



SUMMIT

Document Automation with Adobe Sign for Microsoft Power Platform

Ben Vanderberg & Steve Sidell

Adobe SUMMIT 2021

LAB REFERENCE GUIDE



Adobe Sign

Content

- [Exercise 0: What You Will Need](#)
- [Exercise 1: Store signed agreement in SharePoint with Adobe Sign](#)
- [Exercise 2: Automated approval process to get e-signature with Adobe Sign](#)
- [Exercise 3: Automated document OCR with Adobe PDF Tools](#)
- [Exercise 4: Automated document assembly with Adobe PDF Tools](#)

Authors



Ben Vanderberg

Principal Product Evangelist, Adobe

Twitter: [@benvanderberg](#)

LinkedIn: [in/benvanderberg](#)

Blog: [benvanderberg.medium.com](#)



Steve Sidell

Sr. Solutions Consultant, Adobe

LinkedIn: [in/stevesidell](#)

EXERCISE 0: What You Will Need

What you are going to need

- Microsoft 365 and Power Automate familiarity
- Adobe Sign knowledge
- Microsoft 365 account with access to SharePoint and Power Automate
(Basic for Adobe Sign, Premium for Adobe PDF Tools)
- Adobe Sign for enterprise or Adobe Sign developer account

What we are using

Adobe

- Adobe Sign
- Adobe PDF Tools

Microsoft

- Microsoft 365 E3
- Microsoft SharePoint Online
- Microsoft Power Automate

Getting lab resources

Download the lab resources from <https://github.com/benvanderberg/adobe-sign-pdf-tools-powerautomate-tutorial>.

Exercises 1 and 2

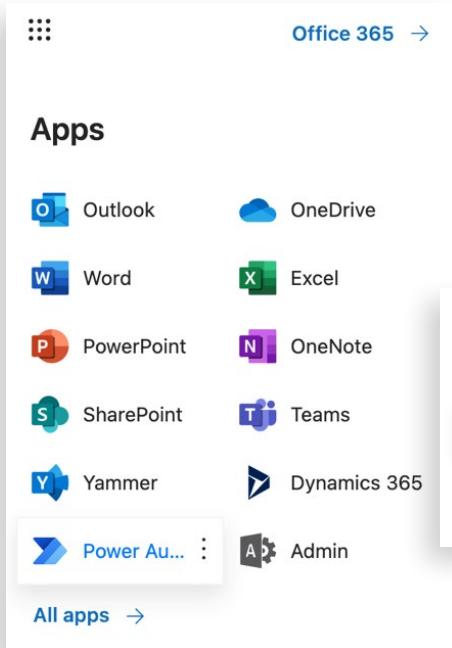
- Adobe Sign account with access to the API. A developer account or an Enterprise account.
- SharePoint site accessible by Power Automate that you have edit permissions to. Full admin access is recommended.
- Sample document for the Signature approval request and signing. (I assume there will be a download packet where I can put this?)

Exercise 3 and 4

EXERCISE 1: Store signed agreement in SharePoint with Adobe Sign

In this exercise, we will use a Power Automate Flow template to set up an automated workflow that will save all signed agreements to your SharePoint Site.

1. Navigate to Power Automate



2. Search for Adobe Sign

Take care of what's important. Automate the rest.

Create automated workflows with Microsoft Power Automate.

[Watch a quick video](#)

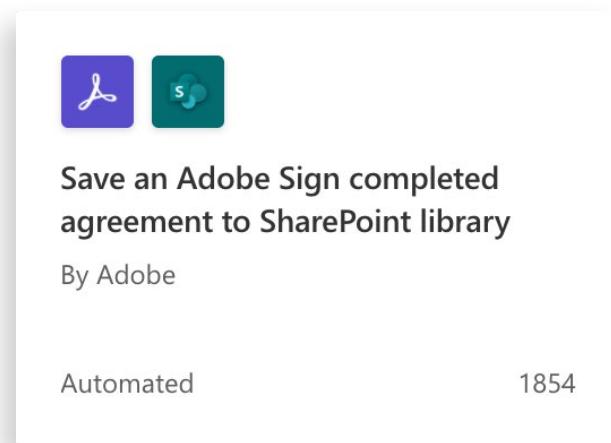
Adobe Sign



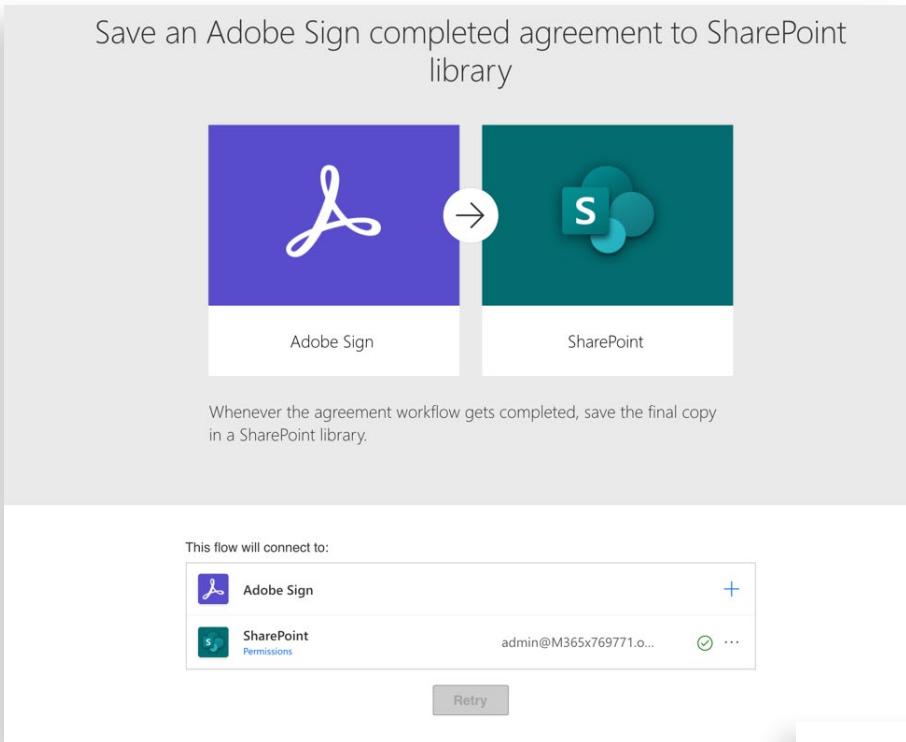
[Start from a template](#)

[Search all templates](#)

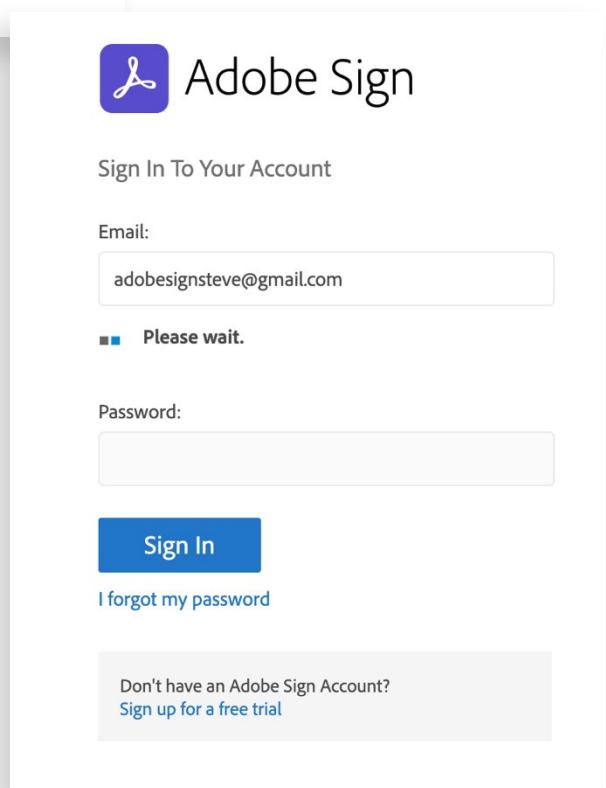
3. Choose "Save an Adobe Sign completed agreement to SharePoint library"



4. Review the screen and configure any needed connections



- a. Adobe Sign Connection
 - i. Click the blue Plus
 - ii. In the new window, enter your Adobe Sign account email then hit "tab" or click the password field and wait a moment for Adobe to check your account.
Note: this check will route you to the appropriate login if you are using an Adobe ID or our corporate SSO.
 - iii. Complete login



5. Click **Continue** to go to the Flow editing screen. There are two steps here for us to edit.

6. In the first step, the trigger, Give it a name.

The screenshot shows the 'Save Completed Agreements to SharePoint' trigger configuration. It includes fields for Name (Save Completed Agreements to SharePoint), Scope (Any of my agreements), and Agreement Name/ID (Specify only when "Scope" is "A specific agreement". Select an agreement). A 'Show advanced options' link is also present.

a. **Name:** Give the trigger a name

7. In the second step configure your SharePoint settings

The screenshot shows the 'Create file' action configuration. It includes fields for Site Address (Adobe Summit - https://m365x769771.sharepoint.com/sites/AdobeSummit), Folder Path (/Shared Documents), File Name (Agreement Na... -signed.pdf), and File Content (Agreement Sig...).

a. **Site Address:** Your SharePoint Site

b. **Folder Path:** Path to the Shared Documents you want to use

c. **File Name:** Accept the default

d. **File Content:** Accept the default

8. Save the flow



9. Navigate to the flow overview screen with blue back arrow. We will test this flow in exercise 2.

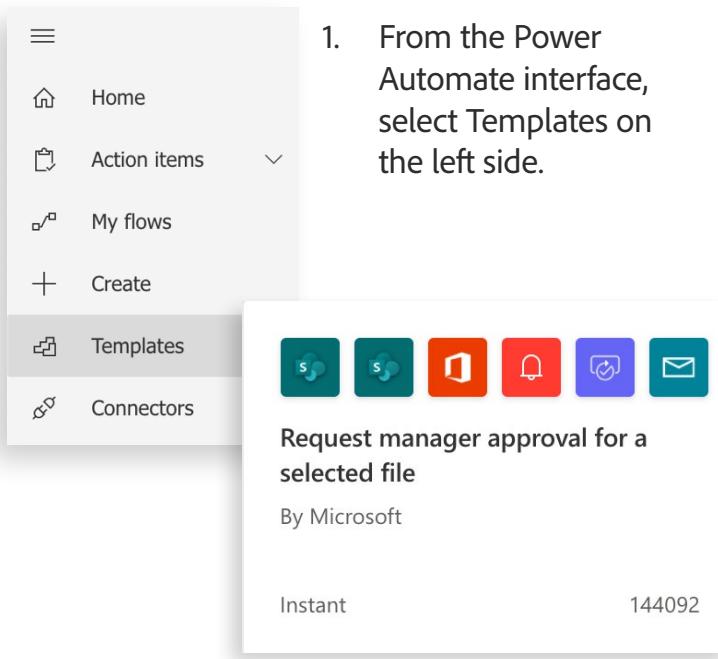


Save an Adobe Sign completed agreement to SharePoint library

EXERCISE 2:

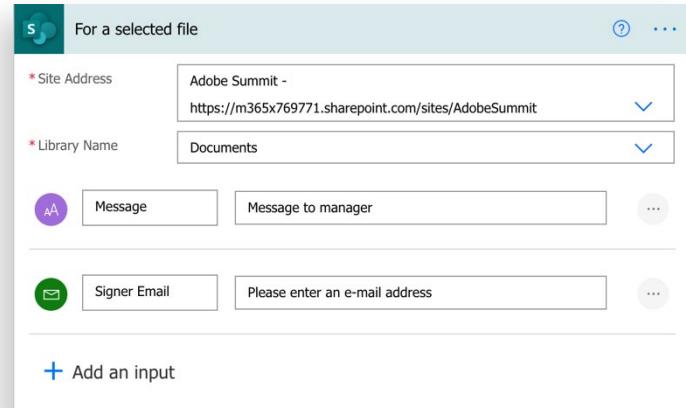
Automated approval process to get e-signature with Adobe Sign

In this exercise, we will build off the Exercise 1 with a more robust Flow and test both Flows to see them in action.

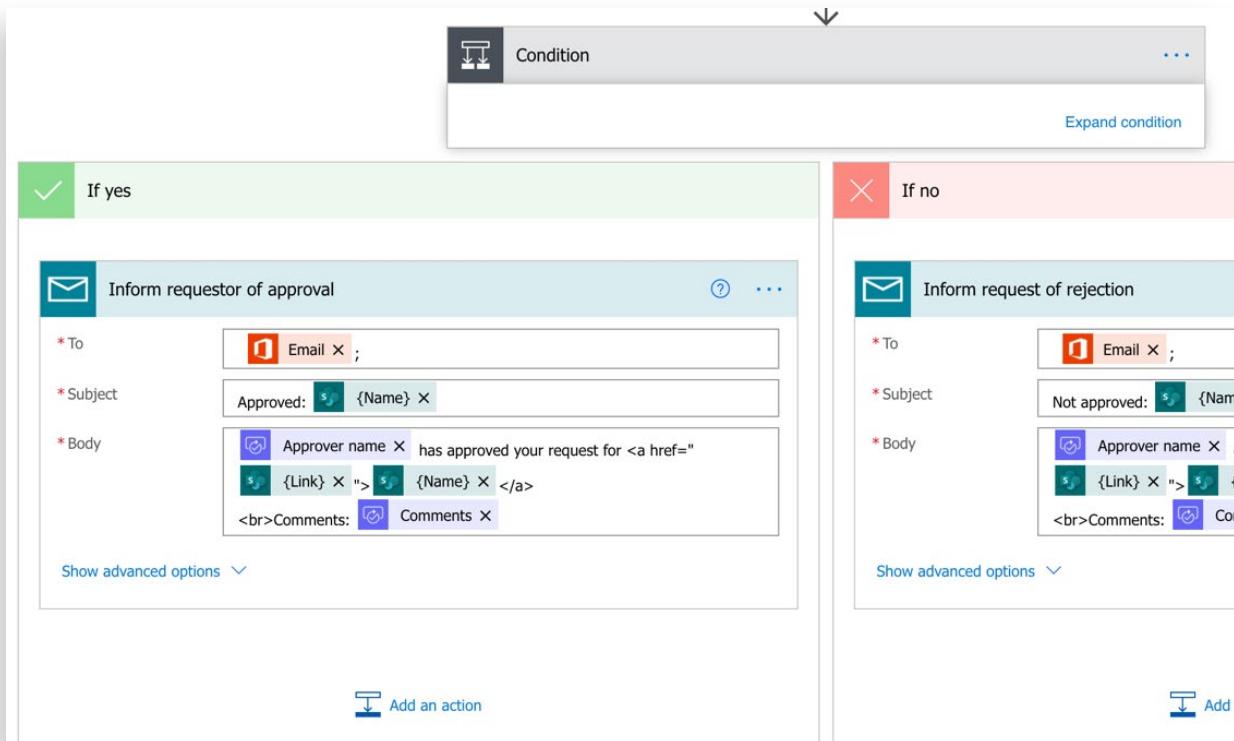


1. From the Power Automate interface, select Templates on the left side.
2. Search for "manager approval"
3. Select "Request manager approval for a selected file"
4. Review the connections and add any you are missing.
Note: if this is the first flow you are doing with approvals, they will be fully configured when the flow runs.

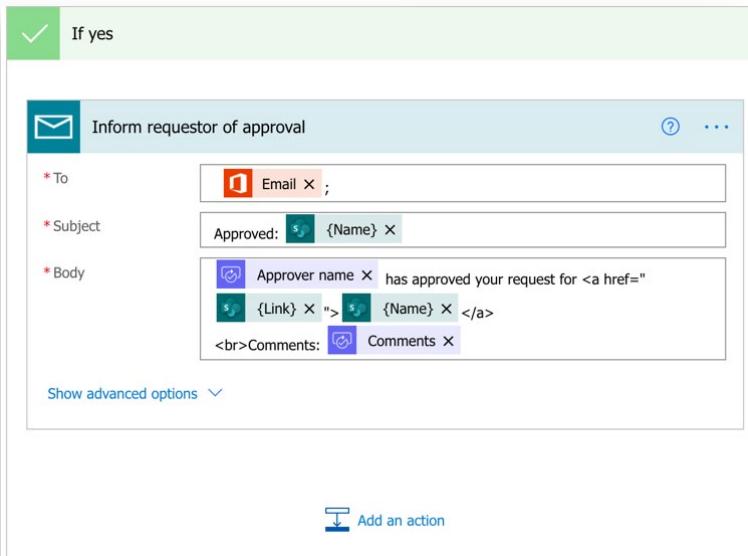
5. Click **Continue** to go to the flow editing screen. This flow has a lot of pre-configured steps including error checking and nested conditional steps.
6. In the first step "For a selected file", the trigger, make the following edits
 - a. Site Address: Your SharePoint site
 - b. Library Name: your Documents repository
 - c. Add an input:
 - i. **Type:** Email
 - ii. **Name:** Signer Email



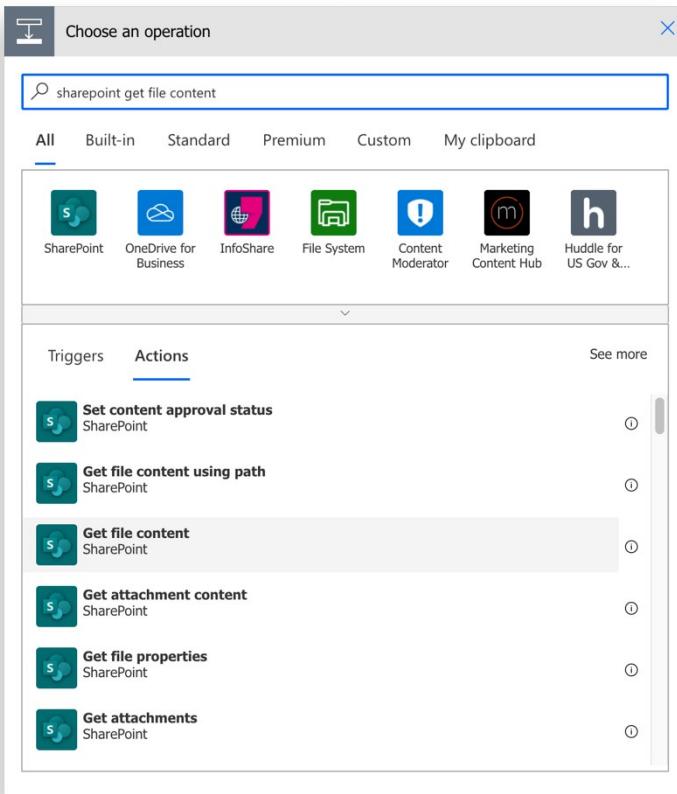
7. In the “Get File Properties” Step
 - a. **Site Address:** Your SharePoint site
 - b. **Library Name:** your Documents repository
8. Scroll down and look for “If yes”. We will add the steps to send for signature here.



9. Click “Add an action” in the “If yes” box (not the bottom most one)



10. Search for "SharePoint get file content" and choose "Get file content"

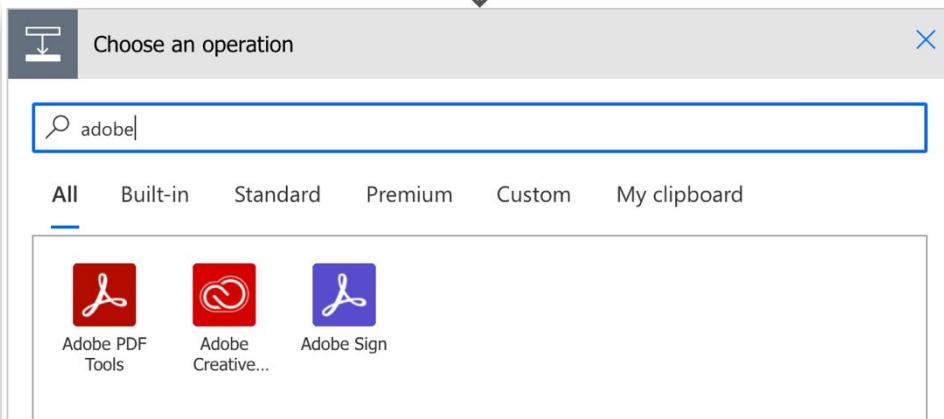


11. Fill out the Get file content information

The screenshot shows the configuration screen for the "Get file content" action. At the top, there's a logo and the title "Get file content". Below that, there are two input fields. The first field is labeled "* Site Address" and contains the value "Adobe Summit - https://m365x769771.sharepoint.com/sites/AdobeSummit". The second field is labeled "* File Identifier" and contains the value "Identifier" with a small "X" icon to its right. At the bottom left, there's a link "Show advanced options" with a dropdown arrow.

- Site Address:** Your SharePoint site
- File Identifier:** Search for "identifier" and choose Identifier from the Get file properties step

12. Add another action, search for “Adobe” and choose Adobe Sign



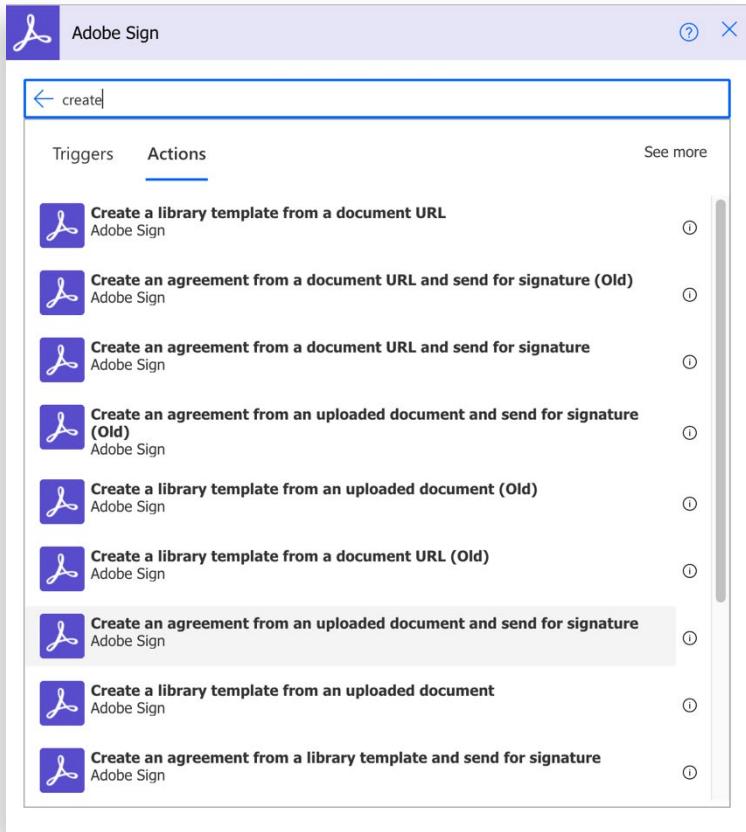
13. Enter “upload” into the search box for Adobe Sign and select “Upload a document and get a document ID”

- a. **File Name:** search for the dynamic variable **Name** to get the name of the item/document selected in the trigger.
- b. **File Content:** This one has a quirk where you need to click **Expression** in the variable assistant, add a single apostrophe, then click back to **Dynamic Content**, delete out your apostrophe, select **File Content** and then click **Ok**. Make sure that there are no additional **apostrophes** and it looks like the sample to the right.

14. Add another Adobe Sign action

- Search for "create" in the Adobe Sign search area

Look for "Create an agreement from an uploaded document and send for signature".



15. Fill out the required information

- File Name:** choose **Name** from the dynamic variable assistant
- Document ID:** Choose **Document ID** from the dynamic variable assistant
- Participant Email:** Choose **Signer Email** from the dynamic variable assistant
- Participant Order:** Enter "1"
- Participant Role:** Choose **Signer** from dropdown

16. Save the Flow

17. Go to your SharePoint site's document repository to test it out.

A screenshot of the 'Create an agreement from an uploaded document and send for signature' configuration page in Microsoft Power Automate. The page shows fields for 'Agreement Name' (set to 'Name'), 'Document ID' (set to 'Document ID'), 'Signature Type' (set to 'ESIGN'), 'Participant Email' (set to 'Signer Email'), 'Participant Order' (set to '1'), and 'Participant Role' (set to 'SIGNER'). There are also 'Add new item' buttons for each field.

18. Select the document and choose "Automate" and the Flow we just created

The screenshot shows a SharePoint 'Documents' library. A single file, 'Contoso Sample Signature.docx', is selected. A context menu is open at the top right, with 'Automate' being the last item in the sequence. A submenu for 'Power Automate' is displayed, containing options like 'Request sign-off' and 'Request manager approval for a selected file'.

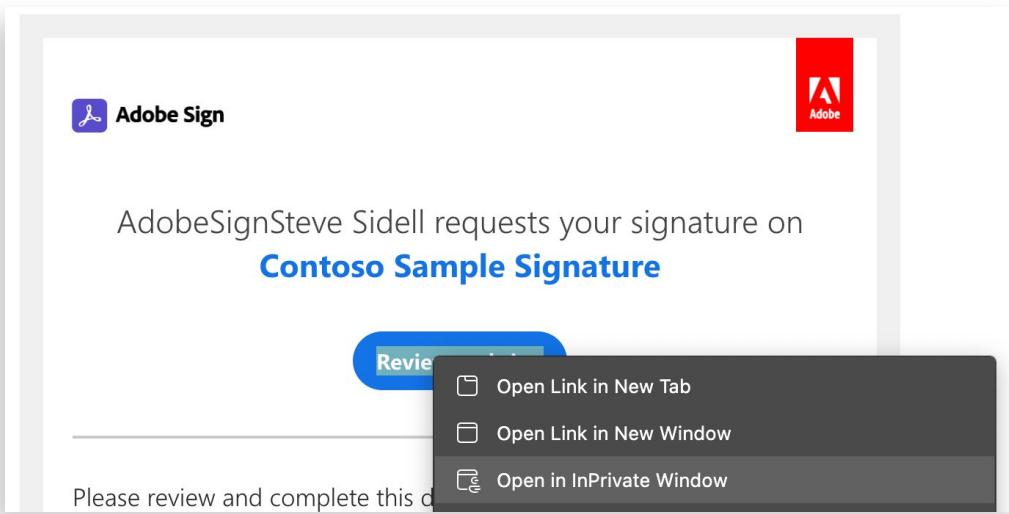
19. Validate the connections (first run of flow only)

20. Start the flow

- Message:** enter a nice message to the approver
- Signer Email:** Enter email for the document Signer
- Click Run Flow

21. The configured approver for the user starting the flow will get an approval request. You can approve through email or through The Power Automate Action Items menu.

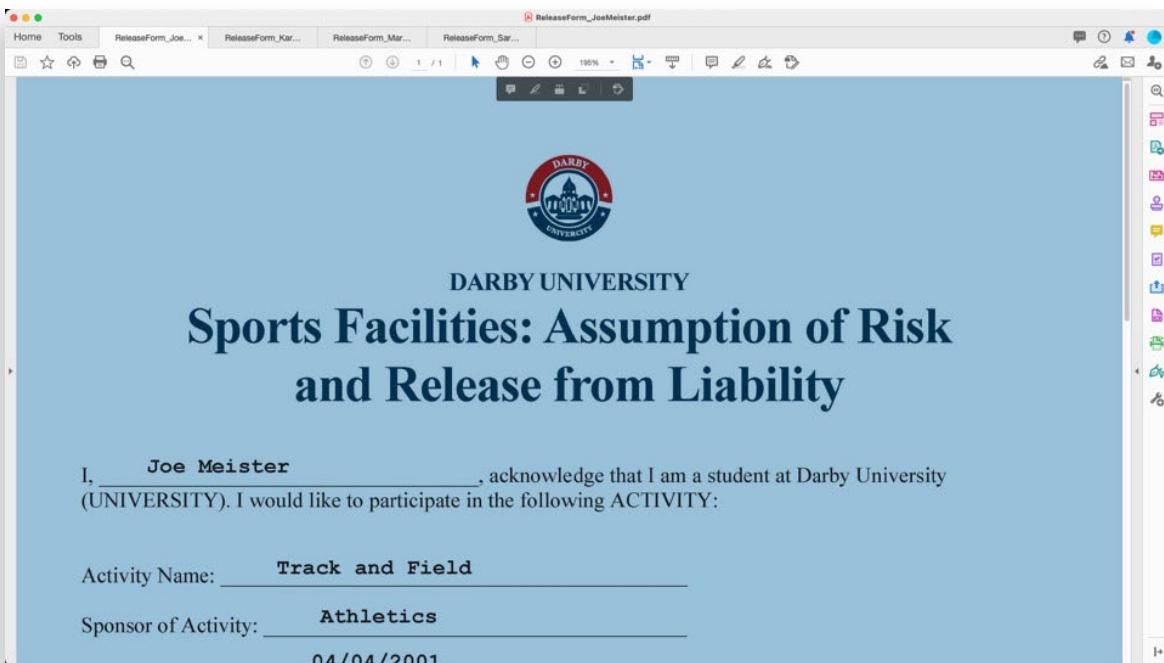
22. Once approved, sign your document. Depending on your user and if they are logged into Sign, you may need to open the signing windows in a private browser window.



23. Complete the signing and then look back in your SharePoint folder.

The screenshot shows the 'Adobe Summit' SharePoint site. In the 'Documents' library, there are two files: 'Contoso Sample Signature.docx' (modified March 10 by Adobe Sign) and 'Contoso Sample Signature-signed.pdf' (modified a few seconds ago by Adobe Sign).

EXERCISE 3: Automated document OCR with Adobe PDF Tools



In this exercise, we will learn how you can automate OCRing PDFs when they are imported into Microsoft SharePoint. This is to address the issue that commonly occurs with scanned PDF documents that are not searchable in SharePoint.

Setup a folder in SharePoint

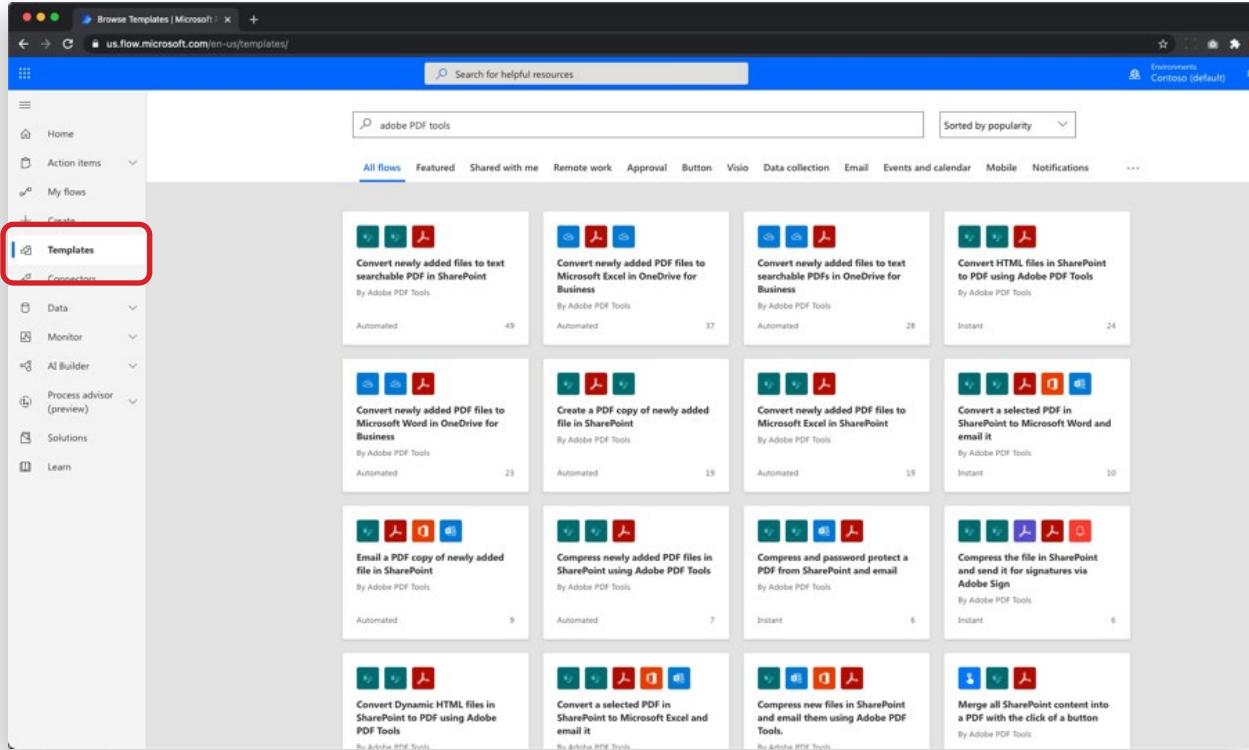
1. Go to Microsoft SharePoint where you would like to store documents.
2. In SharePoint, click on + New to create a new folder called "Processed Contracts".
3. In SharePoint, click on + New to create a new folder called "Old Contracts".

The screenshot shows a SharePoint library named "Sales Operations" (Public group). The left navigation bar includes links for Home, Conversations, Documents, Shared with us, Notebook, Pages, Employee onboarding, Site contents, Recycle bin, and Edit. The main area displays a grid of files under "Documents > E03.OCR". The grid has columns for Name, Language, Brand, Modified, Modified By, and Sign-off status. Two items are listed: "Old Contracts" and "Processed Contracts", both modified on March 10 by Ben Vanderberg.

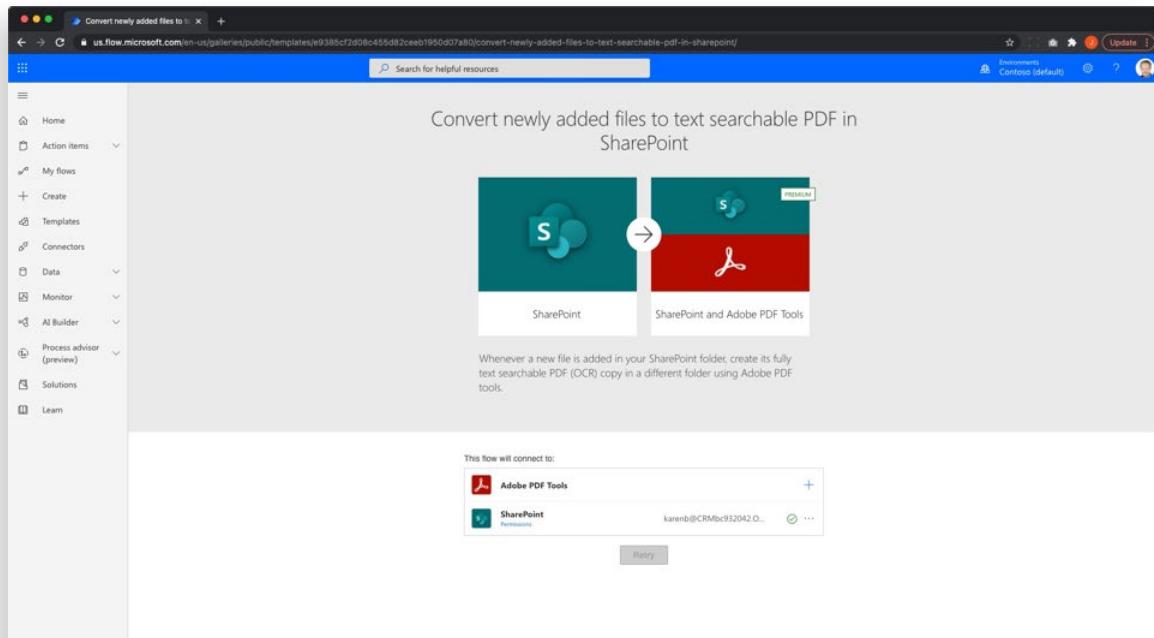
These folders were now be used to reference as part of your Power Automate flow.

Create a flow from a template

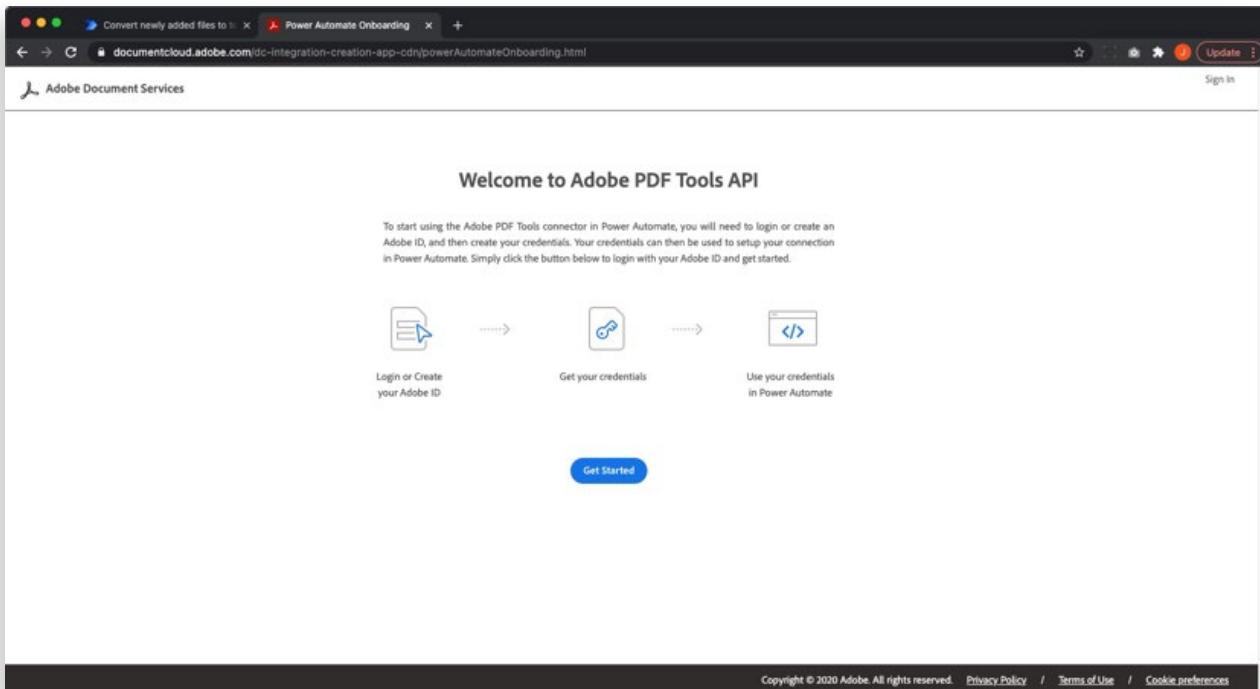
1. Go to <https://flow.microsoft.com> and log in.
2. In the sidebar, click on Templates.



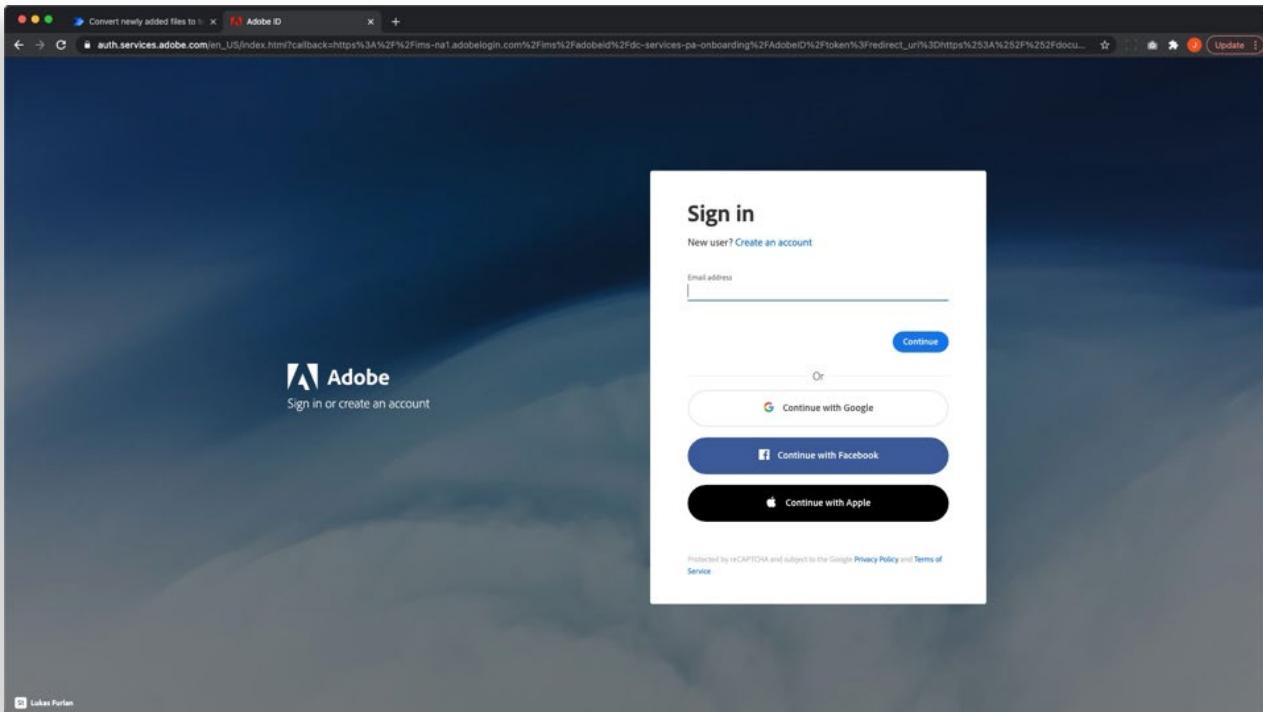
3. Select the first template called *Convert newly added files to text searchable PDF in SharePoint*.
4. Next to Adobe PDF Tools, click on the + symbol.



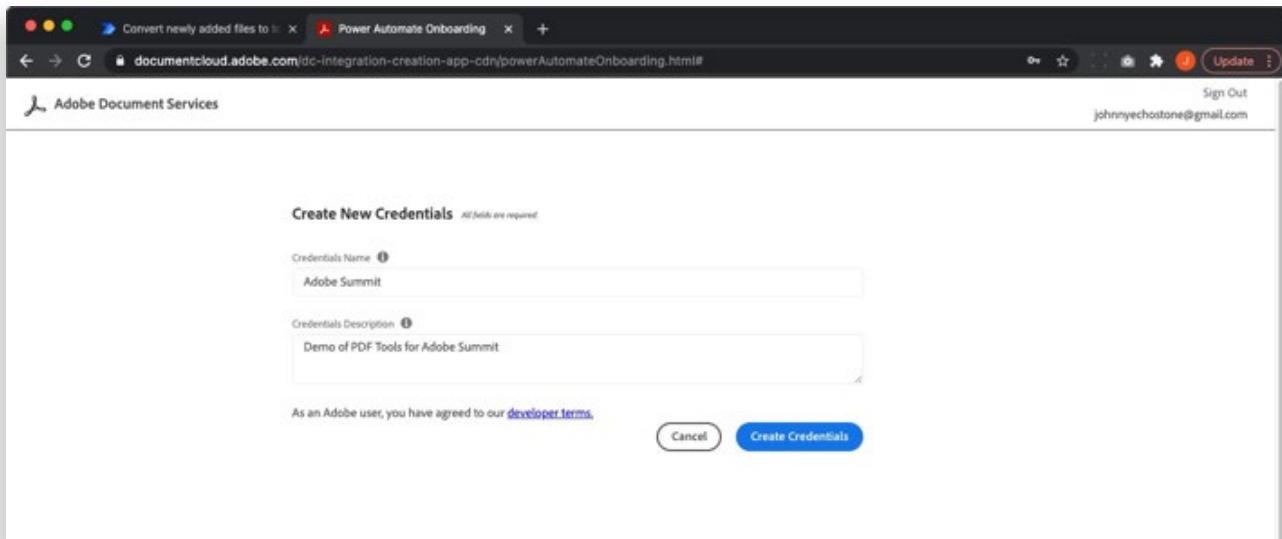
5. In a new tab, navigate to https://www.adobe.com/go/powerautomate_getstarted.



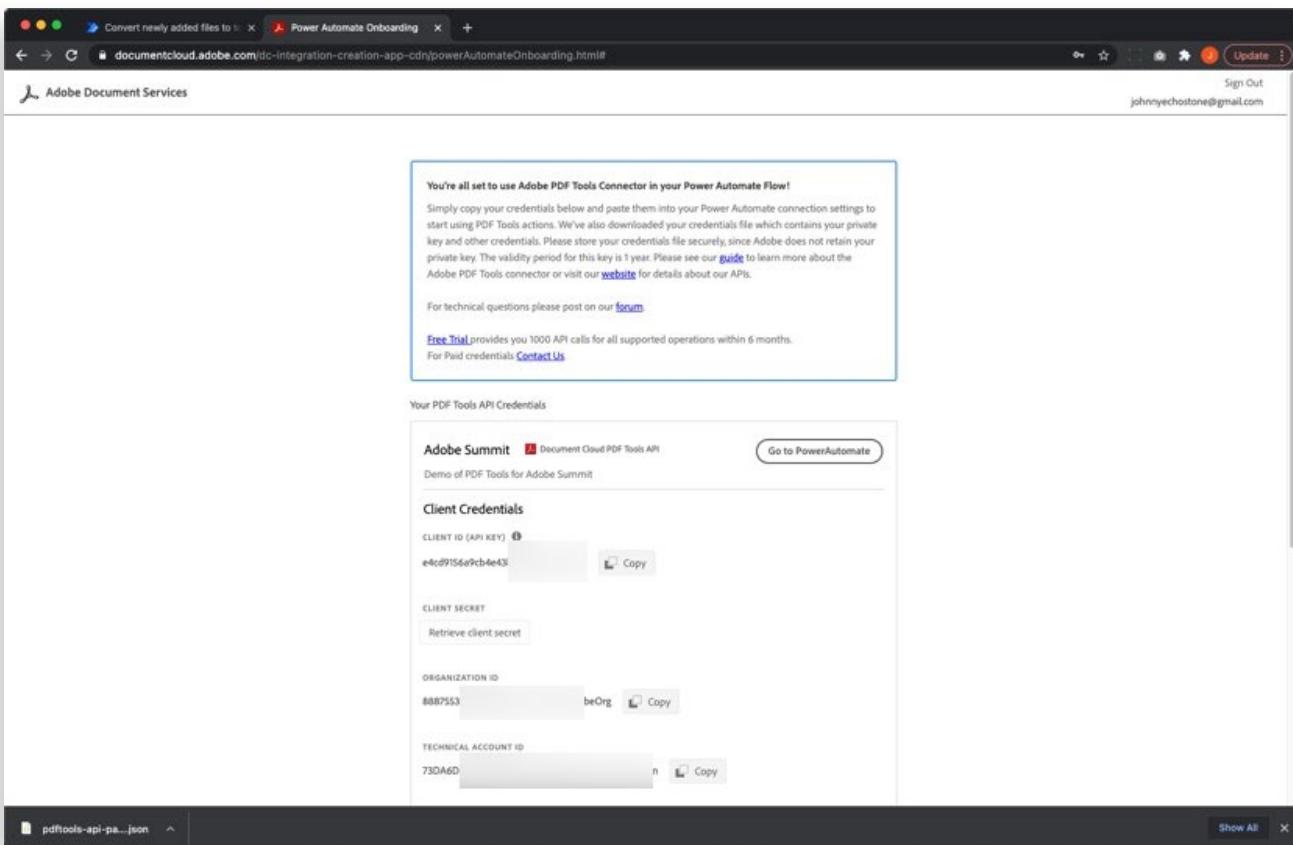
6. Click on Get Started.
7. You will be prompted to log in with your Adobe ID.



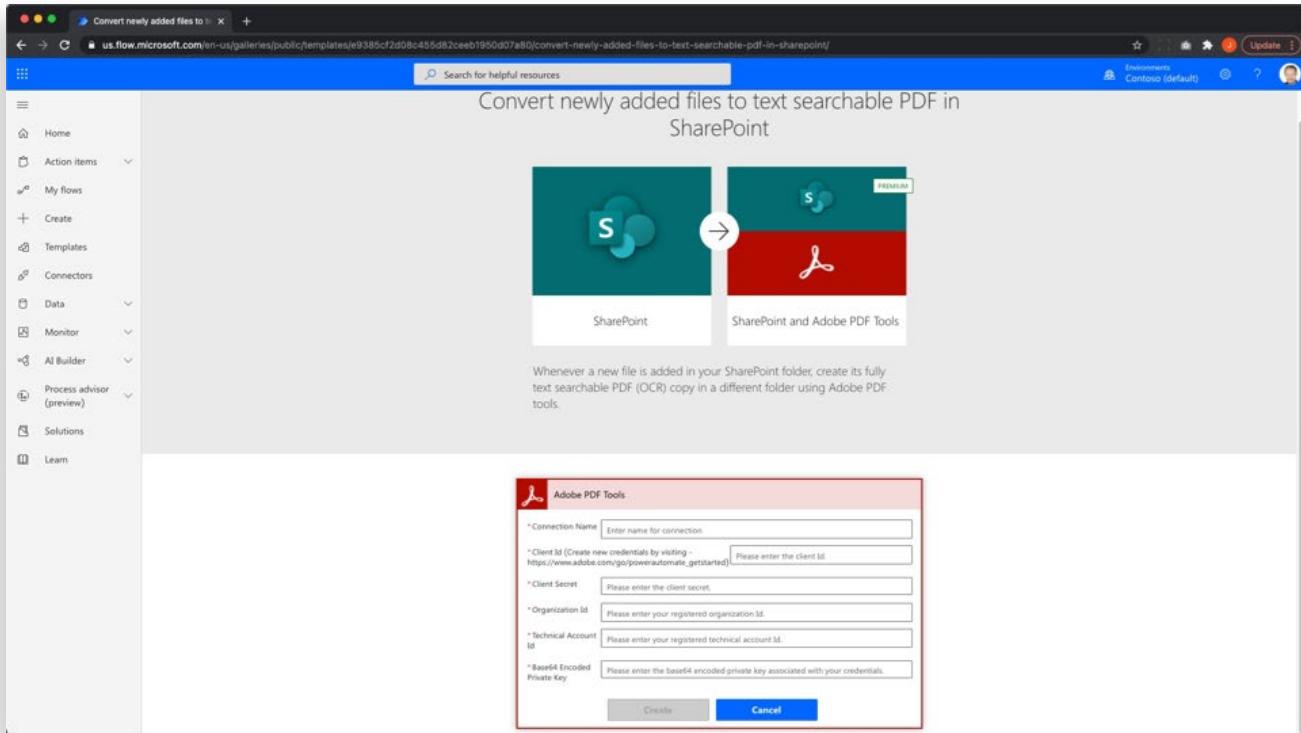
8. Enter Credentials Name and Credentials Description and click Create Credentials.



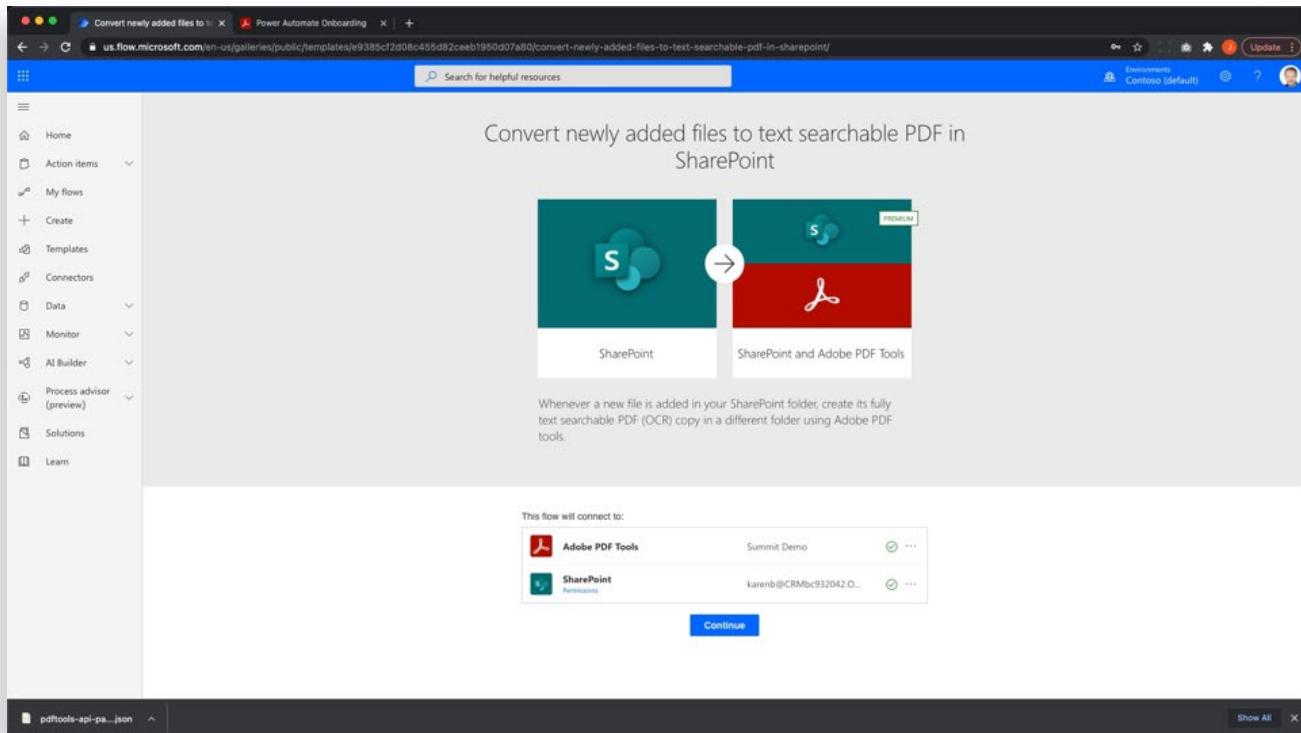
9. Keep the window open with the credentials. You will need them to enter into Microsoft Power Automate.



10. Now, in Microsoft Power Automate fill in the credentials and click Create.

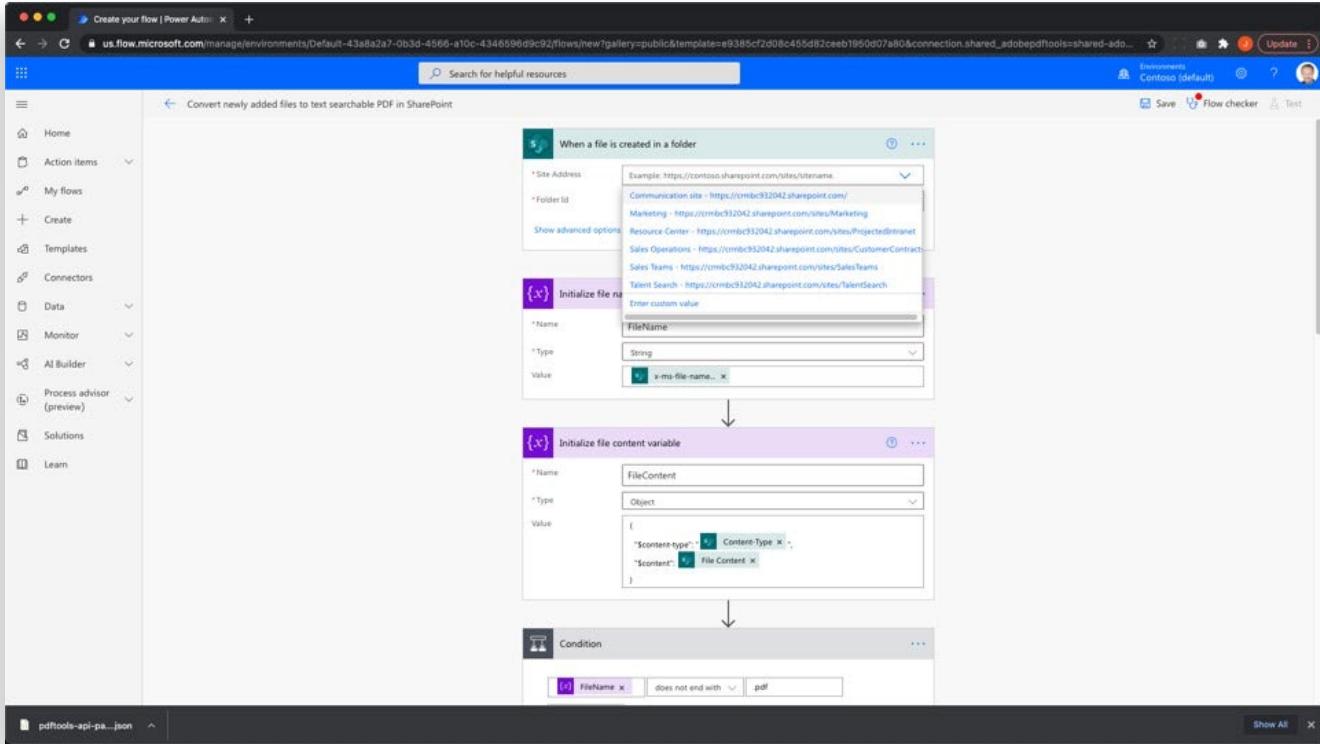


11. Click Continue.

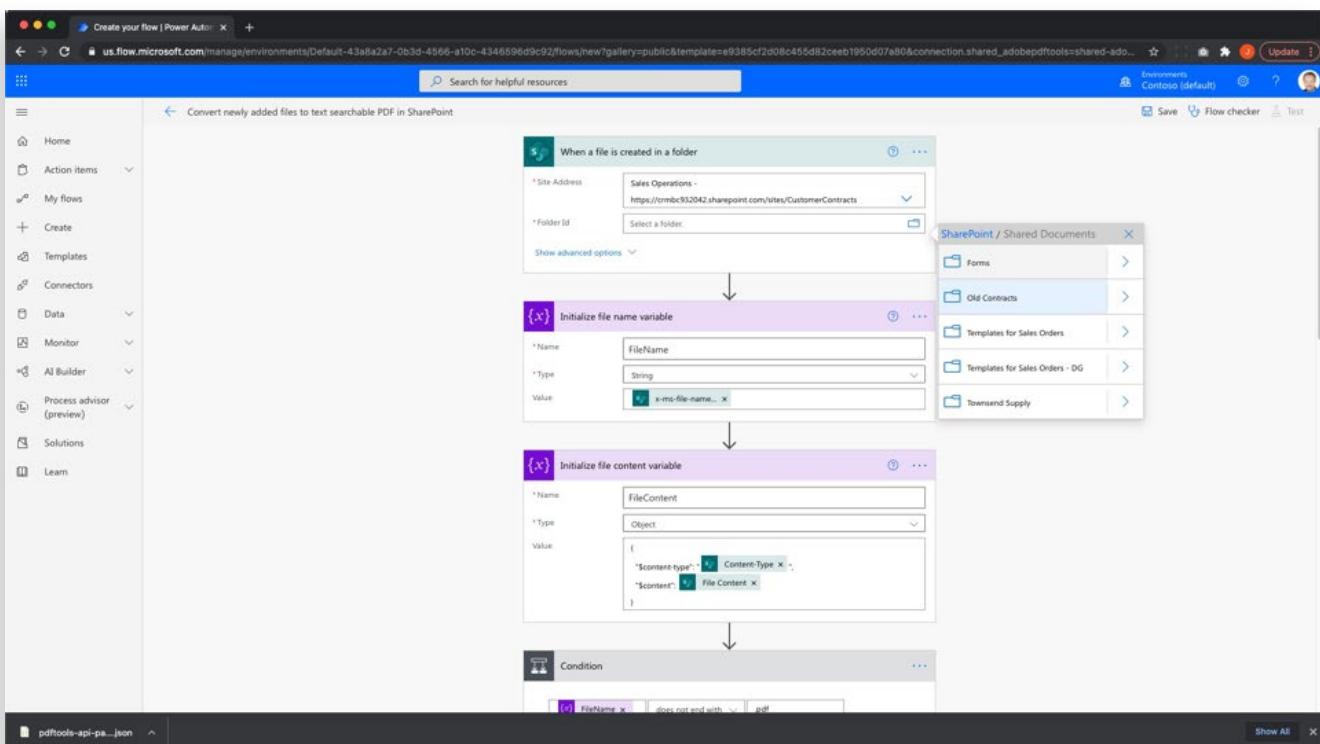


Now you can see a view of the workflow. You will now need to configure it for your environment.

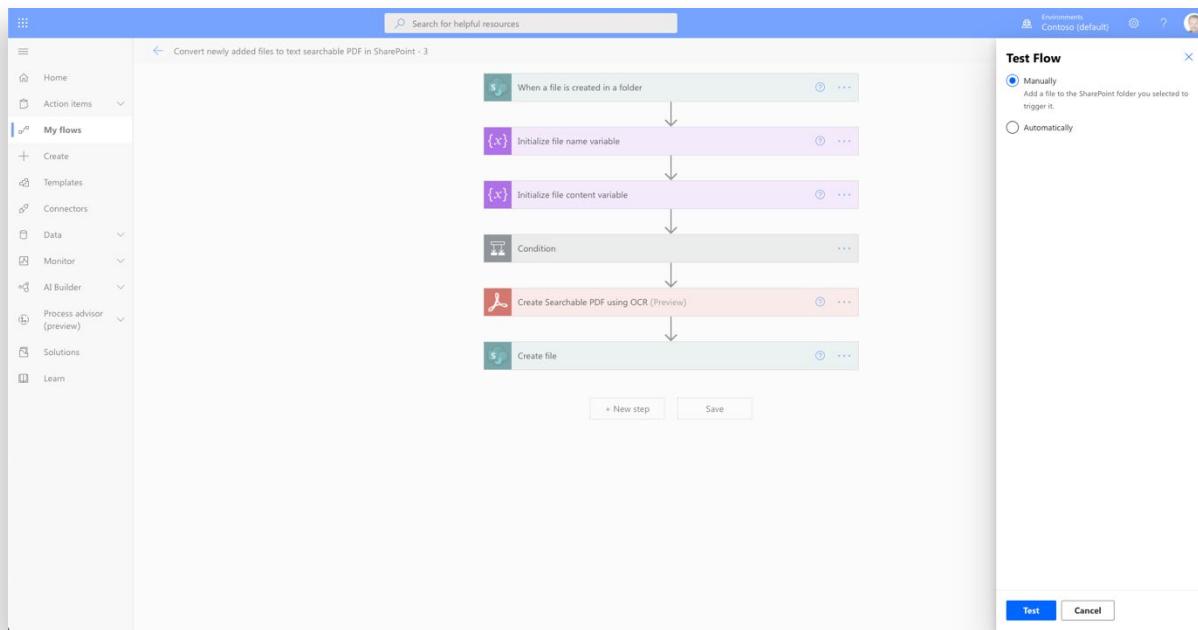
- Under the trigger called "When a file is created in a folder", select Site Address field and choose which SharePoint site you are using.



- Under Folder ID, click on the folder icon to navigate to where the Old Contracts folder is located.

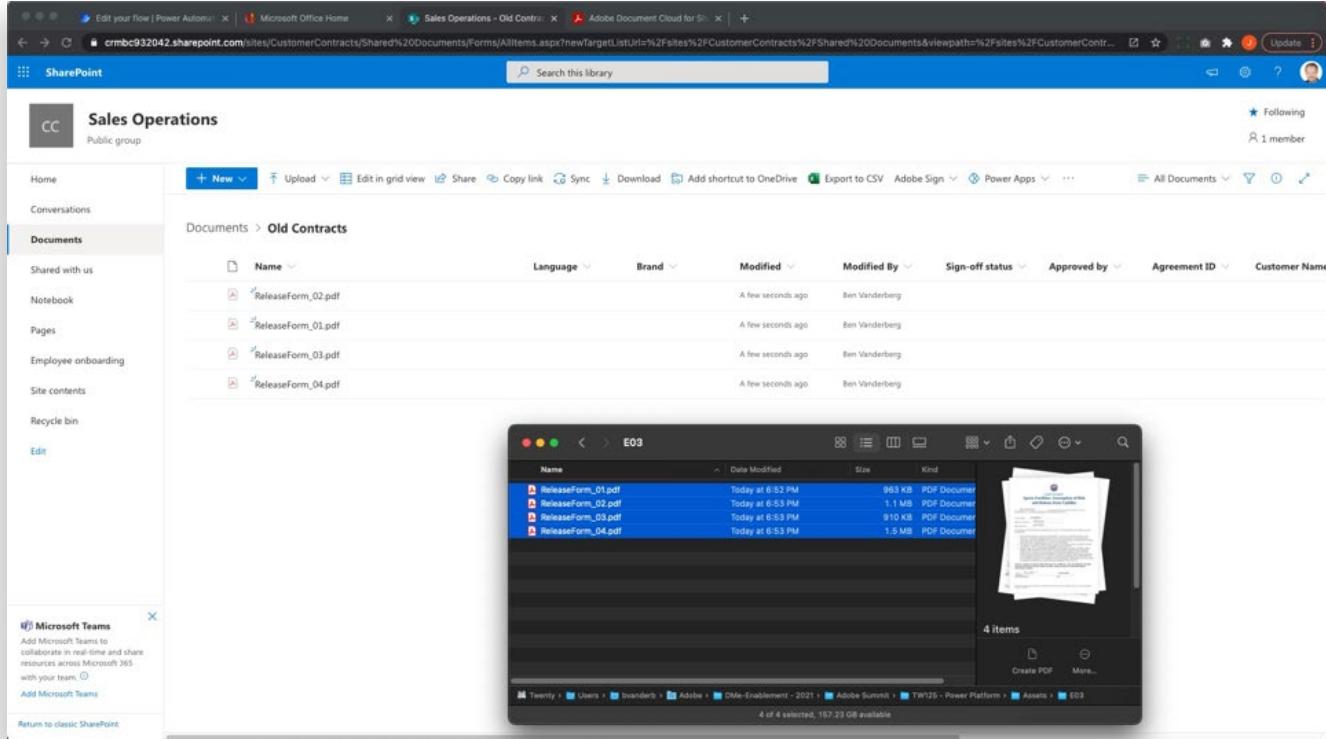


3. Scroll down to the bottom of the flow to edit the Create file action.
4. Just as step 1, change the Site Address to your site address.
5. In the Folder Path, specify the location of the Processed Contracts folder.
6. Click on Save in the top right corner.
7. Click on Test.
8. Select Manually.
9. Click Test.



Try your new flow

1. Go to SharePoint and navigate to the Old Contracts folder.
2. In the exercise files you downloaded, navigate into E03/Old Contracts.
3. Copy the ReleaseFormXX.pdf files into the Old Contracts folder in SharePoint.



Now if you navigate to the Processed Contracts folder, you should then see your PDFs available in that folder after a few moments for the flow to run. If you open the PDFs, you will then see that the text is selectable in the document.

In addition, SharePoint will index that document after a few moments, which will then allow you to search the content of your documents from the search bar in SharePoint.

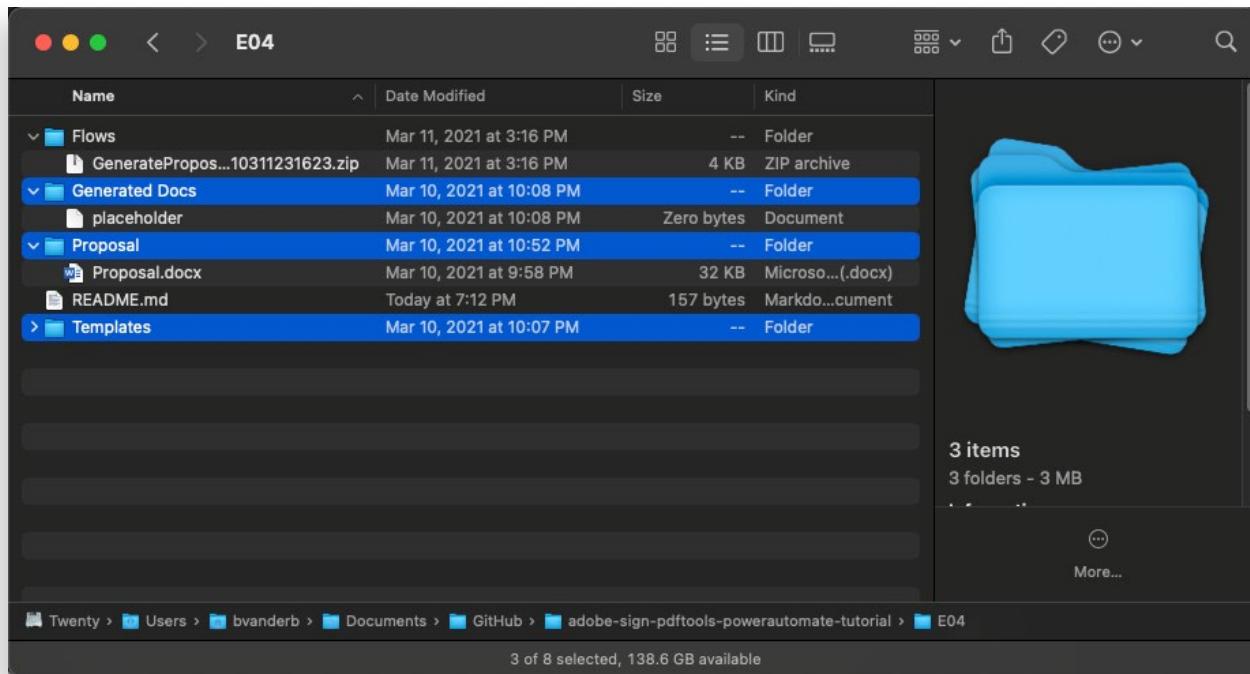
EXERCISE 4: Automated document assembly with Adobe PDF Tools

In this exercise, we will learn how we can merge many documents together based on information provided while selecting and starting a flow from Microsoft SharePoint. In this scenario, the flow will:

- Ask for some information to choose what you want to include in a package for a customer.
- Based on the information provided, it will merge many documents together. These documents include a cover page and optional whitepapers.
- The merged document will get saved to SharePoint.

Import exercise files into SharePoint

1. In the Exercise files, open the E04 folder.
2. In SharePoint, import the Proposal, Templates and Generated Docs folders into SharePoint.



These folders will be used for reference. In particular, we will use the `Proposal.docx` file for our proposal. In the `Templates` folder, there is a `Covers` folder which includes cover page designs for different cities. There is also a `Whitepapers` folder which contains optional additional whitepapers which will get attached to the end if selected.

Import the flow into Microsoft Power Automate

1. Log into Microsoft Power Automate (<https://flow.microsoft.com>).
2. Click on My Flows.

Name	Modified	Type
Send Proposal to Customer	6 min ago	Instant
Convert newly added files to text searchable PDF in S...	3 h ago	Automated
Save Agreements to On-Premise	1 wk ago	Automated
Conditional Approval Workflow	2 wk ago	Instant
Offer Letter Approval - with PDF Tools	2 mo ago	Instant
Offer Letter Approval	2 mo ago	Instant
Change Control Approval	3 mo ago	Instant
Send for Contract Approval	4 mo ago	Instant
Approval Workflow for Document	4 mo ago	Instant
Save an Adobe Sign completed agreement to ShareP...	5 mo ago	Automated

3. Click on Import.

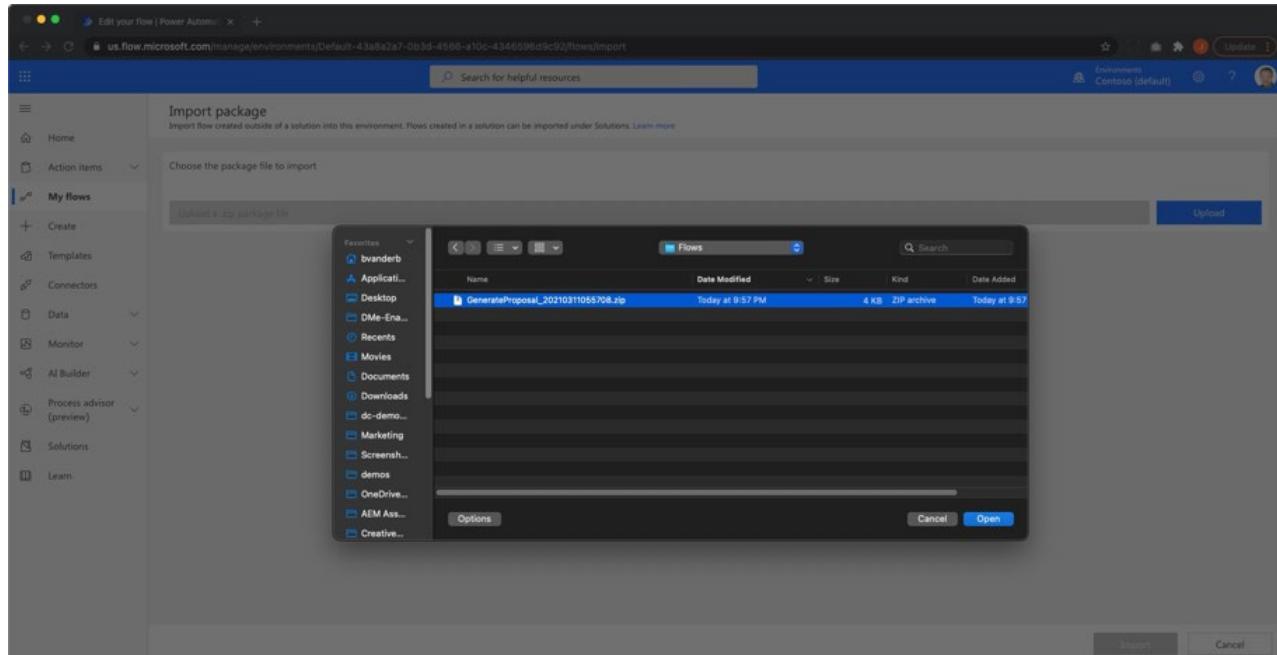
Import package
Import flow created outside of a solution into this environment. Flows created in a solution can be imported under Solutions. [Learn more](#)

Choose the package file to import

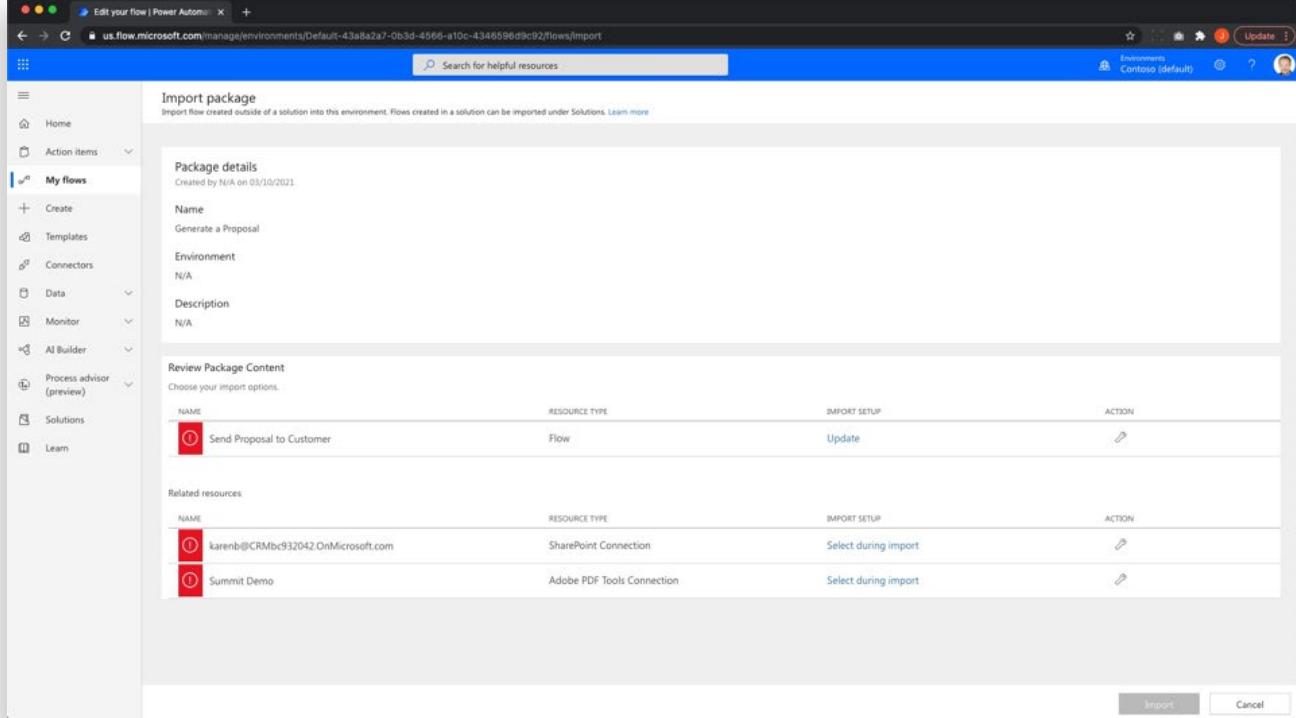
Upload a zip package file Upload

Import Cancel

- Click on Upload and choose the GenerateProposal_20210311231623.zip folder in E04/Flows/.



- Click Import.
- Click on the Wrench icon under Action next to Send Proposal to Customer.



7. Under Setup, select Create as new.

The screenshot shows the Microsoft Power Automate interface for importing a package. On the left, there's a sidebar with options like Home, Action items, My flows (which is selected), Create, Templates, Connectors, Data, Monitor, AI Builder, Process advisor (preview), Solutions, and Learn. The main area is titled 'Import package' and contains sections for 'Package details' (Name: Generate a Proposal, Environment: N/A, Description: N/A), 'Review Package Content' (listing a flow named 'Send Proposal to Customer' with Resource Type 'Flow' and Import Setup 'Update'), and 'Related resources' (listing SharePoint Connection and Adobe PDF Tools Connection). A modal window titled 'Import setup' is open on the right, showing 'Setup' dropdown set to 'Create as new'. Below it, a note says 'The package creator chose this setup. You can make changes to the import here.' and 'This app or flow will be new to the environment when the package is imported.' There's a 'Resource name' field containing 'Send Proposal to Customer 2'. At the bottom of the modal are 'Save' and 'Cancel' buttons.

8. Under Resource Name, set the name of the flow.

9. Click Save.

10. Repeat this for the other Related resources and select your connection.

This screenshot is similar to the previous one but shows the 'Import setup' dialog with a different configuration. The 'Setup' dropdown now has 'Select during import' selected. A note below it states 'The package creator chose this setup. You can make changes to the import here.' and 'The connection or custom API already exists in the environment and must be selected when this package is imported.' A 'Create new' button is available, along with a 'Refresh list' button. The 'Resource name' field still contains 'Send Proposal to Customer 2'. The rest of the interface, including the sidebar and package details, remains the same as in the first screenshot.

11. Once you have made all your connections, click Import button.

Set For a selected file

12. Now that the flow is created, click on Edit.

The screenshot shows the Microsoft Power Automate interface with the following details:

- Flow:** Send Proposal to Customer 2
- Status:** Off
- Created:** Mar 10, 10:05 PM
- Modified:** Mar 10, 10:05 PM
- Type:** Instant
- Plan:** Per-user plan

Connections:

- Adobe PDF Tools: Summit Demo
- SharePoint Permissions: karenb@CRMtc932042.OnM...

Owners: Ben Vanderberg

Run only users: Your flow hasn't been shared with anyone.

13. Select the trigger For a selected file.

14. In the Site Address, specify your SharePoint site.

15. In Library, specify your Library.

The screenshot shows the Microsoft Power Automate interface with the following configuration for the 'For a selected file' trigger:

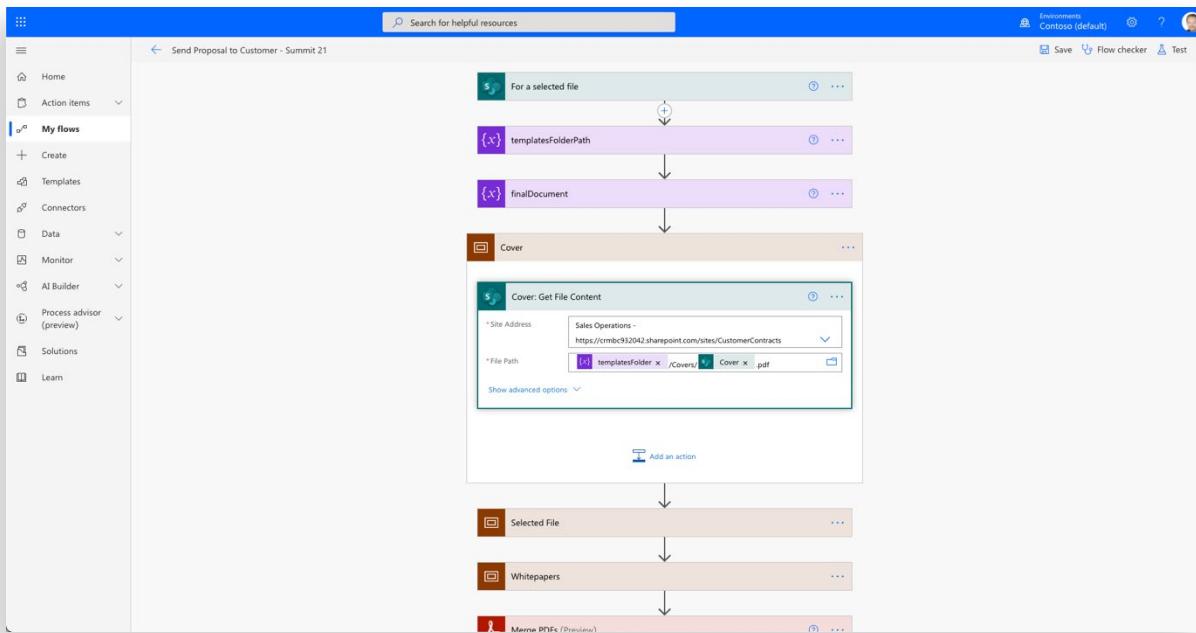
- Site Address:** Sales Operations - <https://ombc932042.sharepoint.com/sites/CustomerContracts>
- Library Name:** Documents
- Drop-down list of options:**
 - London
 - New York
 - San Francisco
 - Seattle
 - Tokyo
 - Enter another option
- Inputs:**
 - Cover: Please enter your input
 - Add Whitepaper 1: Please select yes or no
 - Add Whitepaper 2: Please select yes or no
 - Customer Name: Please enter your input
 - Password: Please enter your input

Set templateFolderPath

16. Click on the templateFolderPath variable.
17. Set the path to where the Templates folder is located relative inside the SharePoint site that you imported before.

Set Cover Get File Content

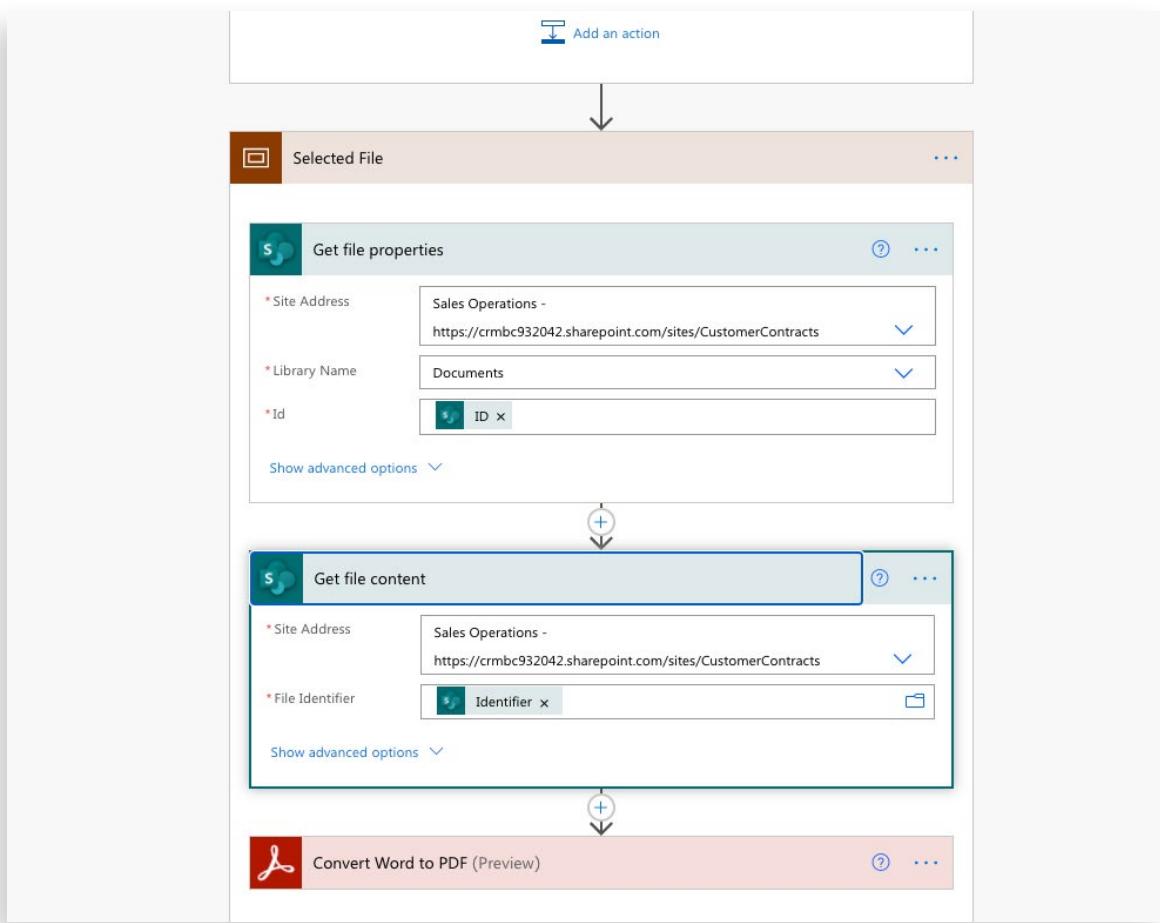
18. Click on the Cover action, which expands the Scope. Underneath that, expand Cover: Get File Content.



19. Under Site Address and set it to your SharePoint site.

Set Selected File

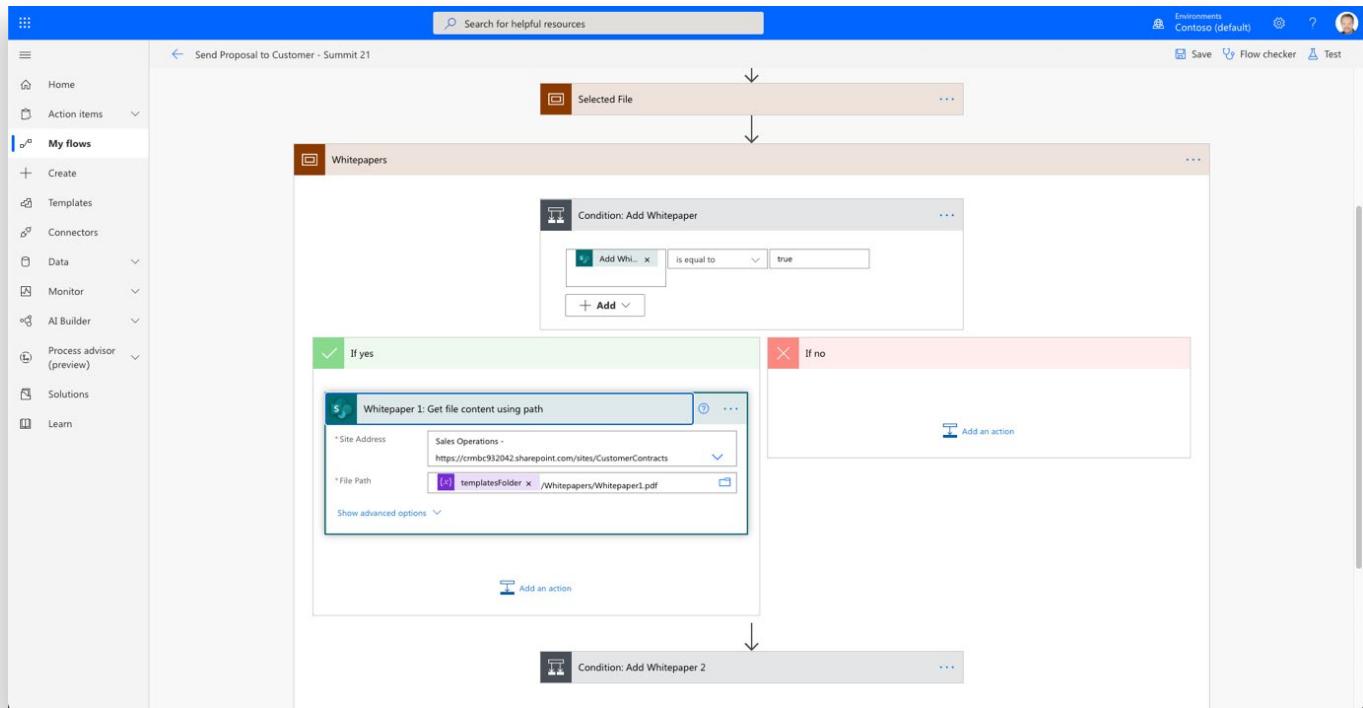
20. Expand the Selected File scope action.



21. Under Get file properties, change the Site Address and Library Name to your SharePoint site and Library respectively.
22. Under Get file content, change the Site Address to your SharePoint site.

Set Whitepapers

23. Click on Whitepapers scope action.
24. Expand the *Condition: Add Whitepaper* condition.



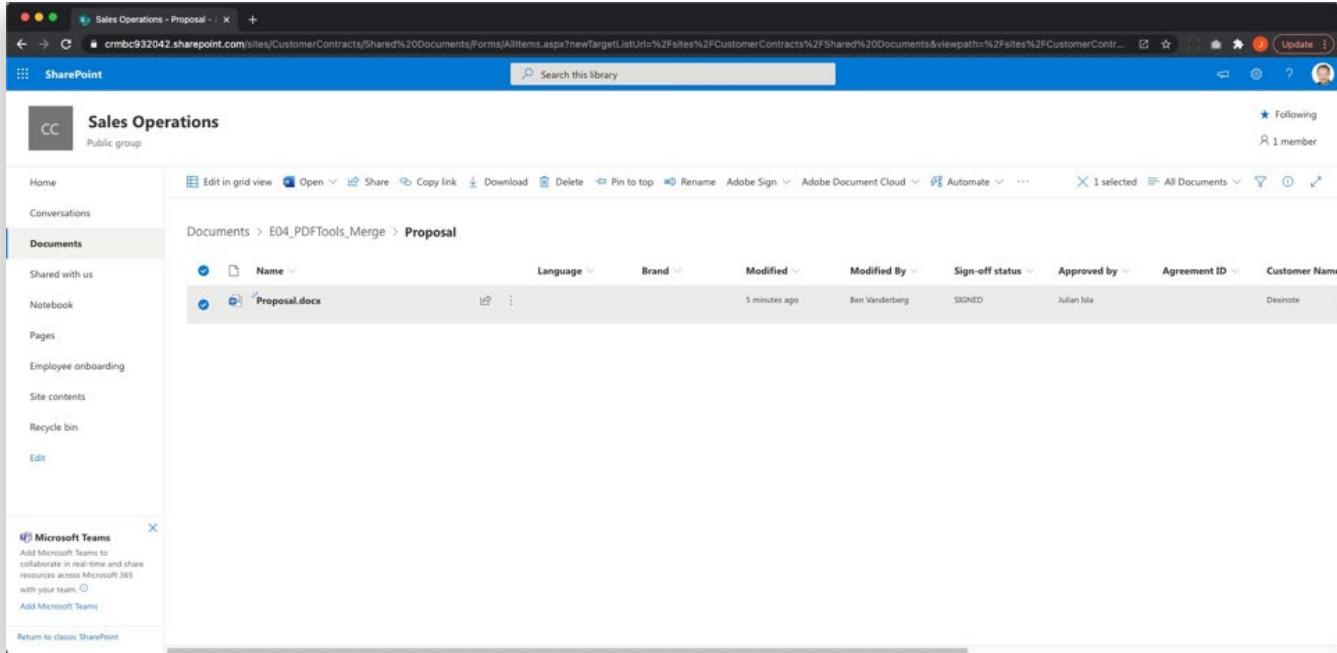
25. Expand the Whitepaper 1: Get file content using path.
26. Change the Site Address to your specified SharePoint site.
27. Repeat the same steps for the Condition: Add Whitepaper 2 condition.

Set Create File

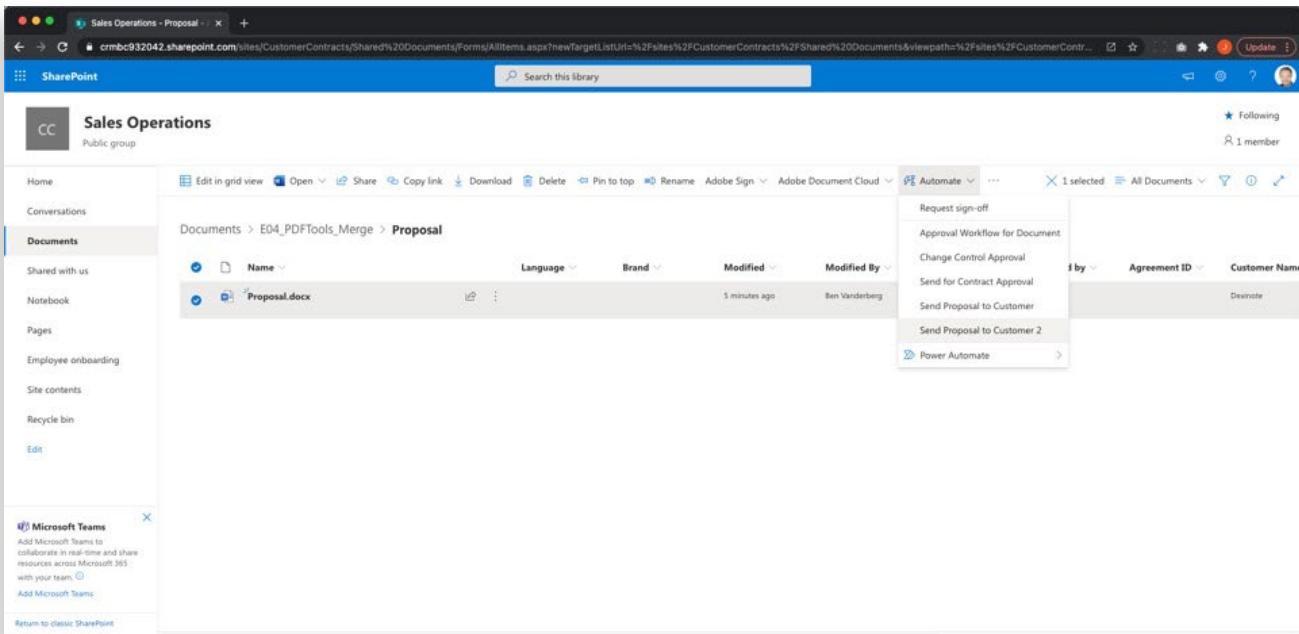
28. As the last step of the action, expand Create File.
29. Set Site Address and Folder Path to the SharePoint site and path where the Generated Docs folder is located.
30. Click Save.

Test your flow

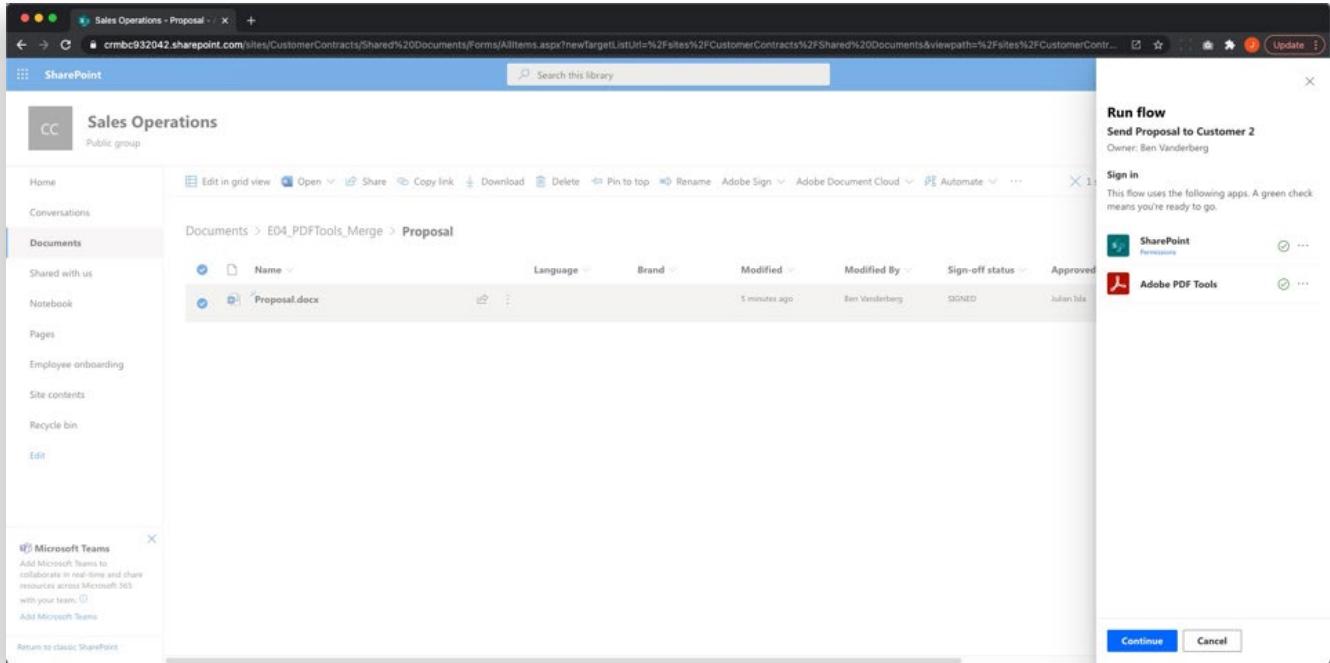
1. In SharePoint, navigate to the Proposal folder.
2. Select the Proposal.docx folder.



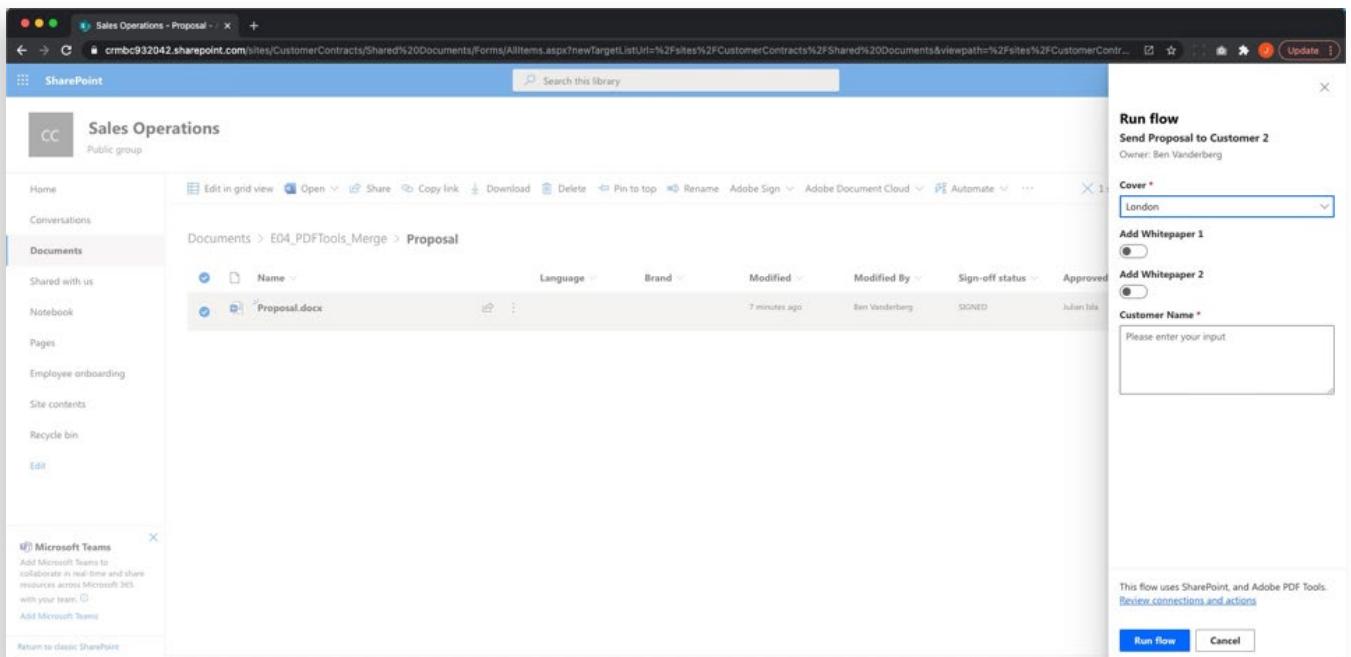
3. Use the Automate menu and select your flow.



- Click on Continue to begin the flow.



- Choose your Cover, and which whitepapers you want to append.
- Click on Run flow.
- Navigate to the Generate Docs folder.



You should now see your generated PDF file.

The screenshot shows a SharePoint document library titled "Sales Operations". The left navigation pane is open, showing sections like Home, Conversations, Documents, Shared with us, Notebook, Pages, Employee onboarding, Site contents, Recycle bin, and Edit. A Microsoft Teams integration is visible on the left. The main area displays a list of documents under "Generated Docs", with columns for Name, Language, Brand, Modified, Modified By, Sign-off status, Approved by, Agreement ID, and Customer Name. The list includes various PDF files such as "Customer_Tokyo.pdf", "John Echostone_Seattle.pdf", and "Sarah Rose_Seattle.pdf".

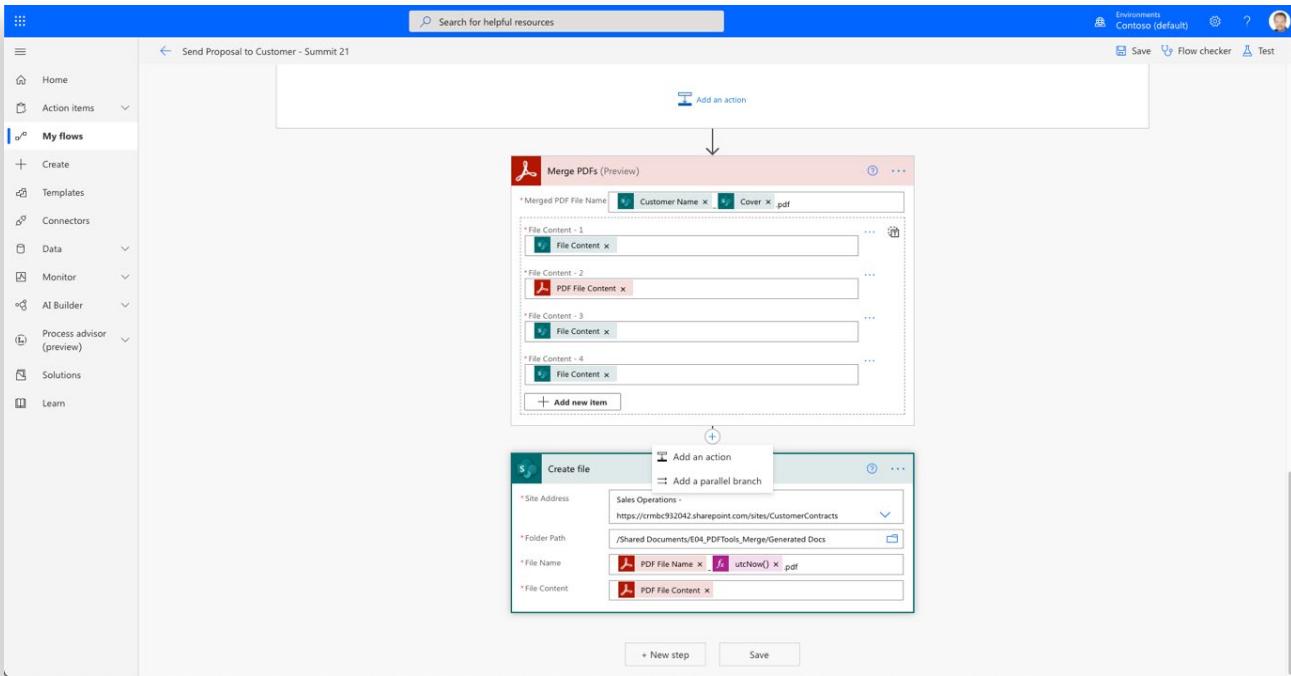
Adding Protect and other actions to flow

Now that we have successfully created a flow, we are going to edit our flow to encrypt the PDF document with a password. This will also walk through how you can utilize other actions.

1. Navigate back into your flow.
2. Navigate to the end of the flow.

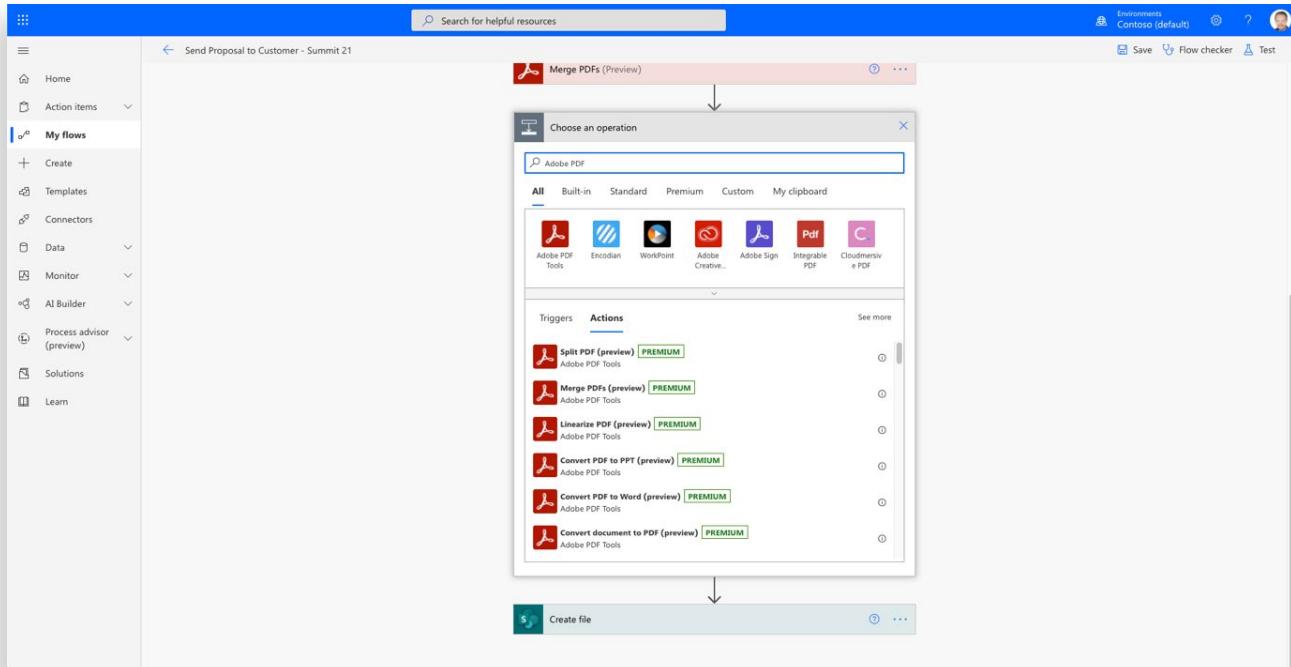
The screenshot shows the Microsoft Flow editor for a flow named "Send Proposal to Customer - Summit 21". The left sidebar shows categories like Home, Action items, My flows, Create, Templates, Connectors, Data, Monitor, AI Builder, Process advisor (preview), Solutions, and Learn. The main canvas shows two actions: "Merge PDFs (Preview)" and "Create file". The "Merge PDFs" action has "File Content" inputs connected to previous steps. The "Create file" action has "Site Address" set to "Sales Operations" and "Folder Path" set to "/Shared Documents/ED4_PDFTools_Merge/Generated Docs". The "File Name" field contains "utcNow() .pdf" and "File Content" is set to "PDF File Content".

3. Click on the plus icon between Merge PDFs and Create file.



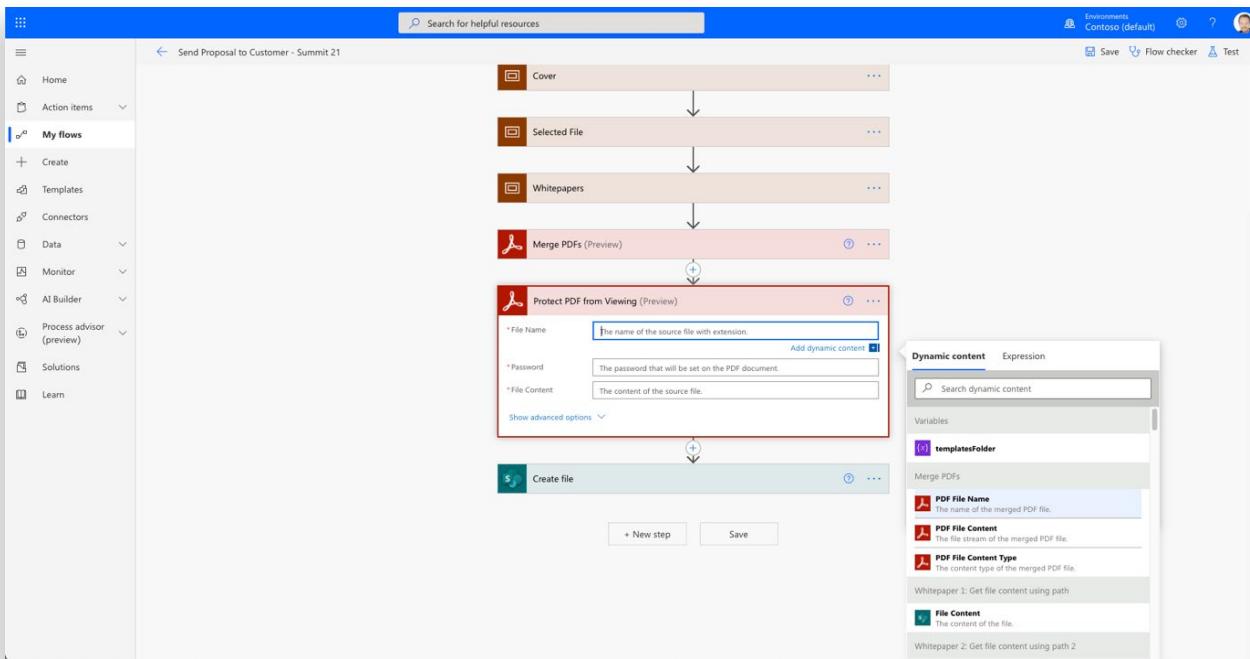
4. Select Add an action.

5. Search for Adobe PDF Tools.

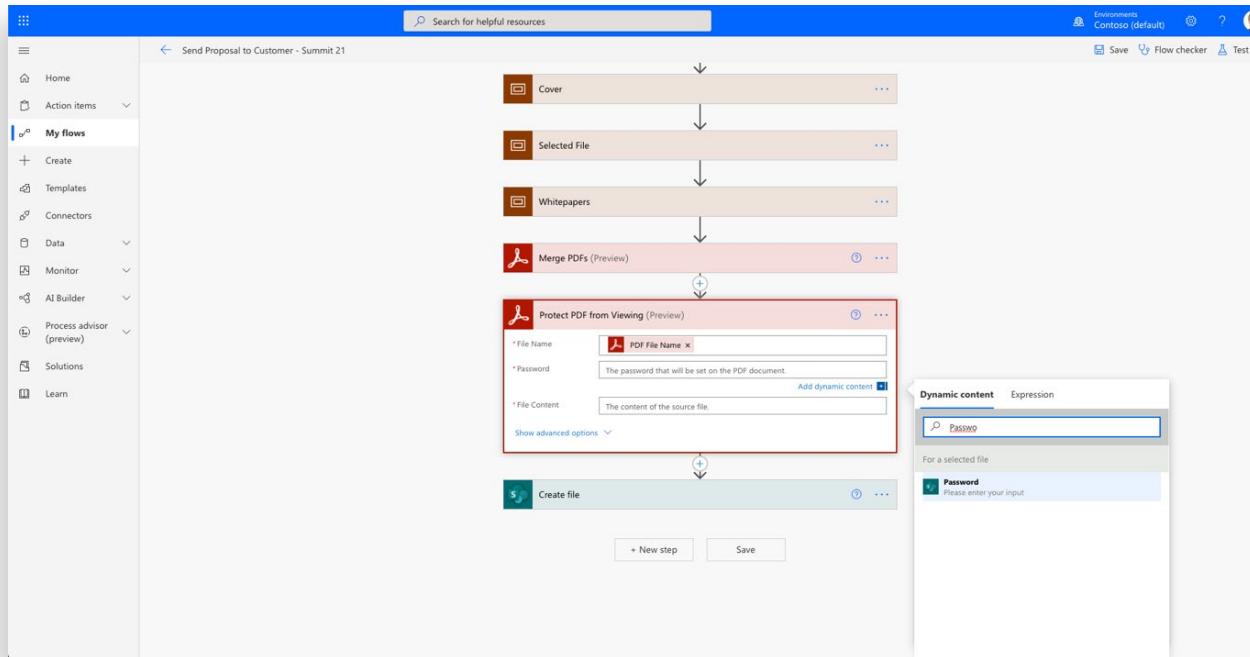


6. Select "Protect PDF from Viewing".

- In the File Name field, use Dynamic Content to set it to PDF File Name from Merge PDF.



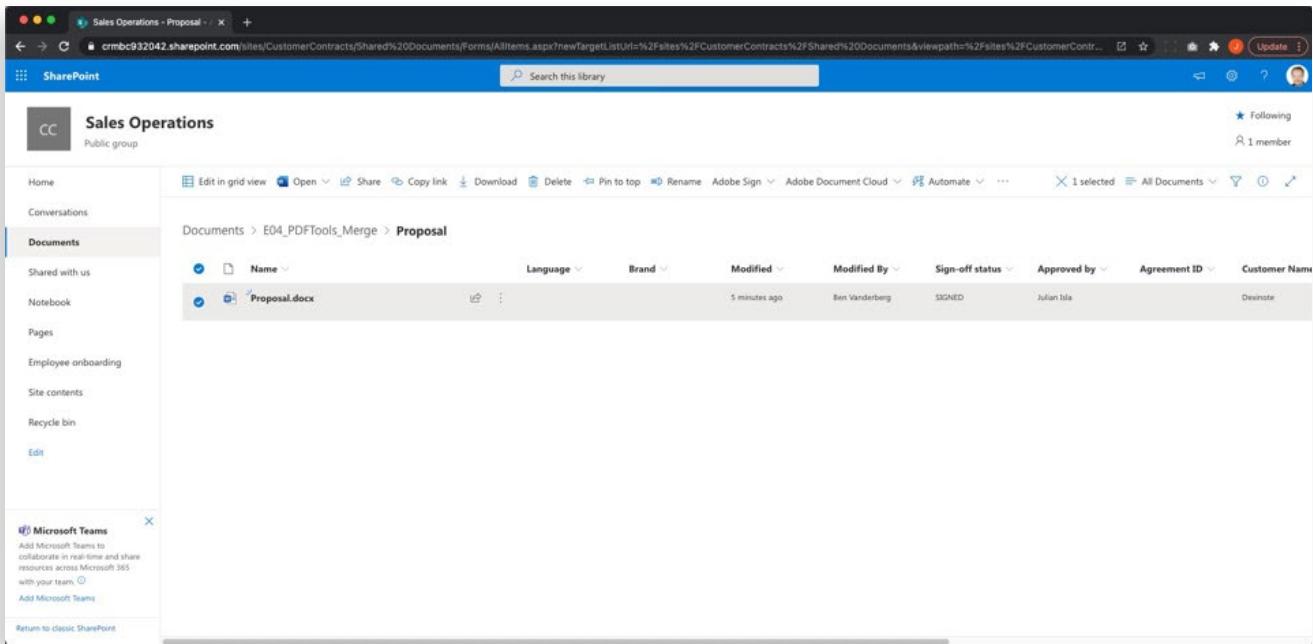
- In the trigger, there is a Password field that is part of the initiation form. We can use that here. Using Dynamic content, search for Password field and place it in the Password field.



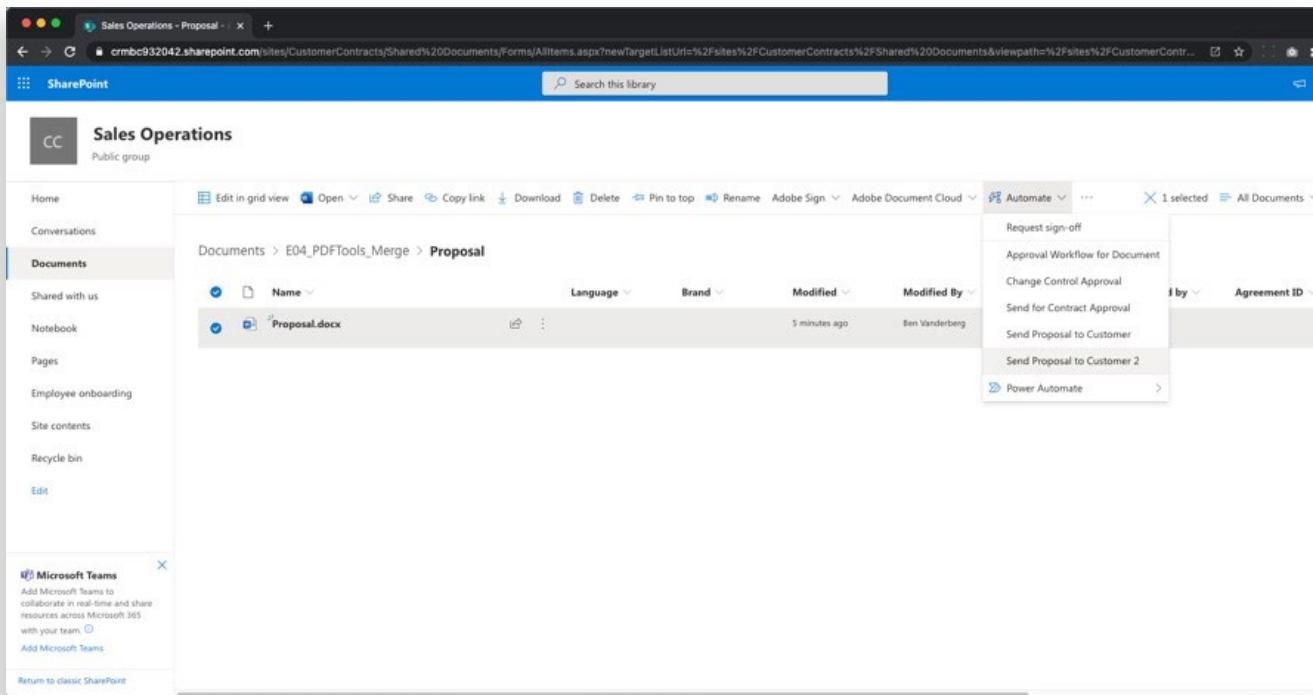
- In the File Content field, use Dynamic content to set it to PDF File Content from Merge PDFs.
- Right now, the Create file needs to be changed to get the file content from Protect PDF, rather than Merge PDFs.
- Expand Create file.
- Clear the File Content field.
- Use Dynamic Content and place PDF File Content from Protect PDF from Viewing.

Test your flow

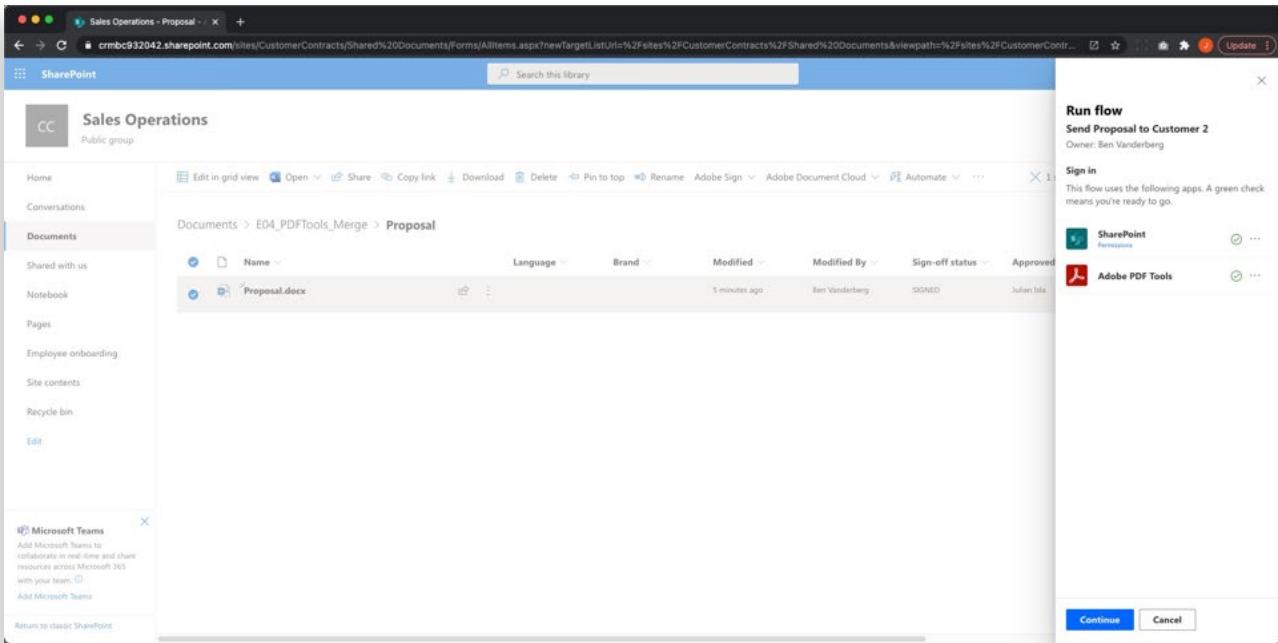
1. In SharePoint, navigate to the Proposal folder.



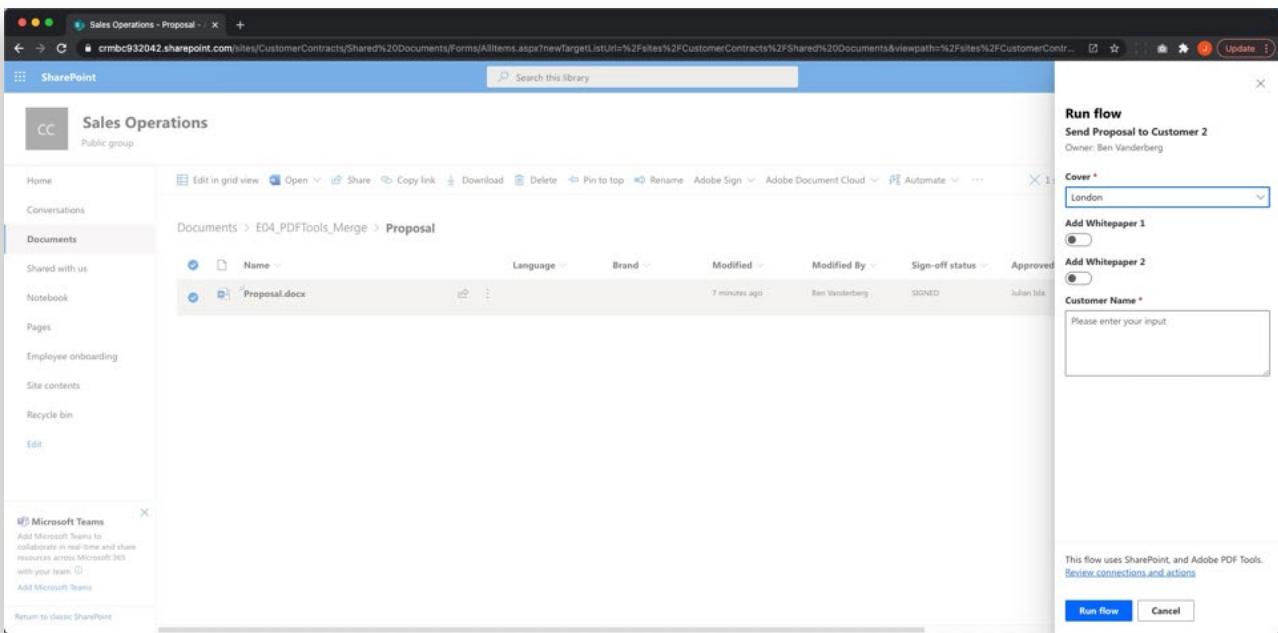
2. Select the Proposal.docx folder.
3. Use the Automate menu and select your flow.



4. Click on Continue to begin the flow.



5. Choose your Cover, and which whitepapers you want to append.



6. Set the Password field to the Password you would like to set.

7. Click on Run flow.

8. Navigate to the Generate Docs folder.

You should now see your generated PDF file.

The screenshot shows a SharePoint library titled "Generated" under the "Sales Operations" site. The library contains a grid of PDF documents. The columns include Name, Language, Brand, Modified, Modified By, Sign-off status, Approved by, Agreement ID, and Customer Name. The PDF names follow a pattern like "Customer_Tokyo.pdf_2021-03-11T05_32_04.4810...". A Microsoft Teams sidebar is open on the left, and a search bar is at the top.

Name	Language	Brand	Modified	Modified By	Sign-off status	Approved by	Agreement ID	Customer Name
Customer_Tokyo.pdf_2021-03-11T05_32_04.4810...			44 minutes ago	Ben Vanderberg				
Customer_Tokyo.pdf_2021-03-11T05_34_05.02270072.pdf			42 minutes ago	Ben Vanderberg				
Customer_Tokyo.pdf_2021-03-11T05_46_08.59666802.pdf			30 minutes ago	Ben Vanderberg				
Customer_Tokyo.pdf_2021-03-11T05_50_29.15985272.pdf			26 minutes ago	Ben Vanderberg				
John Echostone_Seattle.pdf			About an hour ago	Ben Vanderberg				
John Echostone_Seattle.pdf_2021-03-11T05_20_40.26200622...			About an hour ago	Ben Vanderberg				
John Echostone_Seattle.pdf_2021-03-11T05_22_24.82885832...			About an hour ago	Ben Vanderberg				
John Smith_London.pdf_2021-03-11T05_25_00.38235802.pdf			About an hour ago	Ben Vanderberg				
John Smith_London.pdf_2021-03-11T05_27_03.25028312.pdf			49 minutes ago	Ben Vanderberg				
Mary Burstone_Seattle.pdf_2021-03-11T05_28_51.09588072...			47 minutes ago	Ben Vanderberg				
Sarah Rose_Seattle.pdf_2021-03-11T06_15_33...			About a minute ago	Ben Vanderberg				

If you open the PDF file, it will prompt you to enter in your PDF password.