Creation

Create a new folder at the targeted location that will serve as your directory locally or create a new repository in Git/GitHub. Ensure that you are in the new directory when running the command.

Command:

pac pcf init --namespace NameHere --name NameAlsoHere --template field

A screenshot of a computer

Description automatically generated

With Errors:

A screenshot of a computer program

Description automatically generated

This process takes a while and will create several files all shown in the Explorer on the left.

After running “npm install” you will notice that the red files with errors are now all green.

A screenshot of a computer

Description automatically generated

Build

Create a new folder at the targeted location that will serve as your directory locally or create a new repository in Git/GitHub. Ensure that you are in the new directory when running the command.

Commands:

Deployment of Solution

cmd: pac solution init --publisher-name myPublisherName --publisher-prefix mypre

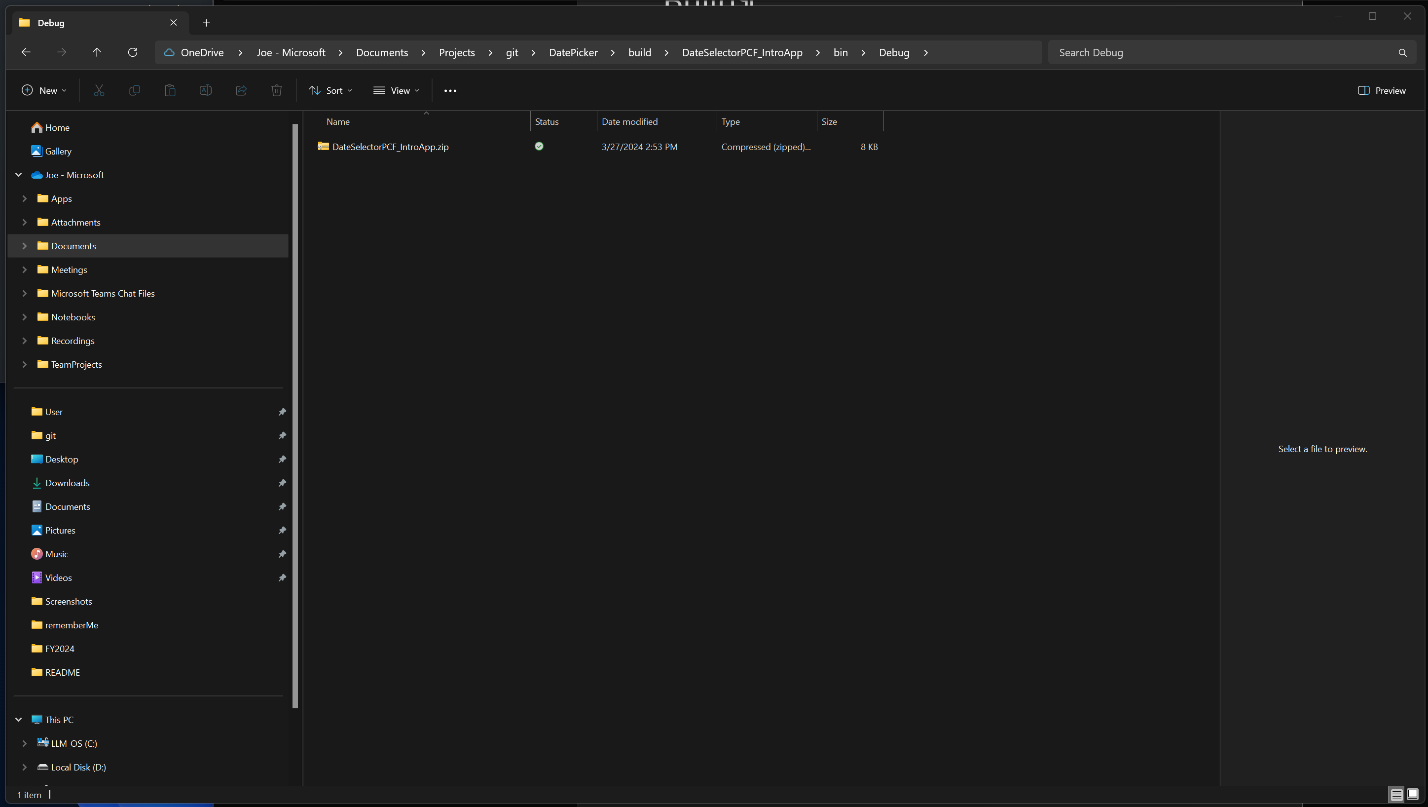
Create Solution Reference

cmd: pac solution add-reference --path “C:\Users\username\directory\andSoOn”

Build Solution (**ensure running in Admin**!!)

msbuild /t:build /restore

This last command generated the file in the targeted build folder. This file is a zipped file just like any other solution you import manually into Dynamics. This file can be uploaded, or it can be pushed by following the deployment section. You will notice that this is deploying to Debug when managed you will see a Release folder for the build.



Deployment

Create a new folder at the targeted location that will serve as your directory locally or create a new repository in Git/GitHub. Ensure that you are in the new directory when running the command.

Command:

Setup Authority

cmd: pac auth create --url <https://MYCRMSPACE.crm.dynamics.com/>

This will create a profile that is authenticated to the dynamics instance. This is the target of our published solution deployment.

cmd: pac pcf push –publisher -prefix myprefix

This deployment can take quite some time to complete, do not expect this to complete quickly. If for some reason you have issues seeing the new changes to your component you may need to delete the component in the environment before the next import.

Power Pages Integration

Create a new form and use my form that has the newly created field called “Portal Contact (Enhanced Accounts).

A screenshot of a computer

Description automatically generated

Add the desired Date/Time Field to your Form, in this example I have created a Date Time Field called “myDateTimeField”.

A screenshot of a computer

Description automatically generated

Selecting your new field on the form you should have the option to enable the code component as shown below.

A screenshot of a computer

Description automatically generated

Enabling the added Component “DateSelector” will display the properties added from the manifest. The bound is the value of the form so it does not however the other two with a usage of “input” are found on the Field Code Component form.

A screenshot of a computer

Description automatically generated

Setting DayOnly to **true** will show only the day, and this will not set a time value in the “myDateTimeField” control on my form.

A screenshot of a computer

Description automatically generated

The code component does not display on the Designer Page, the preview is the only thing displayed.

A screenshot of a computer

Description automatically generated

The code component can be properly viewed through the normal form Preview.

A screenshot of a computer

Description automatically generated

You can see the control appearance will have one look in the MDA and another in your Power Pages. The Pages Theme does affect the appearance of your control, this should be considered when building your control.

A screenshot of a computer

Description automatically generated