## Self-hosting AH documentation (eng)

Steps to create App Registration: [Quickstart: Register an app in the Microsoft identity platform - Microsoft identity platform | Microsoft Learn](https://learn.microsoft.com/en-us/entra/identity-platform/quickstart-register-app?tabs=certificate)

## Configuring Power Catalog (eng)

Follow the steps mentioned here to set up and administer a Power Catalog: [Administer the catalog - Power Platform | Microsoft Learn](https://learn.microsoft.com/en-us/power-platform/admin/administer-catalog#set-up-the-catalog)

## Connecting AH to Power Catalog (eng)

Add the URL of the required environment in appsettings.production.json under Dataverse section for the property CatalogEnvUrl.

## Adding DAW templates to Power Catalog (eng)

Install Pac cli. Use this documentation: [Microsoft Power Platform CLI - Power Platform | Microsoft Learn](https://learn.microsoft.com/en-us/power-platform/developer/cli/introduction?tabs=windows)

In the repository, Templates folder has all the Catalog items that need to be submitted in your target catalog.

In the Templates folder, each catalog item has a SubmissionDoc.json and a zip file.

Update the SubmissionDoc.json for all the catalog items. Add publisher ID, publisher name, Engineering contact details and support contact details.

Use the following command for submitting catalog items. Submit one catalog item at a time.

pac catalog submit -env <envUrl> -p <SubmissionDoc.json> -pz <catalogItem.zip>

# **Add Catalog Extension Metadata zip**

Import the CatalogExtension\_1\_0\_1\_3.zip present in the repository, in the environment where the catalog is created.

# **Deployment of Azure Scripts (Bicep)**

## To Do Prerequisites:

### Clone the repository

Clone the repository and open it in Visual Studio Code. [FND-NEW-TCA-TaskCompletionTeams - Repos](https://microsoftit.visualstudio.com/OneITVSO/_git/FND-NEW-TCA-TaskCompletionTeams)

### Create an enterprise subscription

1. Create an azure subscription or use an existing subscription. ([Create an Enterprise Agreement subscription - Azure Cost Management + Billing | Microsoft Learn](https://learn.microsoft.com/en-us/azure/cost-management-billing/manage/create-enterprise-subscription) )
2. Make a note of the subscription ID

### Create an app registration

1. Create an application registration ([Quickstart: Register an app in the Microsoft identity platform - Microsoft identity platform | Microsoft Learn](https://learn.microsoft.com/en-us/entra/identity-platform/quickstart-register-app?tabs=certificate) )
2. While creating the app registration, use the default options.
3. Make a note of client ID and tenant ID from the app registration

### Create a service connection in Azure DevOps

Create a service connection ([Service connections - Azure Pipelines | Microsoft Learn](https://learn.microsoft.com/en-us/azure/devops/pipelines/library/service-endpoints?view=azure-devops#create-a-service-connection) )

Use the following details, while creating a service connection.

1. Type: Azure Resource Manager
2. Authentication Method: Service Principal (automatic)
3. Scope Level: Subscription
4. Resource Group: (none)
5. Service connection name: <Give your own name>

Make a note of service connection name.

### Create an environment in Azure DevOps

Create an environment ([Create and target environments - Azure Pipelines | Microsoft Learn](https://learn.microsoft.com/en-us/azure/devops/pipelines/process/environments?view=azure-devops#create-an-environment) )

Use the following details, while creating an environment.

* + 1. Name: <Give your own name>
    2. Resource: None

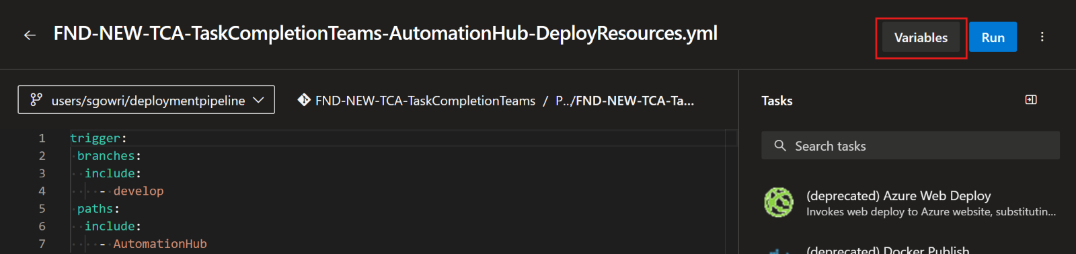
Go to ‘Approvals and Checks’, and create an ‘Approvals’ check. Add people who may have to approve before deploying code into the resources.

Make a note of environment name.

### Create the pipeline

1. Navigate to the Azure Pipelines and create a new pipeline and select a location for the pipeline.
2. Select the code location as ‘Azure DevOps’ and select the repository ‘  
   FND-NEW-TCA-TaskCompletionTeams’
3. Configure the YAML pipeline by selecting an existing YAML file ‘FND-NEW-TCA-TaskCompletionTeams-AutomationHub-DeployResources.yml’ and click ‘Review Pipeline’
4. Save the pipeline.

### Add Pipeline variables



* + - 1. Open the pipeline in Azure DevOps
      2. Click on variables and add the following variables.
      3. Add the values for all the variables. Service Connection and environment names are the same used while creating them in the previous steps.

|  |  |
| --- | --- |
| Variable Name | Can be referenced from |
| poolName | Name of the pool that should be used to run the pipeline (Open you Azure DevOps. Go to Project settings-> Agent Pool to get this information) |
| serviceConnection | (Create a service connection in Azure DevOps) |
| Environment | Create an environment in Azure DevOps |
| ClientId | Create an app registration |
| TenantId | Create an app registration |
| SubscriptionId | Create an enterprise subscription |
| appService | Give your own name |
| appServicePlan | Give your own name |
| applicationInsights | Give your own name |
| storageAccount | Give your own name (*Storage account name must be between 3 and 24 characters in length and use numbers and lower-case letters only.)* |
| ResourceGroupName | Give your own name |
| location | Give location of your preference where the resource group and resources should be present (eg: westus2) |
| catalogEnvUrl | URL of the environment associated with the catalog |
| catalogPublisherId | ID of the catalog publisher created |
| teamsAppId | Keep this empty. This variable should be set later. |
| feedbackFormUrl | URL to your feedback form |
| faqUrl | URL where FAQs are going to be present |
| adminEnvironment | Default Environment ID |
| AppInsightsConnectionString | Keep this empty. This variable should be set later. |

### Role Assignment

The service connection creates the resource group and resources when the pipeline is ran using bicep templates.

Give the following roles to the service connection at subscription level.

[Assign Azure roles using the Azure portal - Azure RBAC | Microsoft Learn](https://learn.microsoft.com/en-us/Azure/role-based-access-control/role-assignments-portal)

Go to your subscription, click on the ‘Access Control (IAM)’ and assign the following roles

1. ‘Corp Tenant DevOps Role’ //MS specific
2. ‘Contributor’
3. ‘Role Based Access Control’ (Revoke this access after deploying resources)

### Run pipeline

1. Run the pipeline [Pipelines - Runs for FND-NEW-TCA-TaskCompletionTeams-AutomationHub-Deploy.yml](https://dev.azure.com/MicrosoftIT/OneITVSO/_build?definitionId=71925&_a=summary) with the updated variables. //This link needs to be updated. Added this only for testing purpose.
2. Make sure to tick the checkbox to Deploy resources.
3. Permit use of service connection when prompted.
4. Permit the environment to access/run the pipeline.

### Role Assignment

When the pipeline ran successfully, assign the following role.

Go to the storage account that got created in the resource group, click on the ‘Access Control (IAM)’.

Assign the following role ‘Storage Blob Table Contributor’ to the ‘appService’ that has been created in the resource group in the Azure subscription.

### Update Manifest File

1. Open the repository in Visual Studio Code and navigate to ‘*AutomationHub.Client/manifest’* directory. This directory has a manifest.json file, color.png and outline.png
2. Open manifest.json and replace the following metadata in the manifest file
   1. Look for <<APPSERVICE>> in the json file. Replace it with name of the appService. Give the same name as given in step 5 - Update the pipeline variables
   2. Look for <<CLIENTID>> in the json file. Replace it with Client ID. Give the same value as in step 2 - Create an app registration
3. Save the json file and zip the manifest directory.

### Update App Registration

1. From the manifest, copy the “webApplicationInfo: resource“.
2. Open the app registration we created earlier, and click on Manage>Expose an API.
3. Add Application ID URI, and paste the value we copied from manifest file and save it.
4. In the "Expose an API" screen, add a scope `access\_as\_user`, with consent set to Admin only.
5. In the "Expose an API" screen, under Authorized client applications, add client applications "1fec8e78-bce4-4aaf-ab1b-5451cc387264", "5e3ce6c0-2b1f-4285-8d4b-75ee78787346" with the above scope.
6. Go to "Manifest" and update `accessTokenAcceptedVersion` key to 2

### Sideload App

Use the following instructions for sideloading the zip file

1. Open teams app and choose Apps in the left side pane.
2. Click on Manage your Apps and select Upload an app.
3. Upload the created zip file.

### Publish App to Teams:

[Publish App on Teams Store - Teams | Microsoft Learn](https://learn.microsoft.com/en-us/microsoftteams/platform/concepts/deploy-and-publish/appsource/publish)

Use the instructions in the above link and publish the teams app. Please note that you may need to reach out to the admin team to understand any tenant-specific steps.

### Note App Insights Connection String Value:

Open the application insights that got created. Open Configure->Properties and make note of the connection string present there.

### Add Pipeline variable

1. Open the pipeline in Azure DevOps
2. Click on variables and update the value the following variable, after publishing the app to teams.

|  |  |
| --- | --- |
| Variable Name | Can be referenced from |
| teamsAppId | Teams App ID from the publishing process |
| AppInsightsConnectionString | Application Insights Connection String noted from previous step |

### Pac cli commands

pac catalog submit -p "AutoManageIncomingMeetingInvites.json" -sz "AutoManageIncomingMeetingInvites.zip"

### Troubleshooting

1. While running the pipeline, if you see an error of the kind: MissingSubscriptionRegistration: The subscription is not registered to use namespace 'Microsoft.Web'.  
   Follow the steps mentioned here: [Resource provider registration errors - Azure Resource Manager | Microsoft Learn](https://learn.microsoft.com/en-us/azure/azure-resource-manager/troubleshooting/error-register-resource-provider?tabs=azure-portal)