

SharePoint Tools

App Catalog Management Solution

|  |
| --- |
| *Prepared by*  Gunjan Datta  12-2-2021  *Contributor*  Dade Register |

App Catalog Management Solution

Contents

[1. Overview 3](#_Toc89371348)

[2. Requirements 3](#_Toc89371349)

[2.1 Installation 3](#_Toc89371350)

[2.1.1 Upload Solution Assets 3](#_Toc89371351)

[2.1.2 Create Dashboard Page 4](#_Toc89371352)

[2.1.3 Configure Document Set Homepage 7](#_Toc89371353)

[3. Solution 11](#_Toc89371354)

[3.1 User Agreement 11](#_Toc89371355)

[3.2 Configuration 12](#_Toc89371356)

[3.3 Main Dashboard 13](#_Toc89371357)

[3.3.1 Admin Navigation Menu 13](#_Toc89371358)

[3.3.2 Developer Navigation Menu 14](#_Toc89371359)

[3.3.2.1 Add/Update an App 14](#_Toc89371360)

[3.3.2.2 Filters 15](#_Toc89371361)

[3.3.2.3 Refresh 16](#_Toc89371362)

[3.3.3 Dashboard Actions 17](#_Toc89371363)

[3.4 App Dashboard 17](#_Toc89371364)

[3.4.1 Navigation Options 17](#_Toc89371365)

[3.4.1.1 Breadcrumb 17](#_Toc89371366)

[3.4.1.2 Documents 18](#_Toc89371367)

[3.4.1.2.1 Templates 18](#_Toc89371368)

[3.4.1.3 Help 19](#_Toc89371369)

[3.4.2 Application Dashboard Actions 19](#_Toc89371370)

[3.5 Application Processes 20](#_Toc89371371)

[3.5.1 Developer Access Request 20](#_Toc89371372)

[3.5.2 Application Life Cycle 20](#_Toc89371373)

[3.5.3 Developer App Submission 21](#_Toc89371374)

[3.5.4 Developer App Review/Approval 22](#_Toc89371375)

[3.5.5 Approver App Deployment/Retraction 23](#_Toc89371376)

[3.5.6 App Support 24](#_Toc89371377)

# Overview

This document will give an overview of the “App Catalog Management” proof of concept solution for SharePoint On-Premise and Online environments. This solution will allow development groups to help manage custom SharePoint applications requests for the tenant app catalog. A main group identified as approvers will have the ability to approve and deploy/retract solutions from the tenant app catalog.

# Requirements

This JavaScript (client-side) solution is designed to run in a SharePoint 2013+/Online environment. It utilizes the following features in SharePoint:

* Document Set Content Type
  + Site Collection Feature
* Security Groups
  + App Developers – Ability to add app requests
  + App Approvers – Ability to approve and deploy apps to the tenant app catalog
    - Must have security rights to the tenant app catalog

Currently, the solution utilizes a classic webpart page and a content editor webpart. The solution can be updated as a SPFx solution.

## Installation

The solution can be deployed to a classic page or modern page. This section will go over both options.

### Classic Page

This section will go over the steps to install the solution to a classic SharePoint page. In order to use this solution it will require “Custom Scripts” to be enabled on the site collection.

#### Upload Solution Assets

1. Access the site collection to install the solution to
   1. It’s recommended to create a new site collection for this solution
2. Upload the SharePoint solution assets to the “Site Assets” library

Table

Description automatically generated

Figure - Solution Assets

* 1. app-catalog-dashboard.js/app-catalog-dashboard.min.js
     1. JavaScript solution bundled file
  2. config.json
     1. Configuration specific to the client’s environment
  3. dashboard.aspx
     1. Web Part page for the solution (Created in the next step)
  4. index.html
     1. Referenced by the content editor webpart

#### Create Dashboard Page

1. Access the “Site Pages” or “Pages” library
2. Select the “New” menu button and select “Web Part Page”

Graphical user interface, application

Description automatically generated

Figure - Create Web Part Page

1. Set the values and click on “Create”

Graphical user interface, text, application

Description automatically generated

Figure - Web Part Page Creation

1. Click on “Add a web part” and select the “Content Editor” webpart

Graphical user interface, application, Word

Description automatically generated

Figure - Add Content Editor Web Part

1. Edit the web part and set the content link and chrome type properties

Graphical user interface, text, application

Description automatically generated

Figure – Content Editor Web Part Properties

1. Click on “Install” to create the SharePoint solution assets, and then “Security” to create the security groups used by the solution

Graphical user interface, text, application, email

Description automatically generated

Figure - Installation Modal

1. Once the installation completes, click on the “x” button to close the modal and select “Stop Editing” from the ribbon

Graphical user interface, application, Word

Description automatically generated

Figure - Complete Web Part Page Edits

### Modern Page

This section will go over the installation of the SPFx package and initial setup of the solution on a modern page. The deployment option will require the site collection’s app catalog to be enabled.

#### Deploy Solution

1. Access the “Site Assets” or “Documents” library to store the json configuration
2. Click on the “Upload” button and select the json configuration file to reference later

Table

Description automatically generated

Figure - Upload JSON Configuration

1. Access the “App Catalog”
2. Click on the “Upload” button

Graphical user interface, text, application

Description automatically generated

Figure - Upload the SPFx Package

1. Select the package file and click on “OK” to upload it to the catalog

Graphical user interface, application

Description automatically generated

Figure - Select the SPFx Package

1. Select the checkbox to automatically make the solution available and click the “Deploy” button

Graphical user interface, application, Teams

Description automatically generated

Figure - Deploy the SPFx Solution

#### Create Dashboard Page

1. Access the “Pages” library
2. Select “New” and then “Site Page”

Graphical user interface, text, application, email

Description automatically generated

Figure - Create Site Page

1. Update the page information (Title, Author, etc)
2. Click on the “+” icon to add the webpart

Graphical user interface, application

Description automatically generated

Figure - Add Modern WebPart

1. Click on the “X” icon to close the installation dialog

Graphical user interface, text, application, email

Description automatically generated

Figure - Installation Dialog

1. Click on the “Save as Draft” button to save the page

### Install Solution

This section will go over the installation of the solution assets. The order of

1. Access the classic or modern page containing the solution
2. Click “Enable Feature” if the “Document Set” feature has not been enabled
   1. The other buttons will be disabled if this feature is not enabled

Graphical user interface, text, application, email

Description automatically generated

Figure - Enable Document Set Feature

1. The configuration property must be set before we can complete the installation (SPFx Only)
   1. Close the installation dialog and edit the page
   2. Edit the webpart properties

Graphical user interface, text, application, email

Description automatically generated

Figure - Edit WebPart Properties

* 1. Set the link to the configuration json file and click on “Apply”

A picture containing square

Description automatically generated

Figure - Set Configuration Url

* 1. Close the dialog and save the page as a draft

1. Click on “Install” the create the lists and libraries used by this solution

Graphical user interface, text, application, email

Description automatically generated

Figure - Install SharePoint Assets

1. Click on the “X” to close the dialog, and click on the “Status” button to refresh the installation dialog

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure - Refresh Installation Dialog

1. Click on “Security” to create the security groups used by this solution

Graphical user interface, text, application, email

Description automatically generated

Figure - Create Security Groups

1. Page will automatically refresh and the dashboard will be visible. Click on the “Publish” button to complete the installation.

Graphical user interface, text, application, email

Description automatically generated

Figure – View App Catalog Dashboard

### Configure Document Set Homepage

1. Access the “Developer Apps” library settings

Graphical user interface, application

Description automatically generated

Figure - Developer Apps Library Settings

1. Click on the “App” content type under the “Content Types” section

Graphical user interface, text, application, email

Description automatically generated

Figure - Document Set Content Type

1. Click on “Document Set Settings”

Graphical user interface, text, application

Description automatically generated

Figure - Document Set Settings

1. Click on the “Customize the Welcome Page” link

Graphical user interface, text, application

Description automatically generated

Figure - Edit the Document Set Homepage

1. Edit the Page

Graphical user interface, application, Word

Description automatically generated

Figure - Edit Document Set Home Web Part Page

1. Delete the Image Webpart

Graphical user interface, application

Description automatically generated

Figure - Delete Image Web Part

1. Delete Document Set Properties

Text

Description automatically generated with medium confidence

Figure - Delete Document Set Properties Web Part

1. Delete Document Set Contents

Graphical user interface, text, application, email

Description automatically generated

Figure - Delete Document Set Contents

# Solution

## User Agreement

Graphical user interface, text, application

Description automatically generated

Figure - User Agreement

A user agreement is configurable through the json configuration file. It will be displayed for any user that doesn’t belong to either the “Developers” security group when trying to upload/add a new package. Clicking on “Agree” will automatically add you to the developer’s security group, and move forward with allowing the user to add the package.

## Configuration

Graphical user interface, text, application

Description automatically generated

Figure - Solution Configuration

A JSON (config.json) file is used to hold the solution’s dynamic configuration options.

### General

* appCatalogUrl
  + The url to the test site collection to be used for testing the application. This site must be have an app catalog enabled on it.
* dashboardUrl
  + The url to the dashboard page. This will be used by the PageUrl notification content replacement.
* helpPageUrl
  + The url to the help page button located in the navigation bar. If not provided, the help button will not be rendered.
* templatesLibraryUrl
  + The url to the folder containing the templates. If not provided, the templates dropdown will not be rendered.
* tenantAppCatalogUrl
  + The url to the tenant app catalog.
* userAgreement
  + The user agreement displayed for users adding packages to the dashboard.

### Status

* requiresTechReview
  + If set to true, will validate the technical review form as part of the approval process.
* requiresTestCases
  + If set to true, will validate the test cases form as part of the approval process.
* actions
  + An array of buttons to be displayed on the app dashboard.
* approval
  + An array of user type values that can approve the item.
* checklists
  + Displayed on the approval/submission form.
* createTestSite
  + If set to true, will create the test site for the application.
* notification
  + The email information to notify users on submission/approval of the item.

### Actions

* AppData
  + Displays the edit form to modify the metadata related to the app catalog.
* ApproveReject
  + Displays the approve and reject button if the user is listed in the approval property.
* DeploySiteCatalog
  + Displays the deploy to site collection button, allowing the user to deploy the application to a site of their choice.
* DeleteTestSite
  + Displays the delete test site button if the site exists.
* DeployTenantCatalog
  + Displays the option to deploy to the tenant app catalog.
* Edit
  + Displays the edit form to modify the app metadata.
* EditTechReview
  + Displays the edit form to modify the technical review assessment.
* EditTestCases
  + Displays the edit form to modify the test cases assessment.
* Submit
  + Displays the submission form to request approval.
* TestSite
  + Displays a create or view button to the test site collection.
* View
  + Displays the app metadata view form.
* ViewTechReview
  + Displays the last technical review for the application.
* ViewTestCases
  + Displays the last test cases for the application.

### Notification

* approval
  + If set to true, will send the notification when the approval button is clicked.
* submission
  + If set to true, will send the notification when the submit button is clicked.
* to
  + An array of user type values to notify.
* cc
  + An array of user type values to notify.
* subject
  + The email subject.
* content
  + The email html content.

#### Notification Content

The notification email content can be dynamically set by using the [[Internal Field Name]] template as a placeholder within it. Complex field types, like a user field, can be used to access their properties. For example, to set the sponsor's name the value [[AppSponsor.Title]] would be used as the placeholder. Other wildcard values that can be used are listed below.

* Internal Field Name
  + The item's list field internal name.
* PageUrl
  + The html link to display the item.

### User Types

* ApproversGroup
  + A reference to the approver's security group.
* Developers
  + The developers of the application. Linked to the AppOwners metadata field.
* DevelopersGroup
  + A reference to the developer's security group.
* Sponsor
  + The sponsor of the application. Linked to the AppSponsor metadata field.

### Validation

* techReview
  + The valid responses for the technical review form.
* testCases
  + The valid responses for the test cases form.

Each of the validation properties will be an object containing the form fields and valid responses.

Example: { "InternalFieldName": ["Valid Responses"] }

## Main Dashboard

Graphical user interface, application

Description automatically generated

Figure - Main Dashboard

The main dashboard will display the app requests and current status from all developers.

### Admin Navigation Menu

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure 34 - Administration Menu

The “Admin” navigation menu will be made available for the members of the approver group. This menu will provide links to the security groups used by this solution so they are able to add/remove users. The “Manage App” option will display the “Installation” modal allowing an administrator to reinstall the solution, as shown in the figure below.

Graphical user interface, text, application, email, Teams

Description automatically generated

Figure 35 - Administration Installation Menu

This function should only be used for future upgrades when new fields have been added.

### Developer Navigation Menu

The developer menu is available for all users in the developer or approver security groups.

### Add/Update an App

Graphical user interface, text, application

Description automatically generated

Figure - Add/Update an App

Clicking the “Add/Update” button will display a file upload window, allowing the user to select a SPFx package file, as shown in the figure below. The user agreement will be displayed for the user if they do not belong to the “Developer’s Group”. The user must agree to it in order to add packages to the dashboard.

A screenshot of a computer

Description automatically generated with medium confidence

Figure - Upload SPFx Package

Once validated and uploaded, a “New App” form will be displayed for the developer to fill out. Some of the metadata will be pre-filled by the data extracted from the SPFx package file.

Graphical user interface, text, application, email

Description automatically generated

Figure - New App Form

### Filters

Text

Description automatically generated

Figure - Filtering the Dashboard

Clicking on the “Filters” button will display a panel to view apps by status, as shown in the figure below.

Graphical user interface, text, application

Description automatically generated

Figure - Filters Panel

#### Refresh

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure - Refresh the Dashboard

The refresh button will load the latest data for the user.

### Dashboard Actions

Graphical user interface, text, application

Description automatically generated

Figure - Dashboard Actions

Each app will have a set of available actions. The actions will be made available based on the app’s status.

* Details
  + Redirects the user to the app dashboard
* Edit
  + Displays the item form in edit mode
* Submit
  + Submits the application for review
* Review
  + Creates/Continues an app assessment
* Reject
  + Sends the application back to the developer

## App Dashboard

Graphical user interface, application, website

Description automatically generated

Figure - App Dashboard

The application dashboard contains information about the app and all related files for the application.

### Navigation Options

#### Breadcrumb

The breadcrumb’s “App Dashboard” will redirect the user to the main dashboard when clicked.

#### Documents

Graphical user interface, text, application

Description automatically generated

Figure - Documents View

The documents view will contain all of the files related to the application. A templates folder can be linked to the application by the configuration settings. If present, a dropdown will be made available for the user.

##### Templates

Graphical user interface, text, application, chat or text message

Description automatically generated

Figure - Templates Dropdown

The templates dropdown will display the available templates. Clients can utilize this feature along with guidance set in the help page for their defined review and approval processes. Selecting a template will copy it to the app’s folder.

Graphical user interface, text, application

Description automatically generated

Figure - App Documents

Office files will have the “Edit” button made available for the user to easily open in the associated application. The user will have the ability to download and delete the files, with the exception of deleting the .sppkg file.

#### Help

Graphical user interface, text, application, email

Description automatically generated

Figure - Help Page

The optional help page that is configurable by the solution’s configuration file. This will display a help page in a new window/tab.

Figure - Application Request Assets

### Application Dashboard Actions

Graphical user interface, application

Description automatically generated

Figure - Application Dashboard Actions

The application dashboard will display additional actions which are not available on the main dashboard. The actions will be made available based on the app’s status.

* Edit
  + Displays the item form in edit mode
* Submit for Review
  + Submits the application for review
* Review
  + Creates/Continues an app assessment
* Reject
  + Sends the application back to the developer
* Delete App/Solution
  + Deletes the app request
  + This option will only be available for apps that haven’t been deployed to the app catalog
* Assessment
  + Views the last/current assessment of the application
* Approve
  + Approves the application request
* Retract
  + Retracts the app from the tenant app catalog
* Deploy to Site Catalog
  + Copies and deploys the app to the optional site collection app catalog
* Deploy to Tenant
  + Copies and deploys the app to the tenant app catalog Application Processes

## Application Processes

### Developer Access Request

This process will occur when a user tries to upload a SPFx package to the dashboard. Users who do not belong to the developer’s or approver’s security group will have the “User Agreement” displayed for them to agree to. Once they agree to the agreement, the user is automatically added to the developer’s security group.



Figure - Developer Access Request

### Application Life Cycle

The application life cycle will utilize the configuration to associate a status with user(s)/group(s) to complete that stage of the process. Once approved and deployed, a yearly review of the application will be conducted to ensure the application is still needed and an owner is still taking responsibility for the application.



Figure – Application Life Cycle

### Developer App Submission

Members of the developer’s security group will be able to submit packages for review. Some of the metadata will automatically be set based on the package being uploaded.



Figure - Developer App Submission

### Approver App Deployment/Retraction

An app can be deployed and retracted from the app catalog. Please note that retracting an app that is already in use by other sites will cause the solution to no longer work in those sites.



Figure - Approver App Deployment/Retraction

### App Support

The tool will help track the developers and owners of the application. This will assist the helpdesk in helping route issues to the appropriate users. The owner of the application is ultimately responsible for the application as part of the submission process.



Figure - App Support