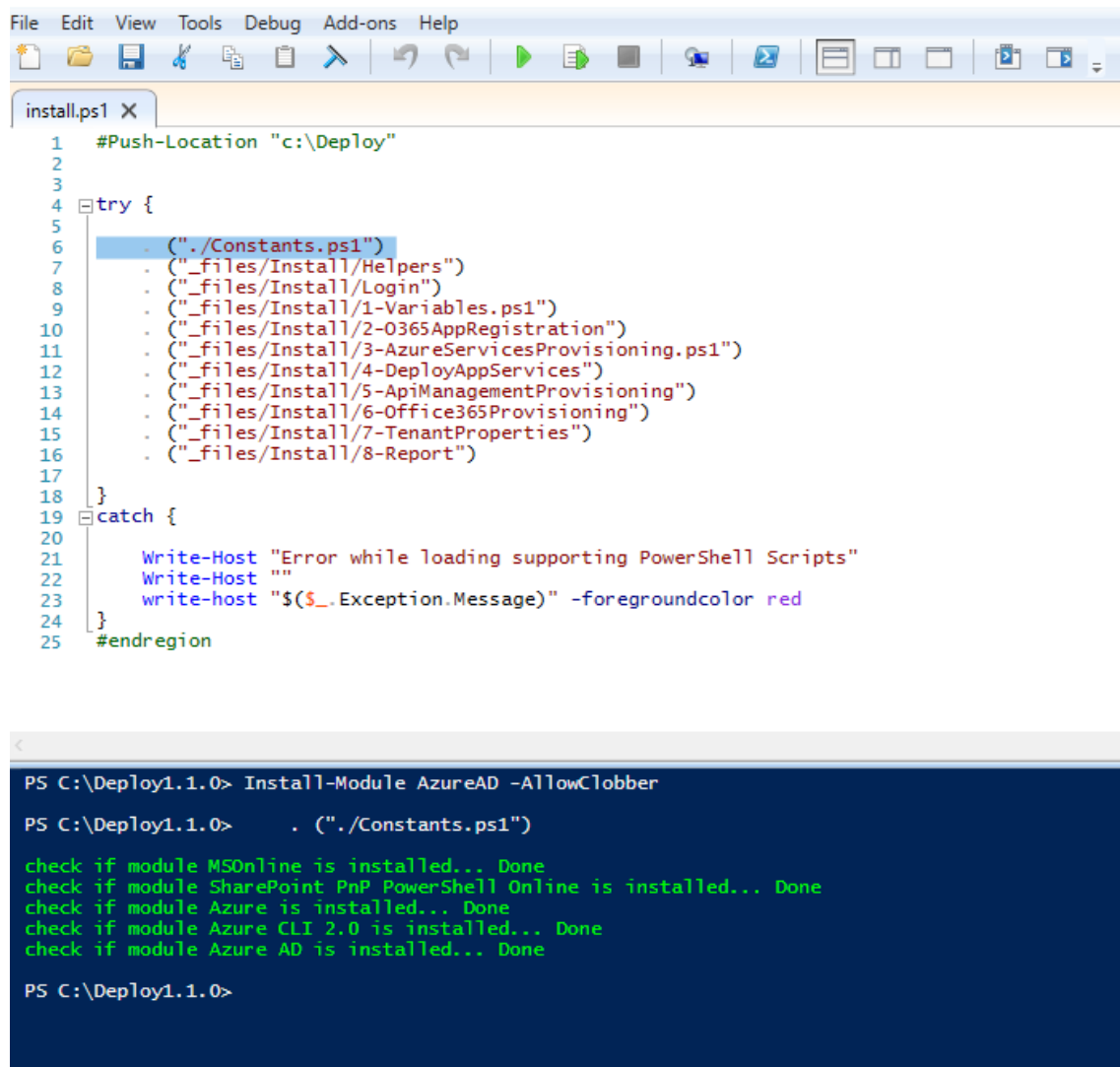
MSOnline       1.1.18...
Check if module OpenSSL is installed...Done
```

### Load environment constants

Next, the script initializes the environment constants with the tenant name. It will first try to retrieve the configuration stored in Azure Storage. If it does not exist, it will retrieve the constant file that we created in the previous steps.

```
Retrieving TÖGÖ environment settings for 'tenantName'...
Done!
```





The image shows a PowerShell script editor window with a file named 'install.ps1'. The script contains a try-catch block that attempts to load several PowerShell scripts from a local directory. The scripts listed are: Constants.ps1, Helpers.ps1, Login.ps1, 1-Variables.ps1, 2-0365AppRegistration.ps1, 3-AzureServicesProvisioning.ps1, 4-DeployAppServices.ps1, 5-APIManagementProvisioning.ps1, 6-Office365Provisioning.ps1, 7-TenantProperties.ps1, and 8-Report.ps1. If an error occurs, the script writes an error message to the console.

```

1 #Push-Location "c:\Deploy"
2
3
4 try {
5
6     . ("./Constants.ps1")
7     . ("_files/Install/Helpers")
8     . ("_files/Install/Login")
9     . ("_files/Install/1-Variables.ps1")
10    . ("_files/Install/2-0365AppRegistration")
11    . ("_files/Install/3-AzureServicesProvisioning.ps1")
12    . ("_files/Install/4-DeployAppServices")
13    . ("_files/Install/5-APIManagementProvisioning")
14    . ("_files/Install/6-Office365Provisioning")
15    . ("_files/Install/7-TenantProperties")
16    . ("_files/Install/8-Report")
17
18 }
19 catch {
20
21     Write-Host "Error while loading supporting PowerShell Scripts"
22     Write-Host ""
23     Write-Host "$($_.Exception.Message)" -foregroundcolor red
24 }
25 #endregion

```

Below the script editor, a terminal window shows the execution of the script. The user runs 'Install-Module AzureAD -AllowClobber' and then executes the script. The output shows several checks for required modules, all of which are successful.

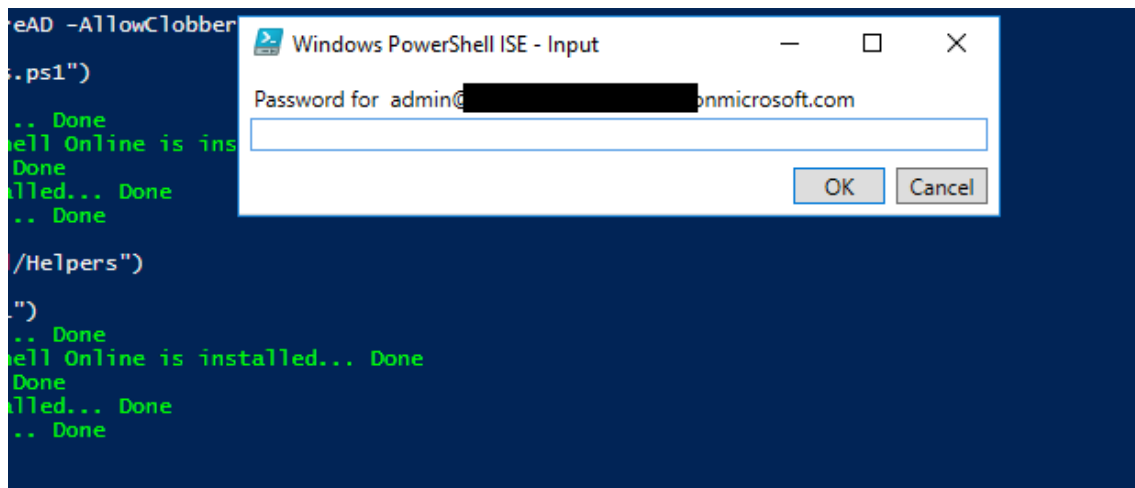
```

PS C:\Deploy1.1.0> Install-Module AzureAD -AllowClobber
PS C:\Deploy1.1.0> . ("./Constants.ps1")
check if module MSOnline is installed... Done
check if module SharePoint PnP PowerShell Online is installed... Done
check if module Azure is installed... Done
check if module Azure CLI 2.0 is installed... Done
check if module Azure AD is installed... Done
PS C:\Deploy1.1.0>

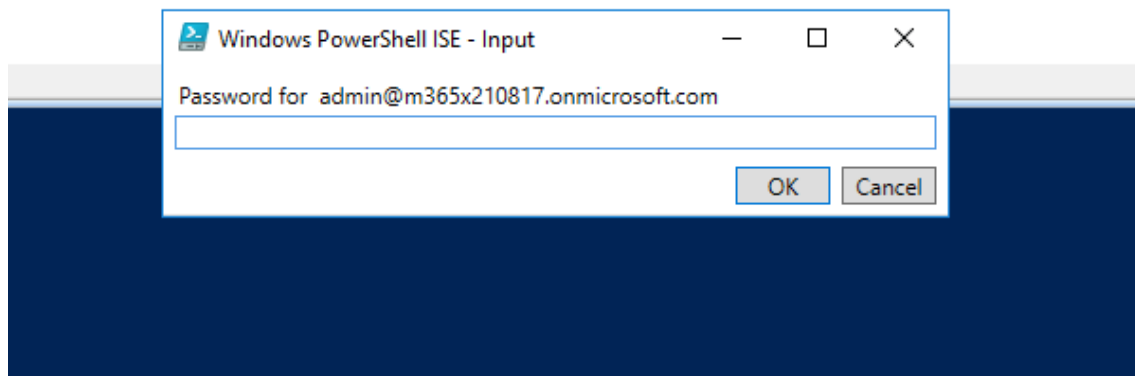
```

- Execute command line . ("\_files/Install/Constants.ps1"):
- Execute command line . ("\_files/Install/Helpers.ps1"):
- Execute command line . ("\_files/Install/Login.ps1"):

In this step the script will prompt for the password of the Azure admin Account

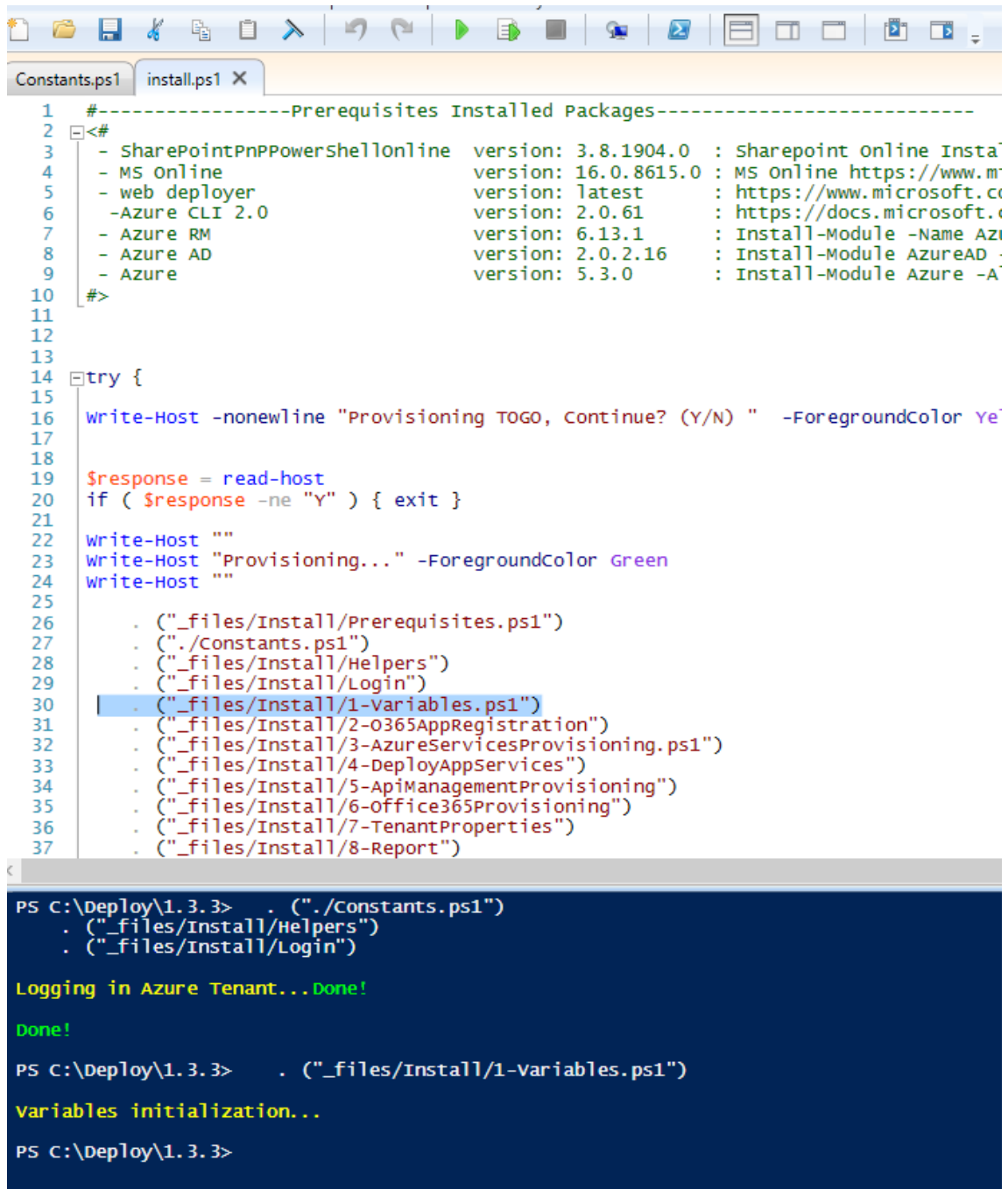


And the password of the O365 Tenant Admin



If the credentials are correct the script output the message “ Done!”

- Execute command line `. ("_files/Install/1-Variables.ps1")`



```

1  #-----Prerequisites Installed Packages-----
2  <#
3  - SharePointPnPPowerShellOnline version: 3.8.1904.0 : Sharepoint Online Insta
4  - MS Online version: 16.0.8615.0 : MS Online https://www.m
5  - web deployer version: latest : https://www.microsoft.co
6  - Azure CLI 2.0 version: 2.0.61 : https://docs.microsoft.co
7  - Azure RM version: 6.13.1 : Install-Module -Name Azi
8  - Azure AD version: 2.0.2.16 : Install-Module AzureAD
9  - Azure version: 5.3.0 : Install-Module Azure -A
10 #>
11
12
13
14 try {
15
16     Write-Host -nonewline "Provisioning TOGO, Continue? (Y/N) " -ForegroundColor Ye
17
18
19     $response = read-host
20     if ( $response -ne "Y" ) { exit }
21
22     Write-Host ""
23     Write-Host "Provisioning..." -ForegroundColor Green
24     Write-Host ""
25
26     . ("_files/Install/Prerequisites.ps1")
27     . ("./Constants.ps1")
28     . ("_files/Install/Helpers")
29     . ("_files/Install/Login")
30     . ("_files/Install/1-Variables.ps1")
31     . ("_files/Install/2-0365AppRegistration")
32     . ("_files/Install/3-AzureServicesProvisioning.ps1")
33     . ("_files/Install/4-DeployAppServices")
34     . ("_files/Install/5-ApiManagementProvisioning")
35     . ("_files/Install/6-Office365Provisioning")
36     . ("_files/Install/7-TenantProperties")
37     . ("_files/Install/8-Report")

```

```

PS C:\Deploy\1.3.3> . ("./Constants.ps1")
. ("_files/Install/Helpers")
. ("_files/Install/Login")

Logging in Azure Tenant...Done!

Done!

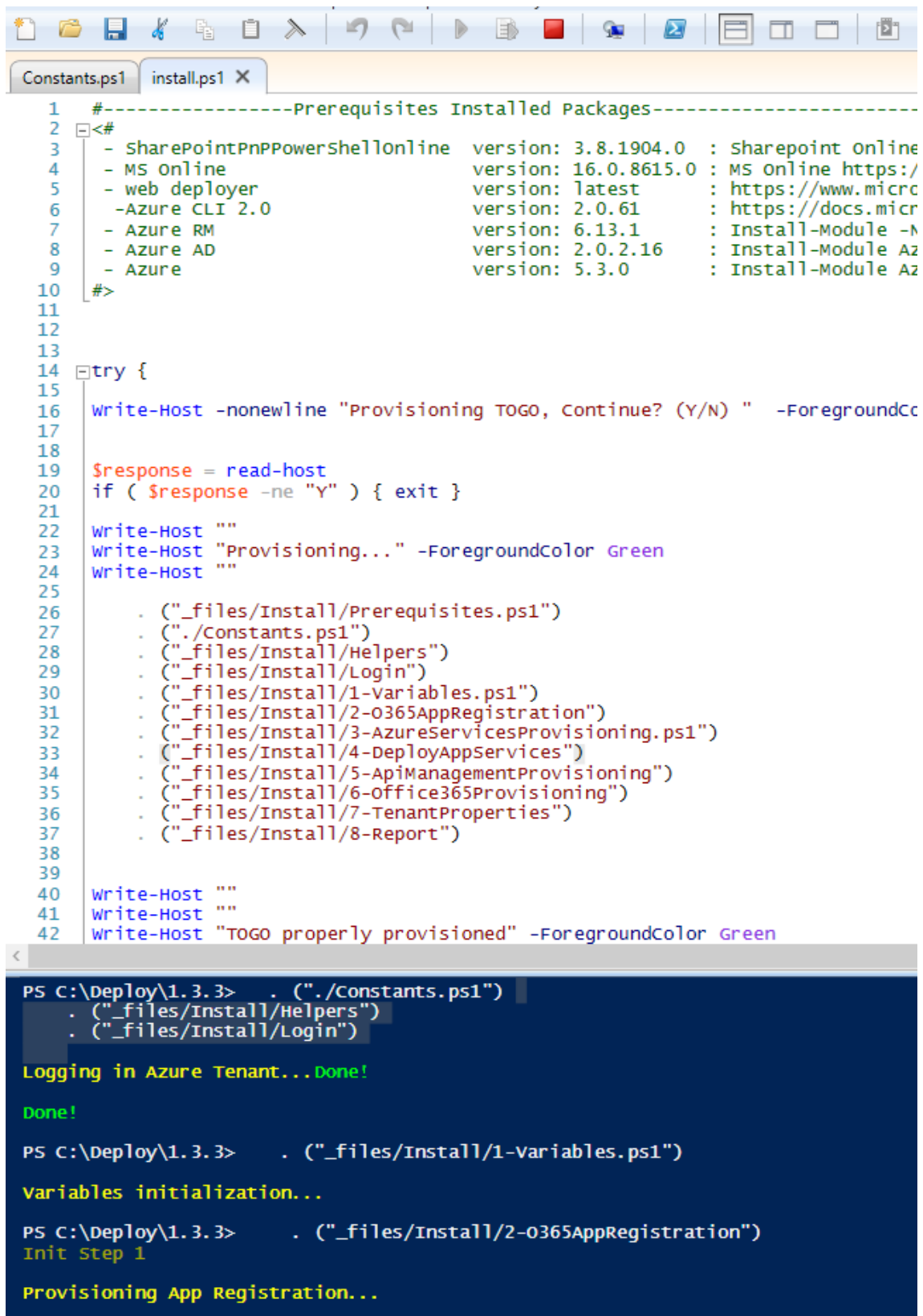
PS C:\Deploy\1.3.3> . ("_files/Install/1-Variables.ps1")

variables initialization...

PS C:\Deploy\1.3.3>

```

- Execute command line `. ("_files/Install/2-0365AppRegistration")`



The image shows a PowerShell script editor with two tabs: 'Constants.ps1' and 'install.ps1'. The 'install.ps1' tab is active, displaying a script that lists prerequisites and installed packages, prompts for provisioning confirmation, and then executes a series of commands to provision the environment.

```

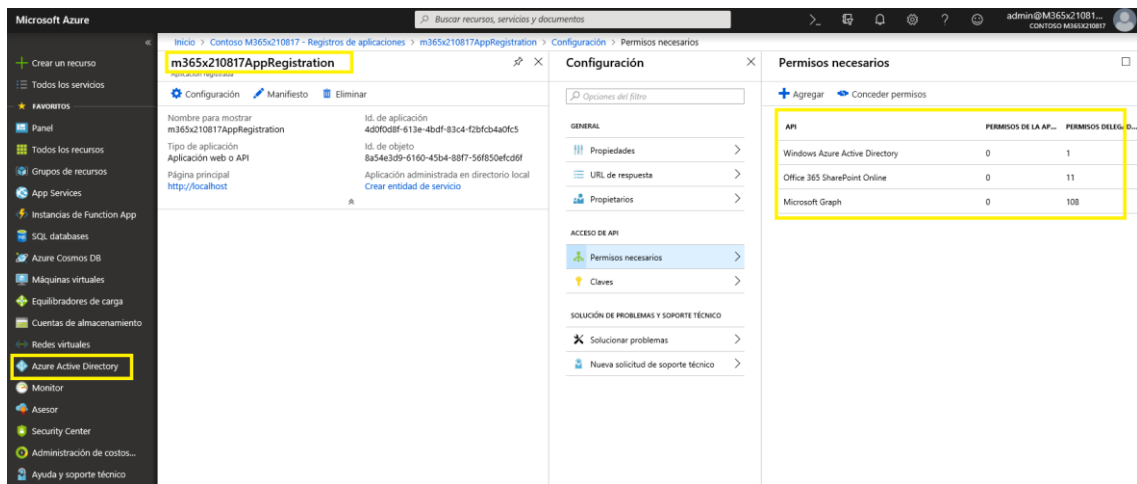
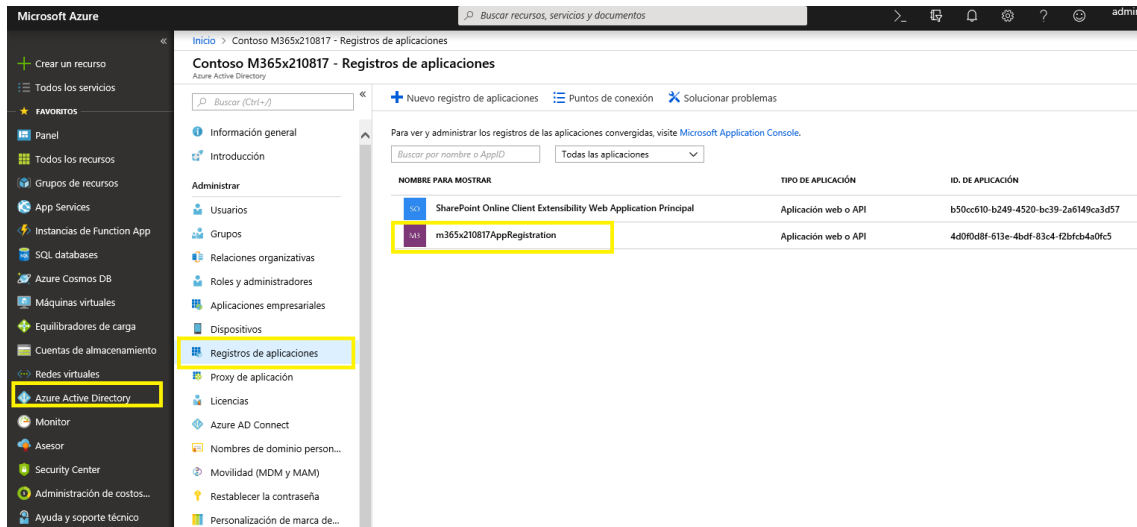
1  #-----Prerequisites Installed Packages-----
2  <#
3  - SharePointPnPPowerShellOnline version: 3.8.1904.0 : Sharepoint Online
4  - MS Online version: 16.0.8615.0 : MS Online https://
5  - web deployer version: latest : https://www.micro
6  - Azure CLI 2.0 version: 2.0.61 : https://docs.micr
7  - Azure RM version: 6.13.1 : Install-Module -N
8  - Azure AD version: 2.0.2.16 : Install-Module Az
9  - Azure version: 5.3.0 : Install-Module Az
10 #>
11
12
13
14 try {
15
16     Write-Host -nonewline "Provisioning TOGO, Continue? (Y/N) " -ForegroundColor Green
17
18
19     $response = read-host
20     if ( $response -ne "Y" ) { exit }
21
22     Write-Host ""
23     Write-Host "Provisioning..." -ForegroundColor Green
24     Write-Host ""
25
26     . ("_files/Install/Prerequisites.ps1")
27     . ("./Constants.ps1")
28     . ("_files/Install/Helpers")
29     . ("_files/Install/Login")
30     . ("_files/Install/1-Variables.ps1")
31     . ("_files/Install/2-0365AppRegistration")
32     . ("_files/Install/3-AzureServicesProvisioning.ps1")
33     . ("_files/Install/4-DeployAppServices")
34     . ("_files/Install/5-APIManagementProvisioning")
35     . ("_files/Install/6-Office365Provisioning")
36     . ("_files/Install/7-TenantProperties")
37     . ("_files/Install/8-Report")
38
39
40     Write-Host ""
41     Write-Host ""
42     Write-Host "TOGO properly provisioned" -ForegroundColor Green

```

The terminal window shows the execution of the script. It starts with the command `PS C:\Deploy\1.3.3> . ("./Constants.ps1")`, followed by `. ("_files/Install/Helpers")` and `. ("_files/Install/Login")`. The output shows "Logging in Azure Tenant...Done!" and "Done!". Then, the command `PS C:\Deploy\1.3.3> . ("_files/Install/1-variables.ps1")` is executed, resulting in "Variables initialization...". Next, the command `PS C:\Deploy\1.3.3> . ("_files/Install/2-0365AppRegistration")` is executed, resulting in "Init Step 1" and "Provisioning App Registration...".

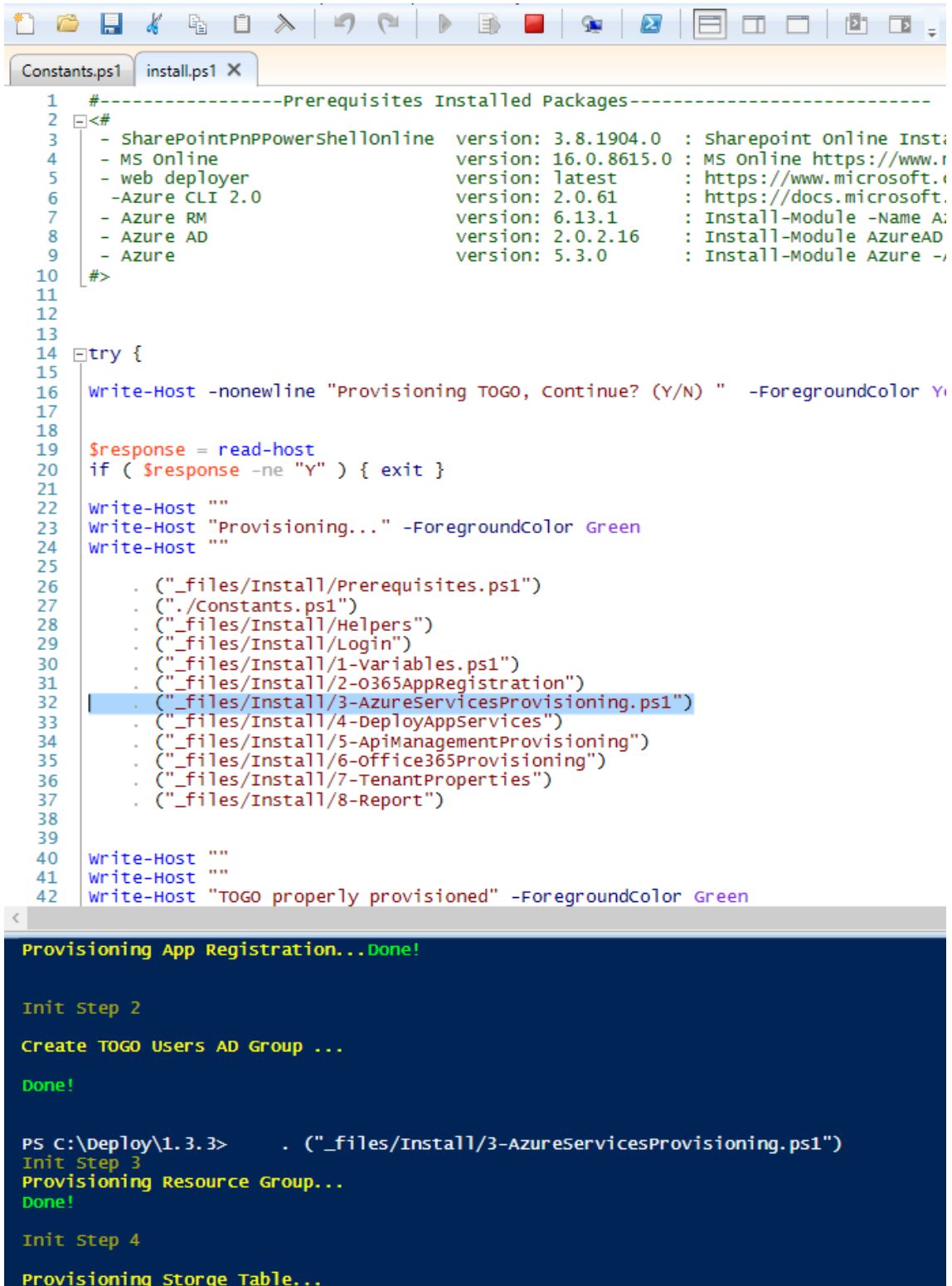
To check if the registration process done you can navigate to <https://azure.portal.com>

Login with the O365 Tenant Admin credentials and go to the Azure Active directory > Application Registration >>TenantName>AppRegistration



If the app have the same permissions showing in the above image, the process has done

- Execute command line . ("\_files/Install/3-AzureServicesProvisioning.ps1")



```

1  #-----Prerequisites Installed Packages-----
2  <#
3  - SharePointPnPPowerShellOnline version: 3.8.1904.0 : Sharepoint Online Inst
4  - MS Online version: 16.0.8615.0 : MS Online https://www.m
5  - web deployer version: latest : https://www.microsoft.c
6  - Azure CLI 2.0 version: 2.0.61 : https://docs.microsoft.
7  - Azure RM version: 6.13.1 : Install-Module -Name Az
8  - Azure AD version: 2.0.2.16 : Install-Module AzureAD
9  - Azure version: 5.3.0 : Install-Module Azure -
10 #>
11
12
13
14 try {
15
16     Write-Host -nonewline "Provisioning TOGO, Continue? (Y/N) " -ForegroundColor Y
17
18
19     $response = read-host
20     if ( $response -ne "Y" ) { exit }
21
22     Write-Host ""
23     Write-Host "Provisioning..." -ForegroundColor Green
24     Write-Host ""
25
26     . ("_files/Install/Prerequisites.ps1")
27     . ("./Constants.ps1")
28     . ("_files/Install/Helpers")
29     . ("_files/Install/Login")
30     . ("_files/Install/1-Variables.ps1")
31     . ("_files/Install/2-0365AppRegistration")
32     . ("_files/Install/3-AzureServicesProvisioning.ps1")
33     . ("_files/Install/4-DeployAppServices")
34     . ("_files/Install/5-APIManagementProvisioning")
35     . ("_files/Install/6-Office365Provisioning")
36     . ("_files/Install/7-TenantProperties")
37     . ("_files/Install/8-Report")
38
39
40     Write-Host ""
41     Write-Host ""
42     Write-Host "TOGO properly provisioned" -ForegroundColor Green
43 }
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Provisioning App Registration...Done!

Init Step 2

Create TOGO Users AD Group ...

Done!

PS C:\Deploy\1.3.3> . ("\_files/Install/3-AzureServicesProvisioning.ps1")

Init Step 3

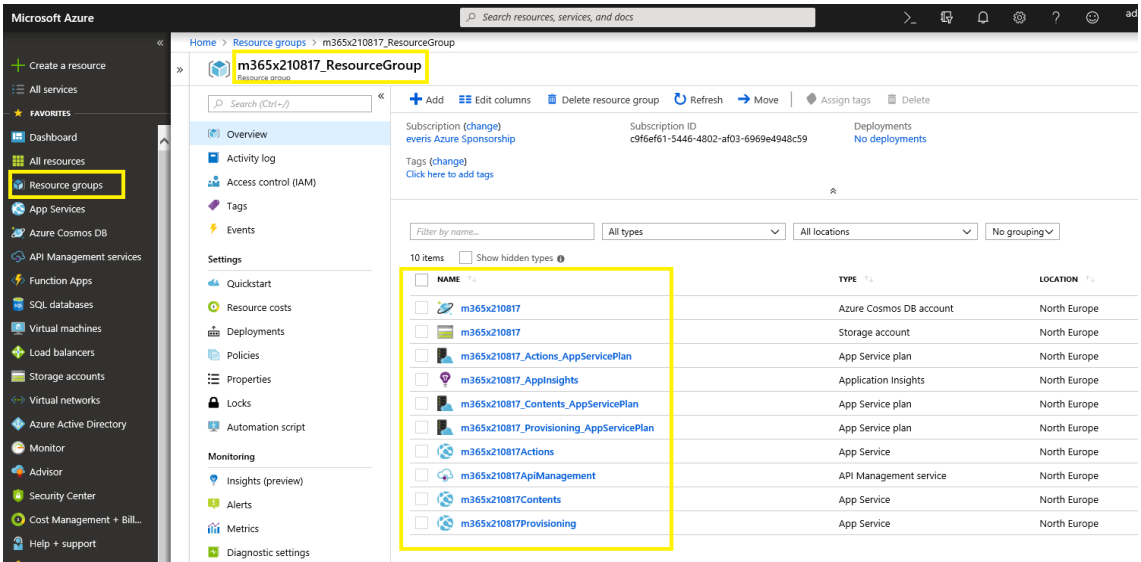
Provisioning Resource Group...

Done!

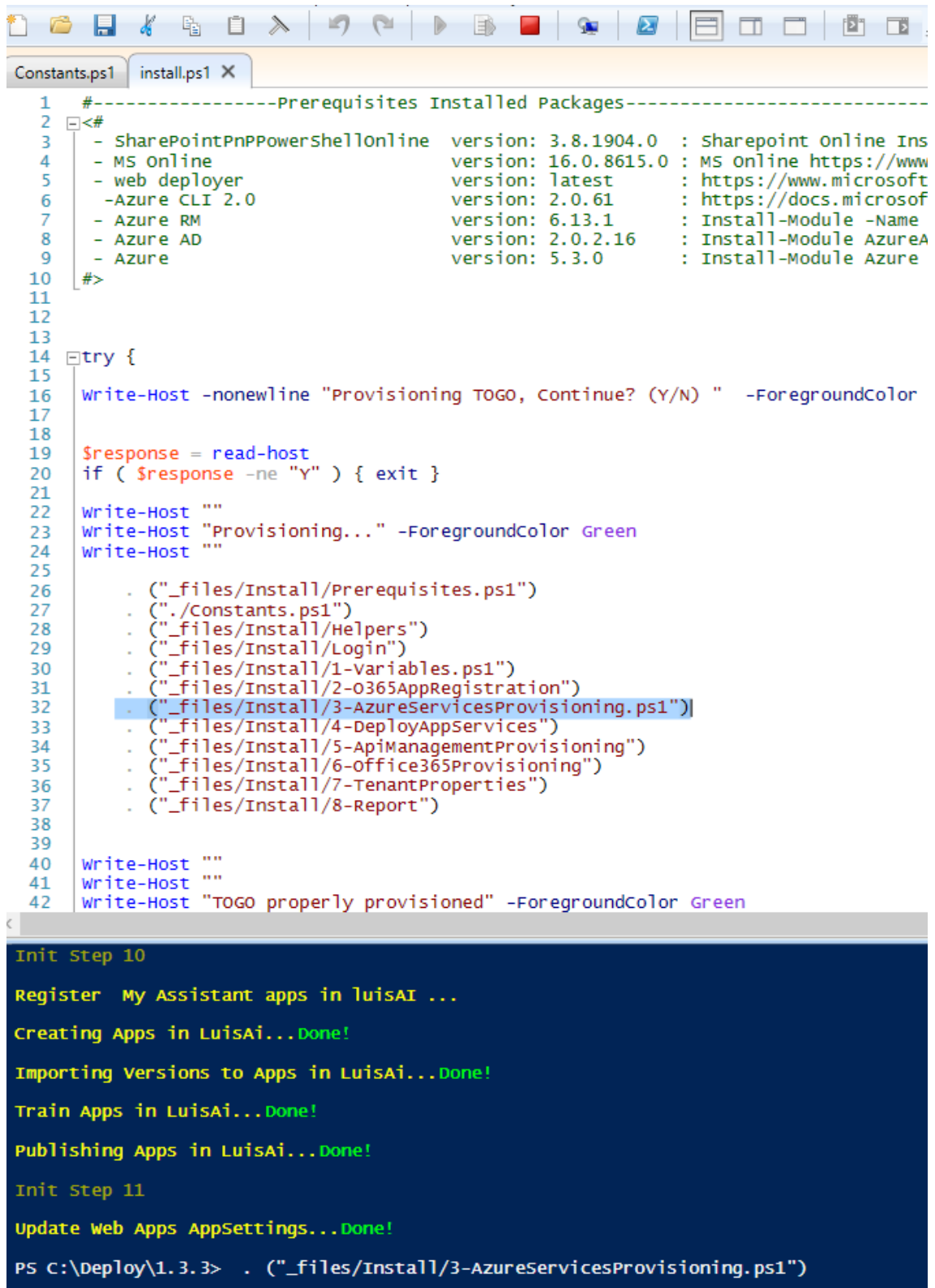
Init Step 4

Provisioning Storage Table...

When the script execution is done, you can go to azure portal and check if all resources had been created



- Execute command line . ("\_files/Install/4-DeployAppServices")



```

1  #-----Prerequisites Installed Packages-----
2  <#
3  - SharePointPnPPowerShellOnline version: 3.8.1904.0 : Sharepoint Online Ins
4  - MS Online version: 16.0.8615.0 : MS Online https://www
5  - web deployer version: latest : https://www.microsoft
6  - Azure CLI 2.0 version: 2.0.61 : https://docs.microsof
7  - Azure RM version: 6.13.1 : Install-Module -Name
8  - Azure AD version: 2.0.2.16 : Install-Module AzureA
9  - Azure version: 5.3.0 : Install-Module Azure
10 #>
11
12
13
14 try {
15
16     Write-Host -nonewline "Provisioning TOGO, Continue? (Y/N) " -ForegroundColor
17
18
19     $response = read-host
20     if ( $response -ne "Y" ) { exit }
21
22     Write-Host ""
23     Write-Host "Provisioning..." -ForegroundColor Green
24     Write-Host ""
25
26     . ("_files/Install/Prerequisites.ps1")
27     . ("./Constants.ps1")
28     . ("_files/Install/Helpers")
29     . ("_files/Install/Login")
30     . ("_files/Install/1-variables.ps1")
31     . ("_files/Install/2-0365AppRegistration")
32     . ("_files/Install/3-AzureServicesProvisioning.ps1")
33     . ("_files/Install/4-DeployAppServices")
34     . ("_files/Install/5-APIManagementProvisioning")
35     . ("_files/Install/6-Office365Provisioning")
36     . ("_files/Install/7-TenantProperties")
37     . ("_files/Install/8-Report")
38
39
40     Write-Host ""
41     Write-Host ""
42     Write-Host "TOGO properly provisioned" -ForegroundColor Green

```

```

Init Step 10
Register My Assistant apps in luisAI ...
Creating Apps in LuisAi...Done!
Importing Versions to Apps in LuisAi...Done!
Train Apps in LuisAi...Done!
Publishing Apps in LuisAi...Done!
Init Step 11
Update Web Apps AppSettings...Done!
PS C:\Deploy\1.3.3> . ("_files/Install/3-AzureServicesProvisioning.ps1")

```

- Execute command line `(".") ("_files/Install/5-APIManagementProvisioning")`



```

1  #-----Prerequisites Installed Packages-----
2  <#
3  - SharePointPnPPowerShellOnline version: 3.8.1904.0 : Sharepoint Online Install-Modul
4  - MS Online version: 16.0.8615.0 : MS Online https://www.microsoft
5  - web deployer version: latest : https://www.microsoft.com/en-us
6  -Azure CLI 2.0 version: 2.0.61 : https://docs.microsoft.com/en-u
7  - Azure RM version: 6.13.1 : Install-Module -Name AzureRM
8  - Azure AD version: 2.0.2.16 : Install-Module AzureAD -AllowCl
9  - Azure version: 5.3.0 : Install-Module Azure -AllowClob
10 #>
11
12
13
14 try {
15     write-Host -newline "Provisioning TOGO, Continue? (Y/N) " -ForegroundColor Yellow
16
17
18     $response = read-host
19     if ( $response -ne "Y" ) { exit }
20
21     write-Host ""
22     write-Host "Provisioning..." -ForegroundColor Green
23     write-Host ""
24
25     . ("_files/Install/Prerequisites.ps1")
26     . ("./Constants.ps1")
27     . ("_files/Install/Helpers")
28     . ("_files/Install/Login")
29     . ("_files/Install/1-Variables.ps1")
30     . ("_files/Install/2-0365AppRegistration")
31     . ("_files/Install/3-AzureServicesProvisioning.ps1")
32     . ("_files/Install/4-DeployAppServices")
33     . ("_files/Install/5-ApiManagementProvisioning")
34     . ("_files/Install/6-Office365Provisioning")
35     . ("_files/Install/7-TenantProperties")
36     . ("_files/Install/8-Report")
37
38
39     write-Host ""
40     write-Host ""
41     write-Host "TOGO properly provisioned" -ForegroundColor Green
42 }

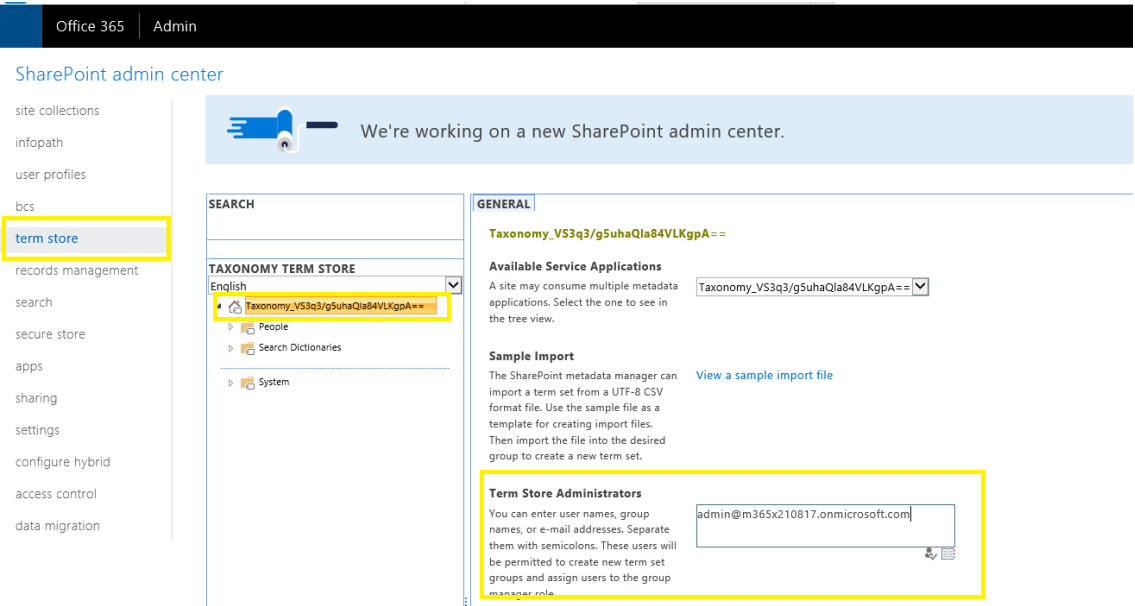
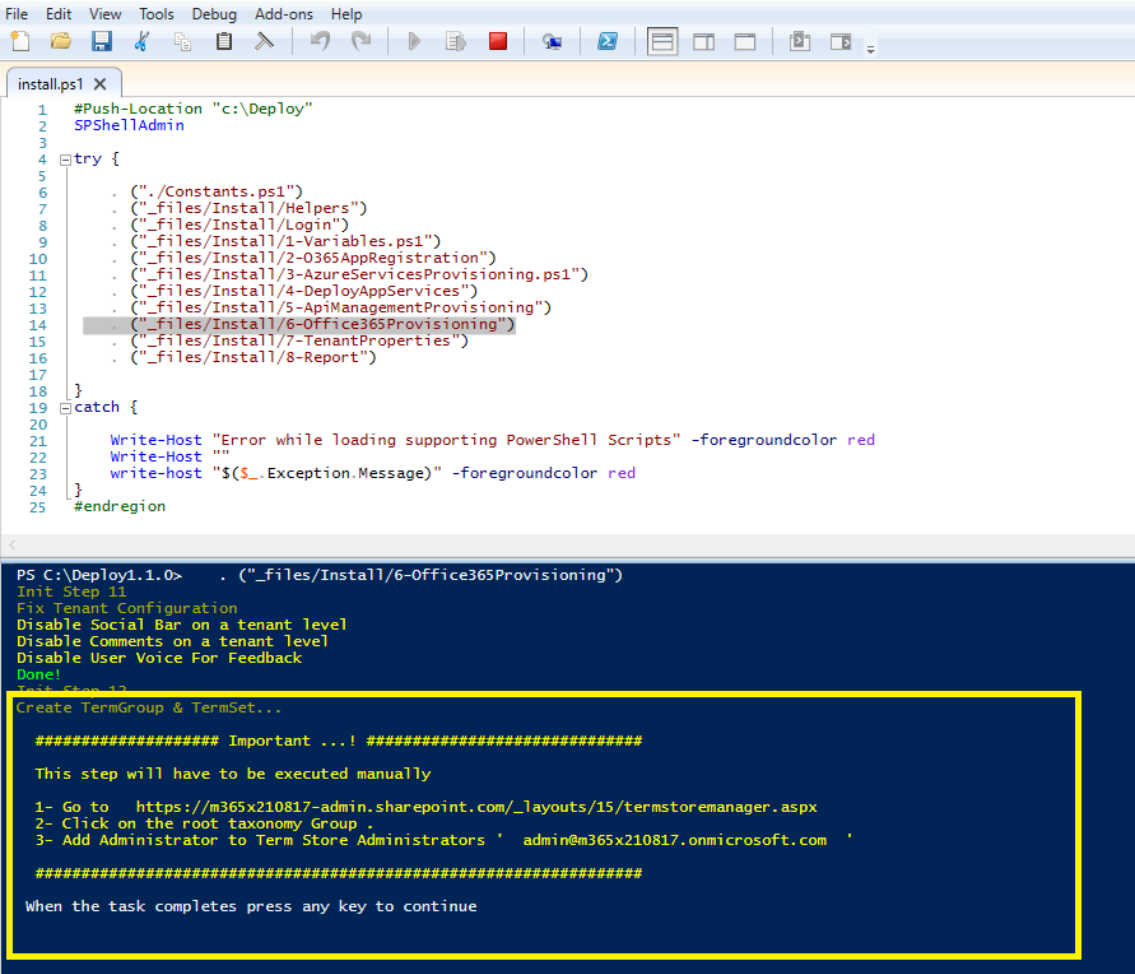
```

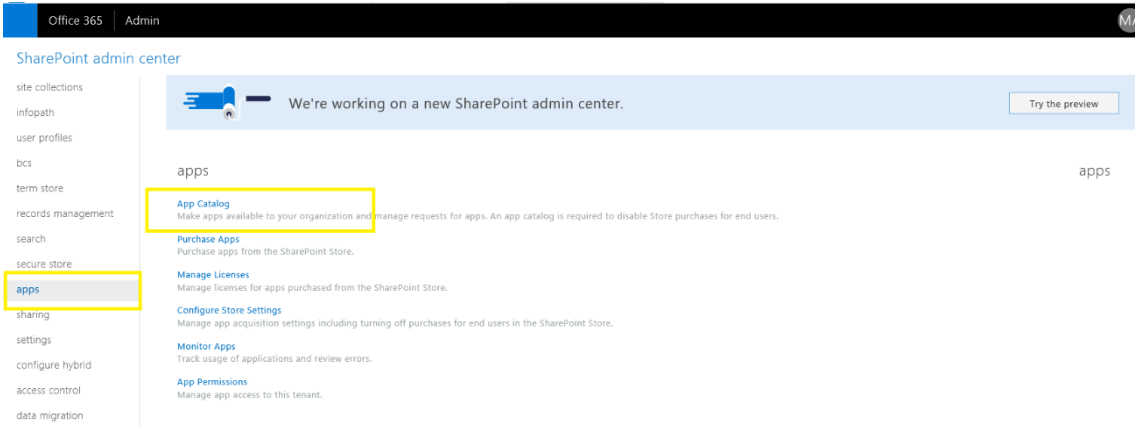
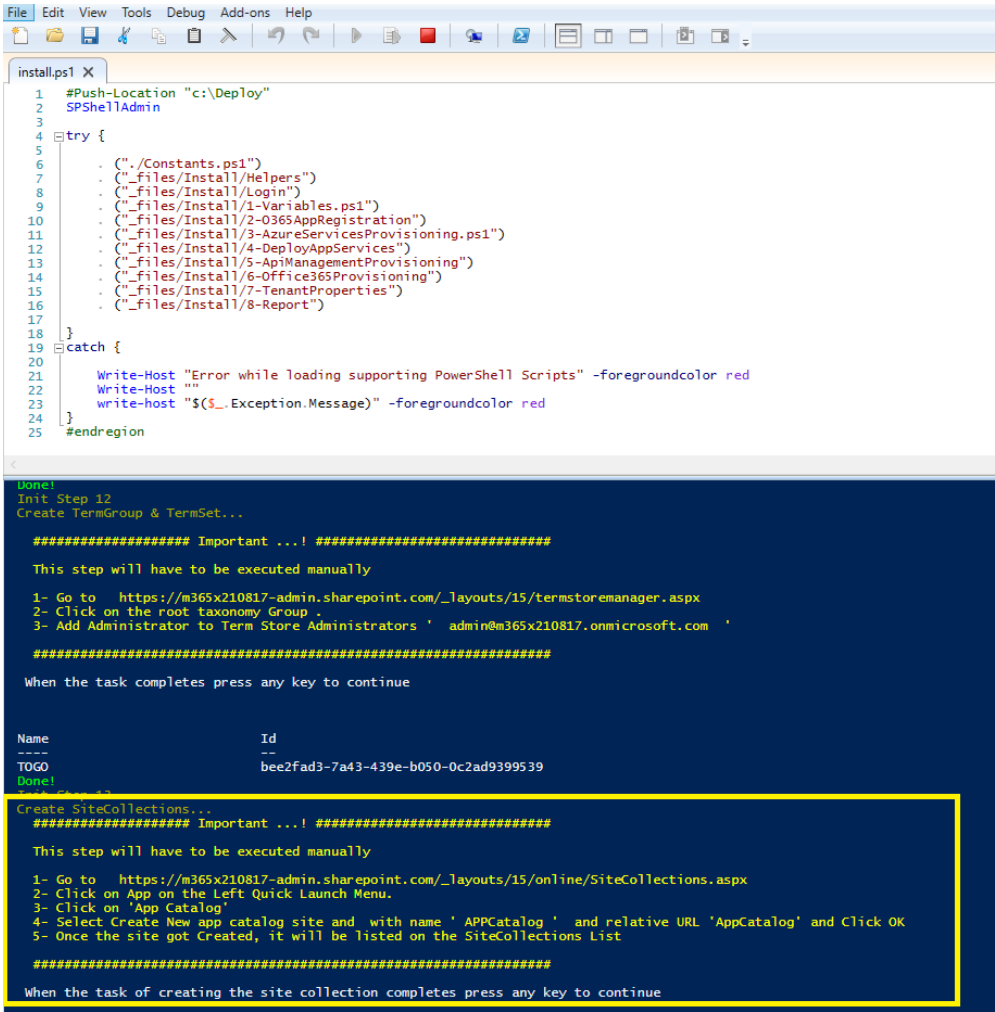
```

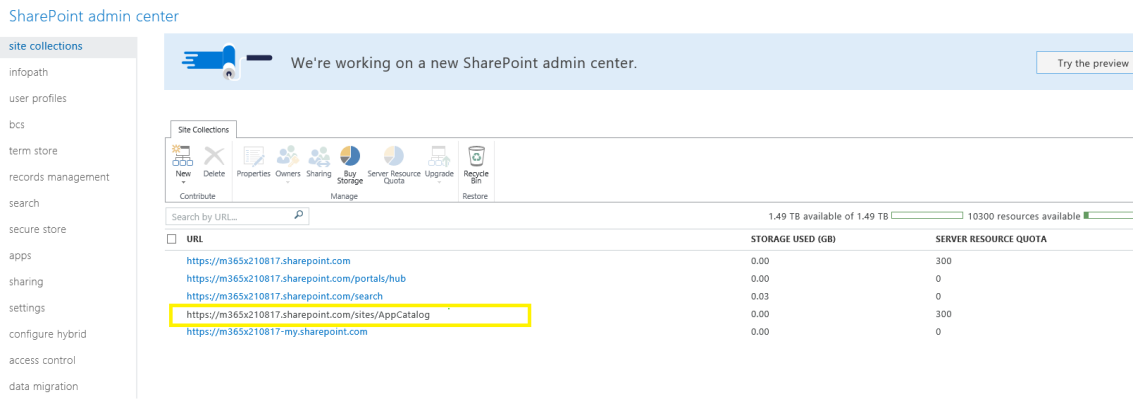
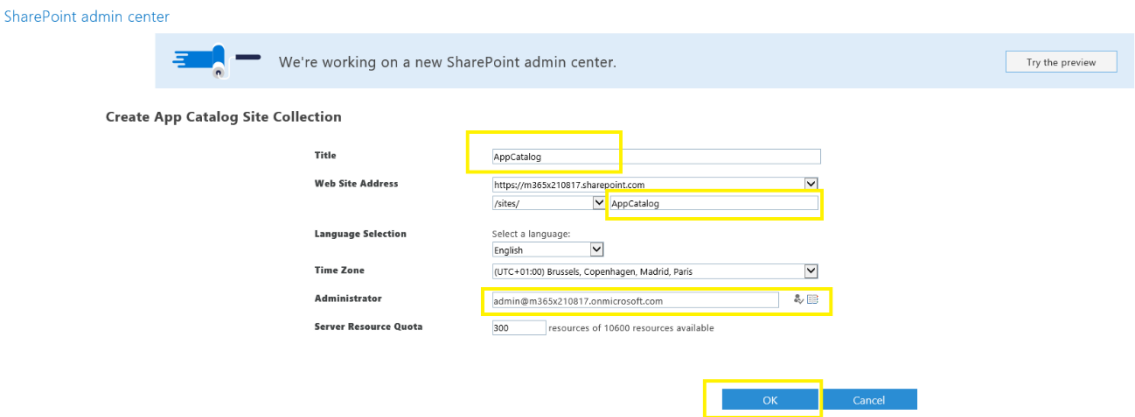
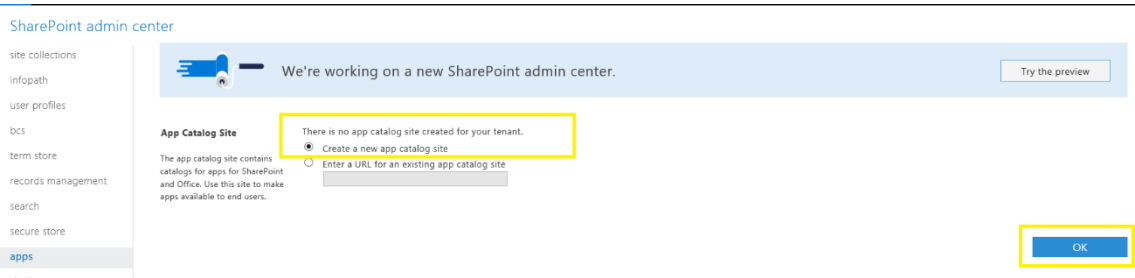
Init Step 10
Register My Assistant apps in LuisAI ...
Creating Apps in LuisAi...Done!
Importing Versions to Apps in LuisAi...Done!
Train Apps in LuisAi...Done!
Publishing Apps in LuisAi...Done!
Init Step 11
Update Web Apps AppSettings...Done!
PS C:\Deploy\1.3.3> . ("_files/Install/5-ApiManagementProvisioning")

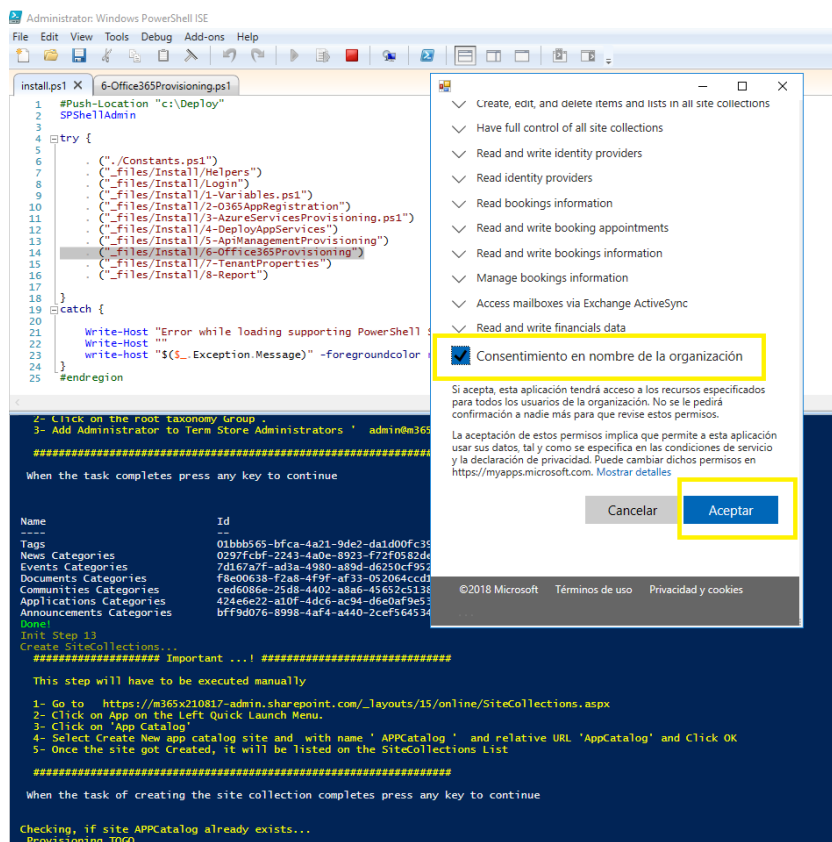
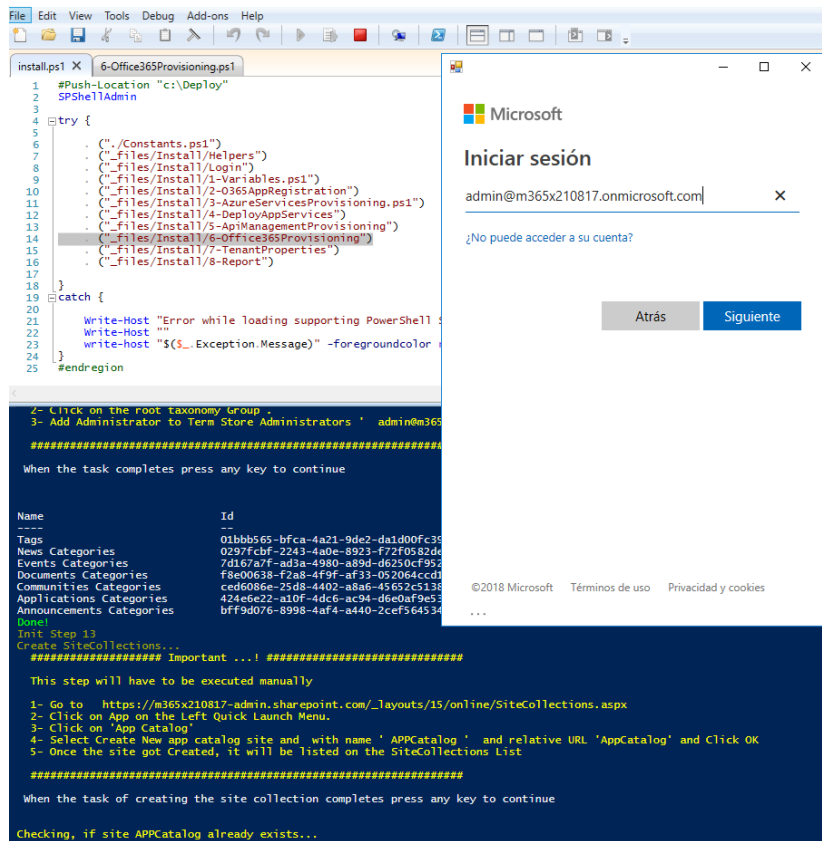
```

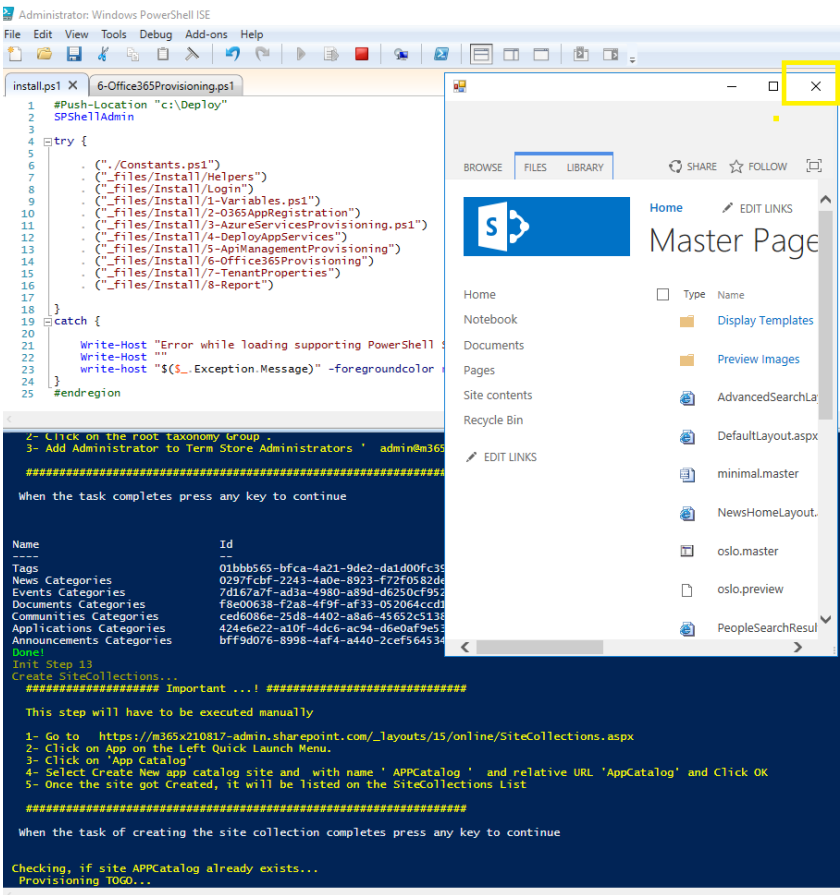
- Execute command line `. ("_files/Install/6-Office365Provisioning")`











```
File Edit View Tools Debug Add-ons Help
install.ps1 X 6-Office365Provisioning.ps1
4 try {
5     . ("./Constants.ps1")
6     . ("Files/Install/Helpers")
7     . ("Files/Install/Login")
8     . ("Files/Install/1-Variables.ps1")
9     . ("Files/Install/2-0365AppRegistration")
10    . ("Files/Install/3-AzureServicesProvisioning.ps1")
11    . ("Files/Install/4-DeployAppServices")
12    . ("Files/Install/5-AppManagementProvisioning")
13    . ("Files/Install/6-Office365Provisioning")
14    . ("Files/Install/7-TenantProperties")
15    . ("Files/Install/8-Report")
16 }
17 }
18 }
19 }
20 }
21 }
22 }
23 }
24 }
25 }
26 }
27 }
28 }
29 }
30 }
31 }
32 }
33 }
34 }
35 }
36 }
37 }
38 }
39 }
40 }
41 }
42 }
43 }
44 }
45 }
46 }
47 }
48 }
49 }
50 }
51 }
52 }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
69 }
70 }
71 }
72 }
73 }
74 }
75 }
76 }
77 }
78 }
79 }
80 }
81 }
82 }
83 }
84 }
85 }
86 }
87 }
88 }
89 }
90 }
91 }
92 }
93 }
94 }
95 }
96 }
97 }
98 }
99 }
100 }
```

When the task of creating the site collection completes press any key to continue

Checking, if site APPCatalog already exists...  
Provisioning TÖGÖ... { "results": { "status":true } }

Sleeping for 2 min to give the Togo Site a chance to finish creating...  
Done!

Init Step 14  
Register TÖGÖ Authorization App...

Credentials : {}  
ExtensionData : System.Runtime.Serialization.ExtensionDataObject  
AccountEnabled : True  
Addresses : {Microsoft.Online.Administration.RedirectUri}  
AppPrincipalId : b69f4305-064c-4d6d-868a-4911bebb842d  
DisplayName : TÖGÖAppAuthorization  
ObjectId : ad7264dd-b311-4265-81cd-4eed9526d6e9  
ServicePrincipalNames : {m365x210817.sharepoint.com, b69f4305-064c-4d6d-868a-4911bebb842d}  
TrustedForDelegation : False

##### Important !!! #####

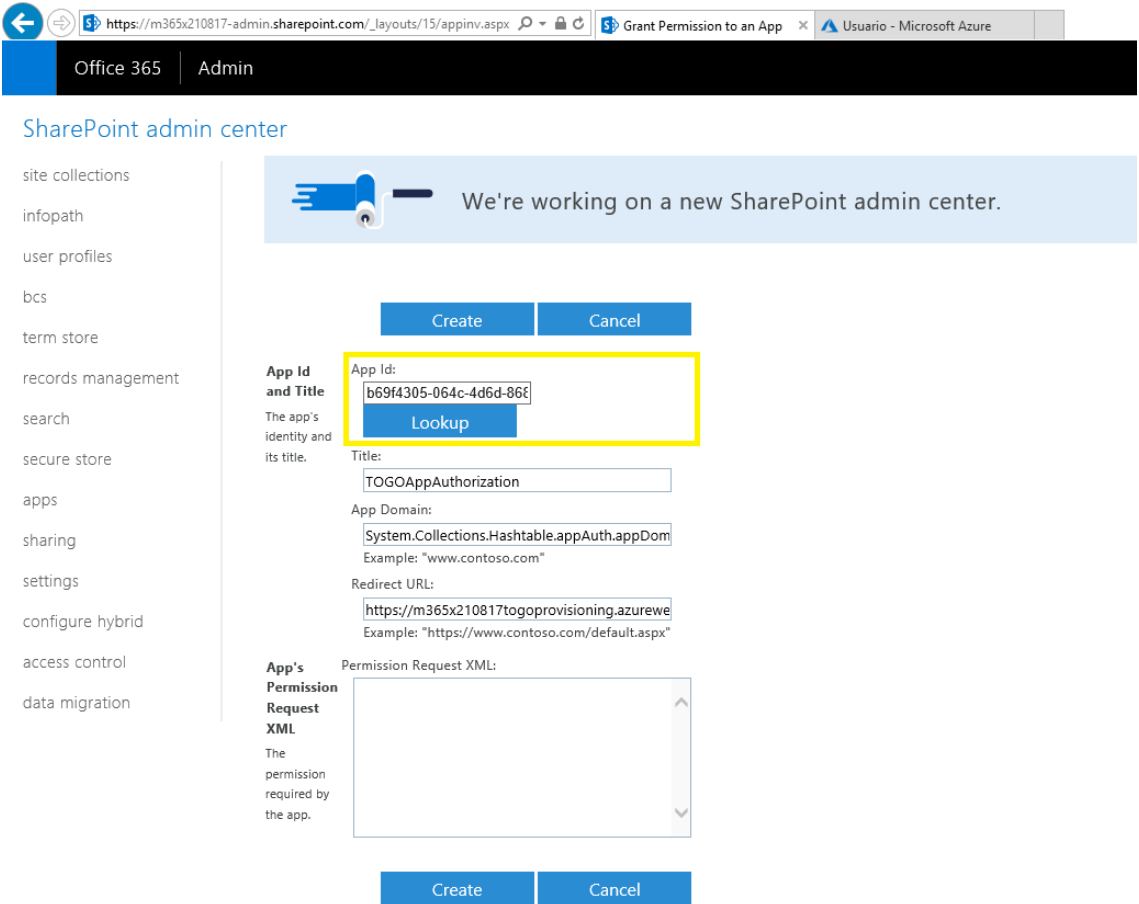
This step will have to be executed manually

1- Go to : [https://m365x210817-admin.sharepoint.com/\\_layouts/15/appinv.aspx](https://m365x210817-admin.sharepoint.com/_layouts/15/appinv.aspx)  
2- Find the The App with the ID " b69f4305-064c-4d6d-868a-4911bebb842d "  
3- Copy and past the XML text in Apps Permission Request XML

<AppPermissionRequests AllowAppOnlyPolicy="true">  
<AppPermissionRequest Scope="http://sharepoint/content/tenant" Right="FullControl" />  
</AppPermissionRequests>


#####

When the task completes press any key to continue



SharePoint admin center

- site collections
- infopath
- user profiles
- bcs
- term store
- records management
- search
- secure store
- apps
- sharing
- settings
- configure hybrid
- access control
- data migration

 We're working on a new SharePoint admin center.

Create

Cancel

**App Id and Title**  
The app's identity and its title.

App Id:  
b69f4305-064c-4d6d-86f8-  

Lookup

Title:  
TOGOAppAuthorization

App Domain:  
m365x210817.sharepoint.com  
Example: "www.contoso.com"

Redirect URL:  
https://m365x210817.sharepoint.com  
Example: "https://www.contoso.com/default.aspx"

**App's Permission Request XML**  
The permission required by the app.


Permission Request XML:  
<AppPermissionRequests  
  AllowAppOnlyPolicy="true">  
  <AppPermissionRequest  
    Scope="http://sharepoint/content/tenant"  
    Right="FullControl" />  
  </AppPermissionRequest>  
</AppPermissionRequests>

Create

Cancel

SharePoint admin center

- site collections
- infopath
- user profiles
- bcs
- term store
- records management
- search
- secure store
- apps
- sharing
- settings
- configure hybrid
- access control
- data migration


 We're working on a new SharePoint admin center.

### Do you trust TOGOAppAuthorization?

Let it have full control of all site collections.

Let it share its permissions with other users.

Let it access basic information about the users of this site.

  
TOGOAppAuthorization

Trust It

Cancel



```

Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
install.ps1 X 6-Office365Provisioning.ps1
4 try {
5     . ("./Constants.ps1")
6     . ("_files/Install/Helpers")
7     . ("_files/Install/Login")
8     . ("_files/Install/1-Variables.ps1")
9     . ("_files/Install/2-0365AppRegistration")
10    . ("_files/Install/3-AzureServicesProvisioning.ps1")
11    . ("_files/Install/4-DeployAppServices")
12    . ("_files/Install/5-APIManagementProvisioning")
13    . ("_files/Install/6-Office365Provisioning")
14    . ("_files/Install/7-TenantProperties")
15    . ("_files/Install/8-Report")
16 }
17 }
18 }
19 catch {
20     Write-Host "Error while loading supporting PowerShell Scripts" -foregroundcolor red
21     Write-Host ""
22     Write-Host "$($_.Exception.Message)" -foregroundcolor red
23 }
24 }
25 #endregion

Init Step 14
Register TÖGÖ Authorization App...
Credentials : {}
ExtensionData : System.Runtime.Serialization.ExtensionDataObject
AccountEnabled : True
Addresses : {Microsoft.Online.Administration.RedirectUri}
AppPrincipalId : b69f4305-064c-4d6d-868a-4911bebb842d
DisplayName : TÖGÖAppAuthorization
ObjectId : ad7264dd-b311-4269-81cd-4eed9526d6e9
ServicePrincipalNames : {m365x210817.sharepoint.com, b69f4305-064c-4d6d-868a-4911bebb842d}
TrustedForDelegation : False

##### Important ...! #####

This step will have to be executed manually

1- Go to : https://m365x210817-admin.sharepoint.com/_layouts/15/appinv.aspx
2- find the The App with the ID : ' b69f4305-064c-4d6d-868a-4911bebb842d '
3- Copy and past the XML text in Apps Permission Request XML

<AppPermissionRequests AllowAppOnlyPolicy="true">
<AppPermissionRequest Scope="http://sharepoint/content/tenant" Right="FullControl" />
</AppPermissionRequests>

#####

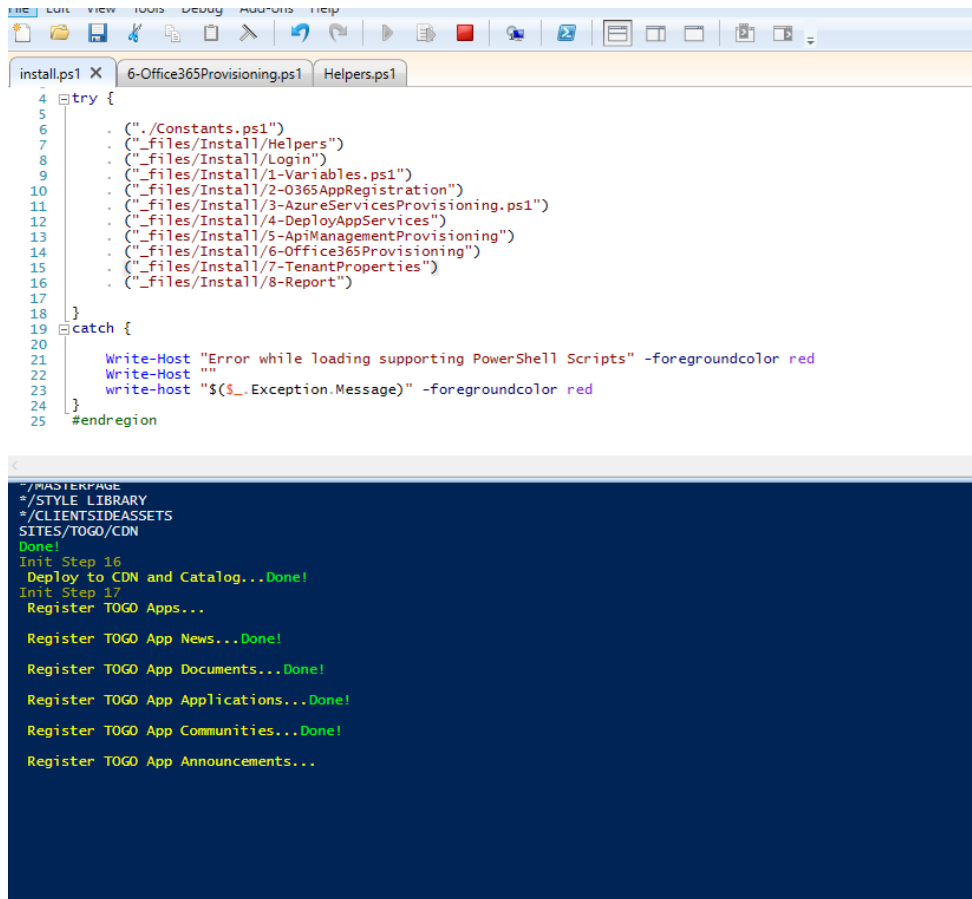
When the task completes press any key to continue

Done!
Init Step 15
Set SPOTenantCDN...Done.
Creating CDN Library...Done.
Done.
Configuring CDN...

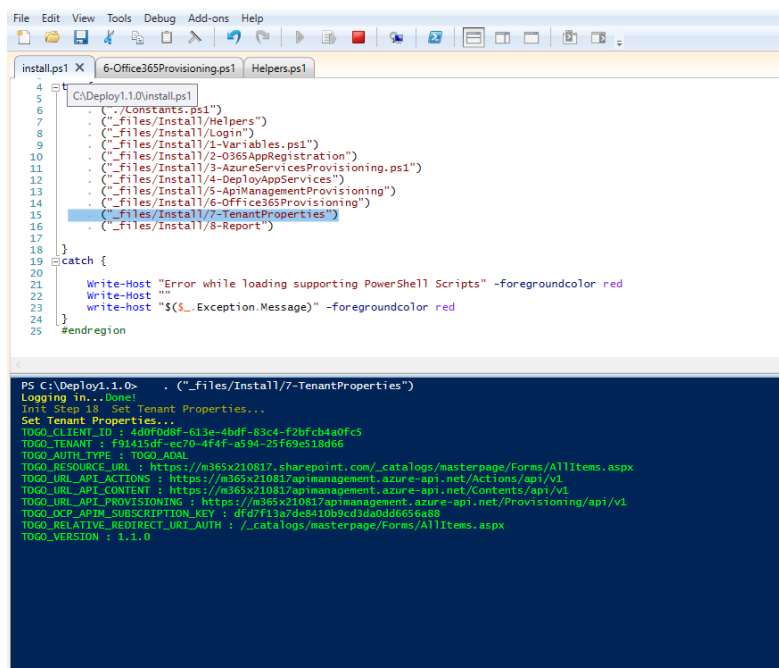
Value : True

WARNING: This is a feature built on a 3rd party application with privacy and compliance standards that differ from the commitments outlined by the
t conform to the Microsoft Data Processing Terms (DPT) and is outside of the Microsoft Office365 Trust Center compliance boundaries.

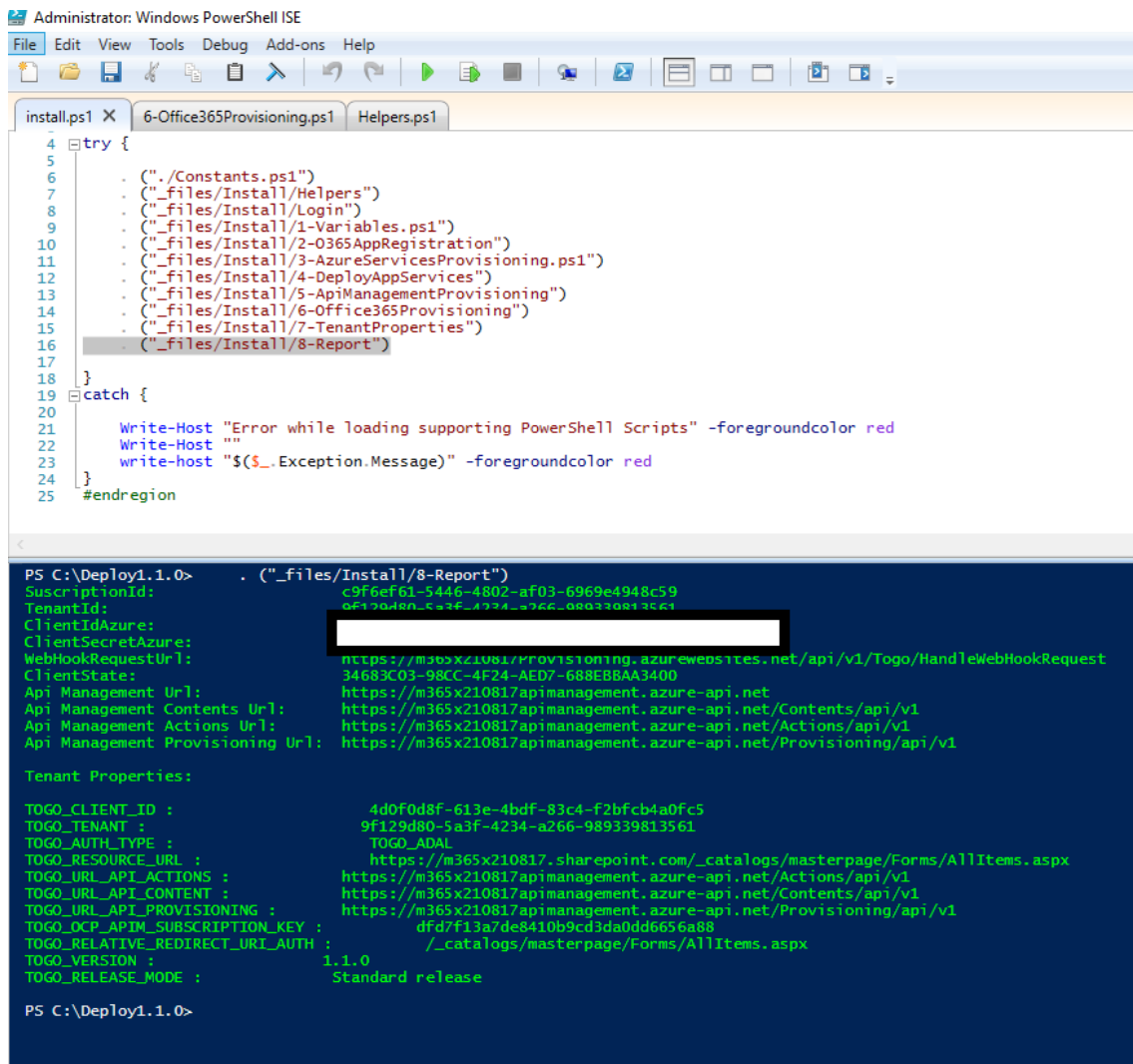
```



- Execute command line . ("\_files/Install/7-TenantProperties")



- Execute command line . ("\_files/Install/8-Report")



```

Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
install.ps1 X 6-Office365Provisioning.ps1 Helpers.ps1
4 try {
5
6     . ("./Constants.ps1")
7     . ("_files/Install/Helpers")
8     . ("_files/Install/Login")
9     . ("_files/Install/1-Variables.ps1")
10    . ("_files/Install/2-0365AppRegistration")
11    . ("_files/Install/3-AzureServicesProvisioning.ps1")
12    . ("_files/Install/4-DeployAppServices")
13    . ("_files/Install/5-APIManagementProvisioning")
14    . ("_files/Install/6-Office365Provisioning")
15    . ("_files/Install/7-TenantProperties")
16    . ("_files/Install/8-Report")
17 }
18
19 catch {
20
21     Write-Host "Error while loading supporting PowerShell Scripts" -foregroundcolor red
22     Write-Host ""
23     Write-Host "$($_.Exception.Message)" -foregroundcolor red
24 }
25 #endregion

PS C:\Deploy1.1.0> . ("_files/Install/8-Report")
SubscriptionId: c9f6ef61-5446-4802-af03-6969e4948c59
TenantId: 9f129d80-5a3f-4234-a266-989339813561
ClientIdAzure: 
ClientSecretAzure: 
WebHookRequestUrl: https://m365x210817provisioning.azurewebsites.net/api/v1/Togo/HandleWebHookRequest
ClientState: 34683C03-98CC-4F24-AED7-688EBBAA3400
Api Management Url: https://m365x210817apimanagement.azure-api.net
Api Management Contents Url: https://m365x210817apimanagement.azure-api.net/Contents/api/v1
Api Management Actions Url: https://m365x210817apimanagement.azure-api.net/Actions/api/v1
Api Management Provisioning Url: https://m365x210817apimanagement.azure-api.net/Provisioning/api/v1

Tenant Properties:
TOGO_CLIENT_ID : 4d0f0d8f-613e-4bdf-83c4-f2bfc4a0fc5
TOGO_TENANT : 9f129d80-5a3f-4234-a266-989339813561
TOGO_AUTH_TYPE : TOGO_ADAL
TOGO_RESOURCE_URL : https://m365x210817.sharepoint.com/_catalogs/masterpage/Forms/AllItems.aspx
TOGO_URL_API_ACTIONS : https://m365x210817apimanagement.azure-api.net/Actions/api/v1
TOGO_URL_API_CONTENT : https://m365x210817apimanagement.azure-api.net/Contents/api/v1
TOGO_URL_API_PROVISIONING : https://m365x210817apimanagement.azure-api.net/Provisioning/api/v1
TOGO_OCP_API_SUBSCRIPTION_KEY : dfd7f13a7de8410b9cd3da0dd6656a88
TOGO_RELATIVE_REDIRECT_URI_AUTH : /_catalogs/masterpage/Forms/AllItems.aspx
TOGO_VERSION : 1.1.0
TOGO_RELEASE_MODE : Standard release

PS C:\Deploy1.1.0>
  
```

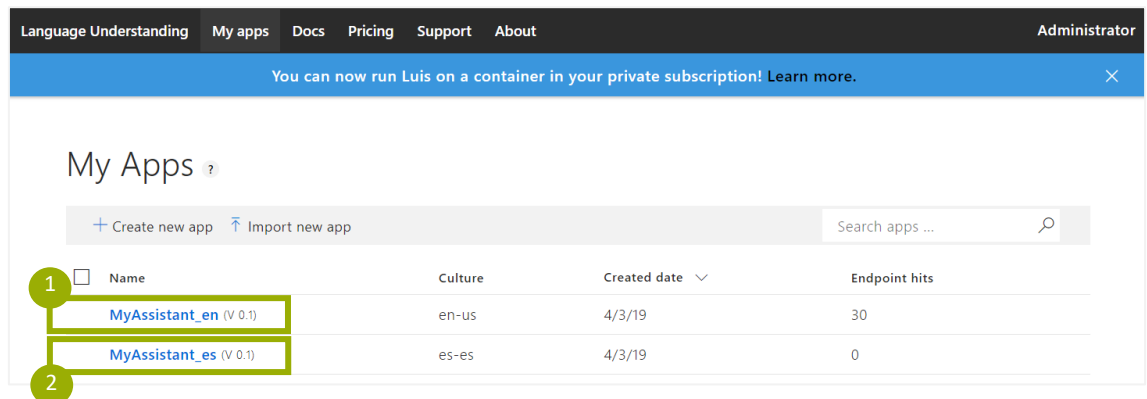
## Post deployment actions

### Link luis.ai applications with the Azure Subscription

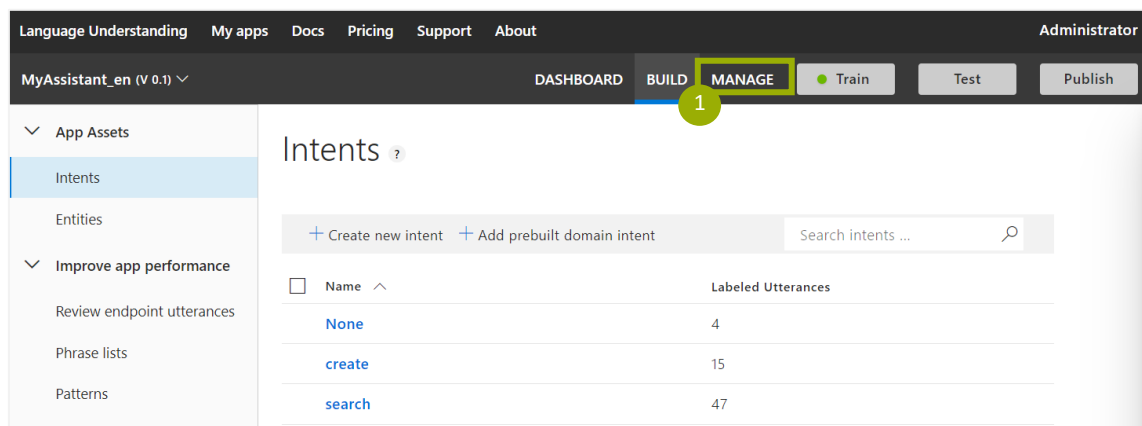
Once the deployment is finished is necessary to change the out-of-the-box configuration of the luis.ai applications.

Following are the steps required:

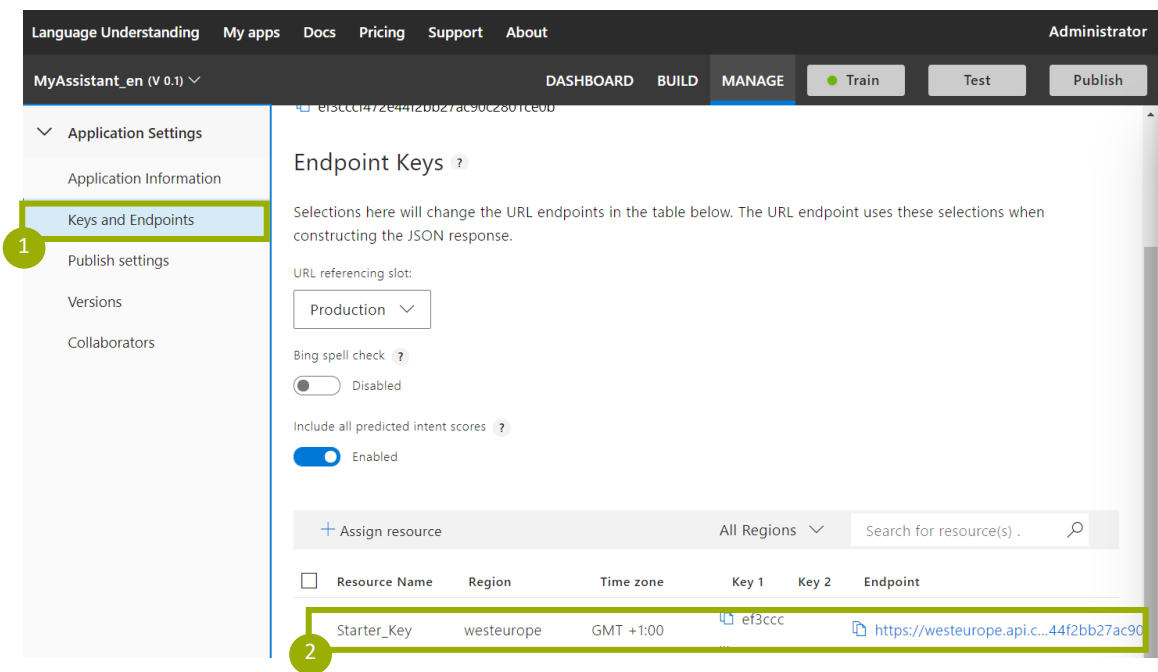
1. Navigate to <https://eu.luis.ai>
2. Sing in with an Azure Subscription Administrator user
3. Select an application



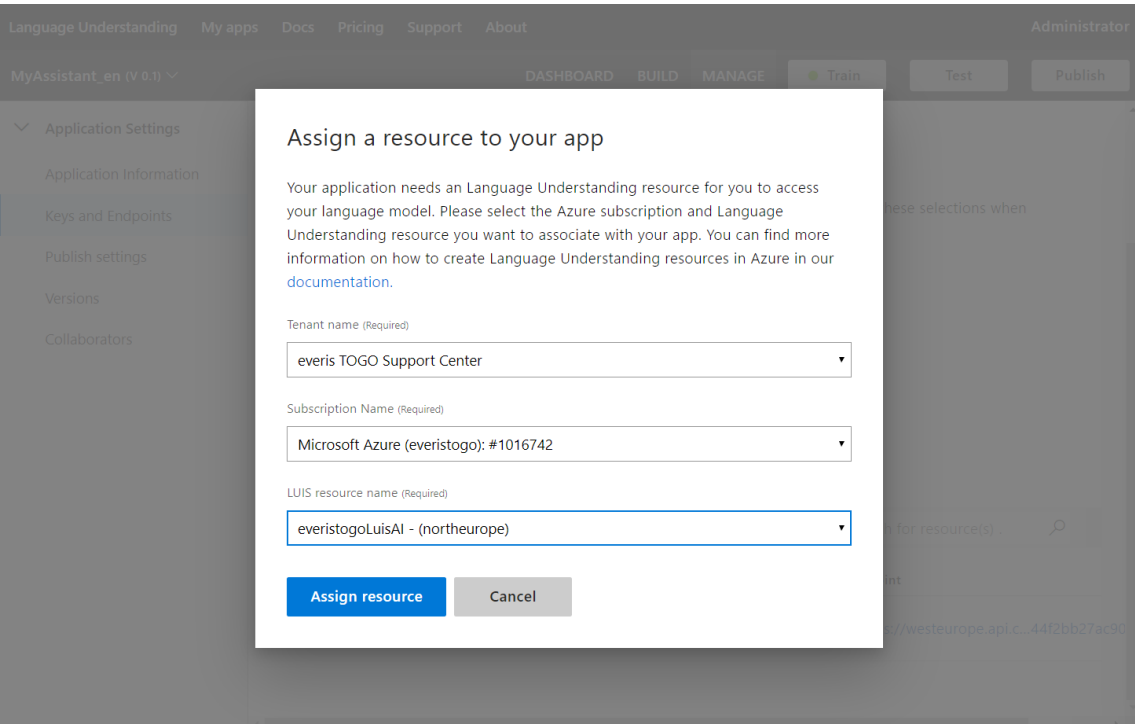
4. Click on manage



5. Click on Keys and EndPoints and check the associated subscription, it will be configured with an Starter\_Key by default



6. Click on add resource and select the Azure tenant name, Subscription and everistogLuisAI that will be automatically created and click on assign resource

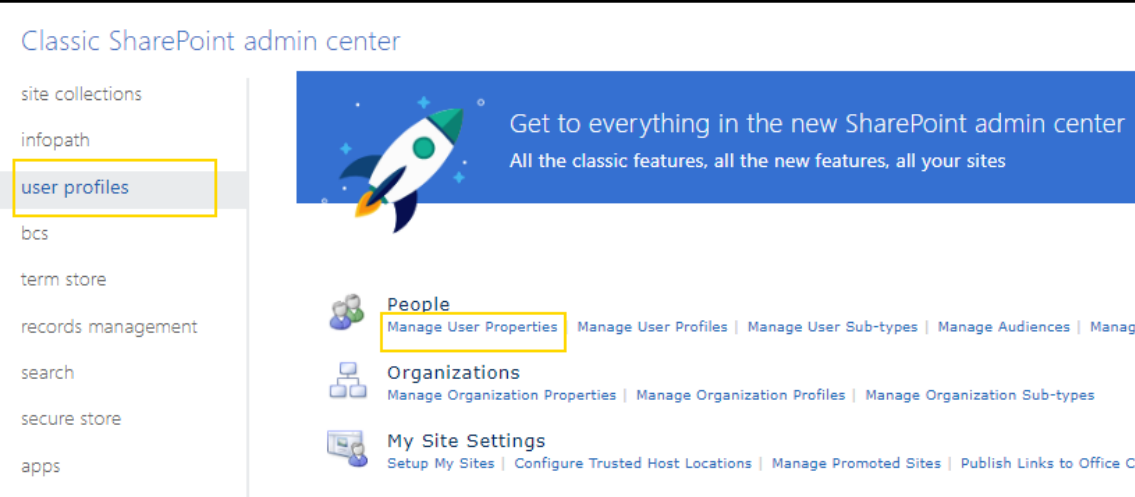


## User profile custom Properties (Audience Targeting)

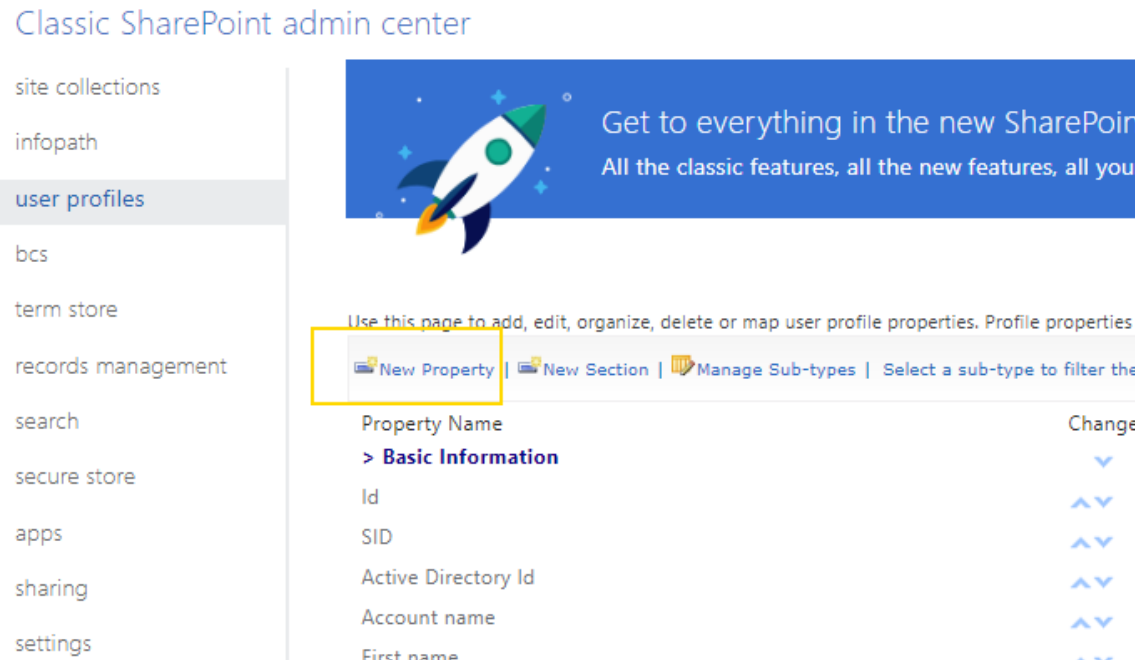
Following there is the description of the manual steps needed to create the two User Profile properties required to the Audience targeting business logic.

At the time of writing this guide the only way to create User Profile properties is through the classic experience of the SharePoint Admin Center.

From the user profiles option, first select “Manage User Properties”



For each property we need, you need to create a “New property”



See below the details needed to create the user property involved in the department audience targeting: TOGO-Audience-Department

Use this page to add a property for user profiles.

\* Indicates a required field

Property Settings

Specify property settings for this property. The name will be used programmatically for the property by the user profile service, while the display name is the label used when the property is shown. After the property is created the only property setting you can change is the display name.

Sub-type of Profile

Please select the sub-type of user profiles with which you want to associate this profile property.

User Description

Specify a description for this property that will provide instructions or information to users. This description appears on the Edit Details page.

Policy Settings

Specify the privacy policy you want applied to this property. Select the Replicate check box if you want the property to display in the user info list for all sites. To replicate properties, the default privacy must be set to

Name: \*  
TOGO-Audience-Departme

Display Name: \*  
Audience department

Edit Languages

Type:  
string (Single Value)

Length:  
25

☒ Configure a Term Set to be used for this property

Pick a Term Set for this property:  
Departments

Department  
Job Title  
Location  
Applications  
Articles  
Communities  
Departments  
Documents  
Events  
Locations  
Quick post  
Tags  
Department  
Tags

Policy Setting:  
Optional

Default Privacy Setting:  
Only Me

☐ User can override

☐ Replicable

TOGO-Audience-Department

Profile Subtype

See below the details needed to create the user property involved in the location audience targeting: TOGO-Audience-Location

Use this page to add a property for user profiles.

\* Indicates a required field

Property Settings

Specify property settings for this property. The name will be used programmatically for the property by the user profile service, while the display name is the label used when the property is shown. After the property is created the only property setting you can change is the display name.

Sub-type of Profile

Please select the sub-type of user profiles with which you want to associate this profile property.

User Description

Specify a description for this property that will provide instructions or information to users. This description appears on the Edit Details page.

Policy Settings

Specify the privacy policy you want applied to this property. Select the Replicate check box if you want the property to display in the user info list for all sites. To replicate properties, the default privacy must be set to Everyone and the User can override check box must not be selected.

Edit Settings

Specify whether users can change the values for this property in their user profile. Users with the Manage Profile permission can edit any property value for any user.

TOGO-Audience-Location

Name: \*  
TOGO-Audience-Location

Display Name: \*  
Audience location  
Edit Languages

Type:  
string (Single Value)

Length:  
25

☒ Configure a Term Set to be used for this property  
Pick a Term Set for this property:  

▼

Department  
Job Title  
Location  
Applications  
Articles  
Communities  
Departments  
Documents  
Events  
Locations  
Quick post  
Tags  
Department  
Tags

Profile Subtype

Policy Setting:  
Required

Default Privacy Setting:  
Only Me  
☐ User can override  
☐ Replicable



☐ Allow users to edit values for this property

Once created the two required user properties, you should view them in the “Custom properties” section as depicted below:

> Custom Properties	^v	Section
OfficeGraphEnabled	^v	boolean
SPS-UserType	^v	integer
SPS-HideFromAddressLists	^v	boolean
SPS-RecipientTypeDetails	^v	big integer
DelveFlags	^v	integer
VideoUserPopUp	^v	string (Single Value)
PulseMRUPeople	^v	string (Single Value)
msOnline-ObjectId	^v	unique identifier
SPS-PointPublishingUrl	^v	URL
SPS-TenantInstancelId	^v	unique identifier
SPS-SharePointHomeExperienceState	^v	integer
SPS-RefreshToken	^v	string (Single Value)
SPS-MultiGeoFlags	^v	integer
PreferredDataLocation	^v	string (Single Value)
Audience department	^v	string (Single Value)
Audience location	^	string (Single Value)



From now on, the new two user properties will be displayed at the end of any user’s property profile page and both properties should be associated to the corresponding termset.

Time Zone:	<div><div></div></div>	<div>Everyone</div>
	Select the time zone for your current location. We will use this information to show the local time on your profile page.	
Choose your settings:	<div><div><div></div> Always use regional settings defined by site administrators.</div><div><div></div> Always use my personal settings</div></div>	<div>Only Me</div>
Locale:	<div><div></div></div>	<div>Only Me</div>
	Select a locale from the list to specify the way sites display numbers, dates, and time.	
Set Your Calendar:	<div><div></div></div>	<div>Only Me</div>
	<div><div></div> Show week numbers in the Date Navigator.</div> Specify the type of calendar.	
Enable An Alternate Calendar:	<div><div></div></div>	<div>Only Me</div>
	Specify a secondary calendar that provides extra information on the calendar features.	
Define Your Work Week:	<div><div><div><div></div>Sun</div><div><div></div>Mon</div><div><div></div>Tue</div><div><div></div>Wed</div><div><div></div>Thu</div><div><div></div>Fri</div><div><div></div>Sat</div></div></div>	<div>Only Me</div>
	First day of week: <div><div></div></div> Start time: <div><div></div></div>	
	First week of year: <div><div></div></div> End time: <div><div></div></div>	
	Select which days comprise your work week and select the first day of each work week.	
Time Format:	<div><div></div></div>	<div>Only Me</div>
	Specify whether you want to use 12-hour time format or 24-hour format.	
Use language and regional settings:	<div><div><div></div></div></div>	<div>Only Me</div>
	Specify whether language and regional settings can be synchronized with site collections.	
OfficeGraphEnabled:	<div><div><div></div></div></div>	<div>Everyone</div>
DelveFlags:	<div><div></div></div>	<div>Everyone</div>
PulseMRUPeople:	<div><div></div></div>	<div>Only Me</div>
 SPS-TenantInstanceld:	<div><div></div></div>	<div>Everyone</div>
 SPS-	<div><div>19247</div></div>	<div>Everyone</div>
SharePointHomeExperienceState:	<div><div></div></div>	
SPS-MultiGeoFlags:	<div><div></div></div>	<div>Everyone</div>
Audience department:	<div><div></div></div>	<div>Only Me</div>
Audience location: *	<div><div></div></div>	<div>Only Me</div>

## Verification

Open your preferred browser

For internet explorer you need to configure it as depicted below

