IOEM Validation Effort

Objective:

To automate the process of validating the information associated with the ~220 ticket queues that are specified in the Incident Ownership and Escalation Matrix, and the contact points for those groups. Contact points in this effort included Prime Shift contact, Secondary contact, After Hours contact, and the 1st-4th escalation contacts.

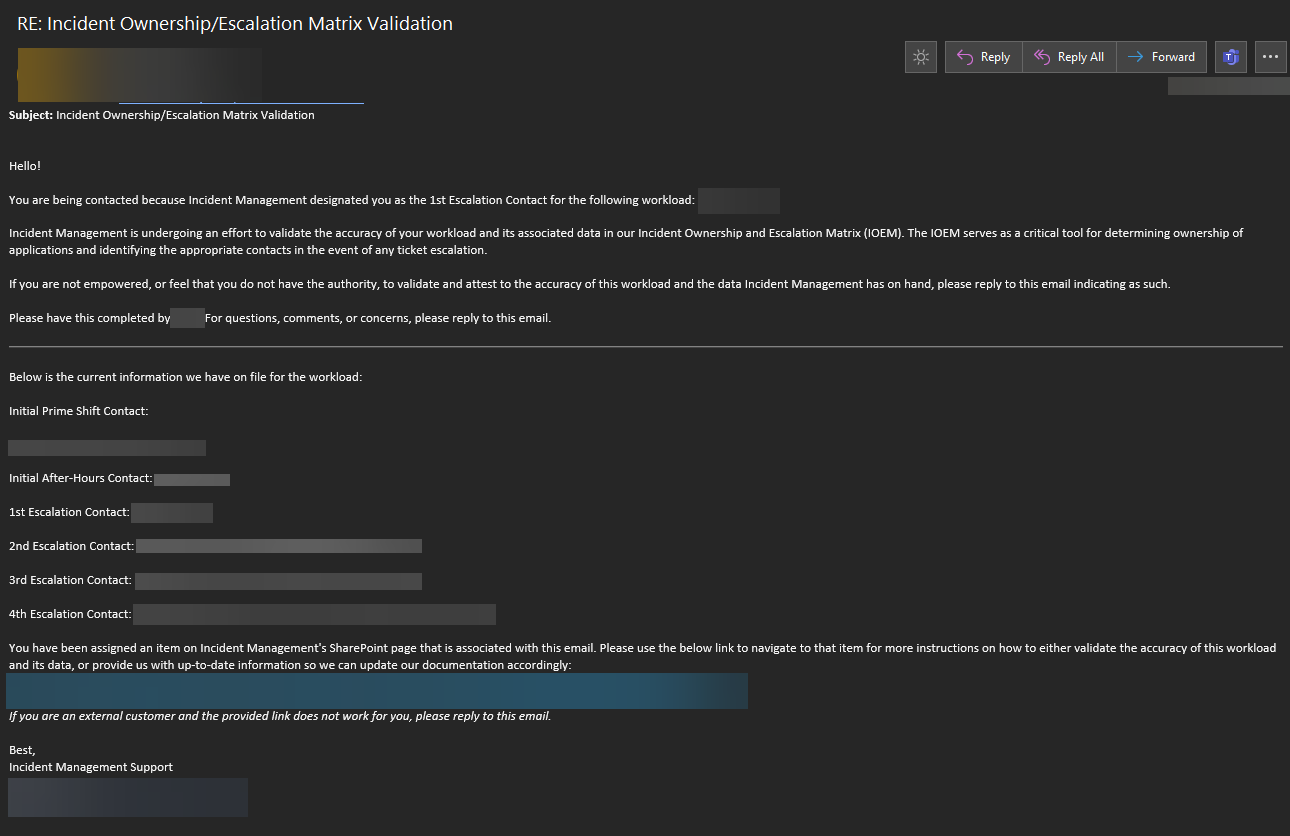
Tools of this trade:

I decided to use four products/services for this effort: Power Automate, Outlook, Forms, and SharePoint (all Microsoft products).

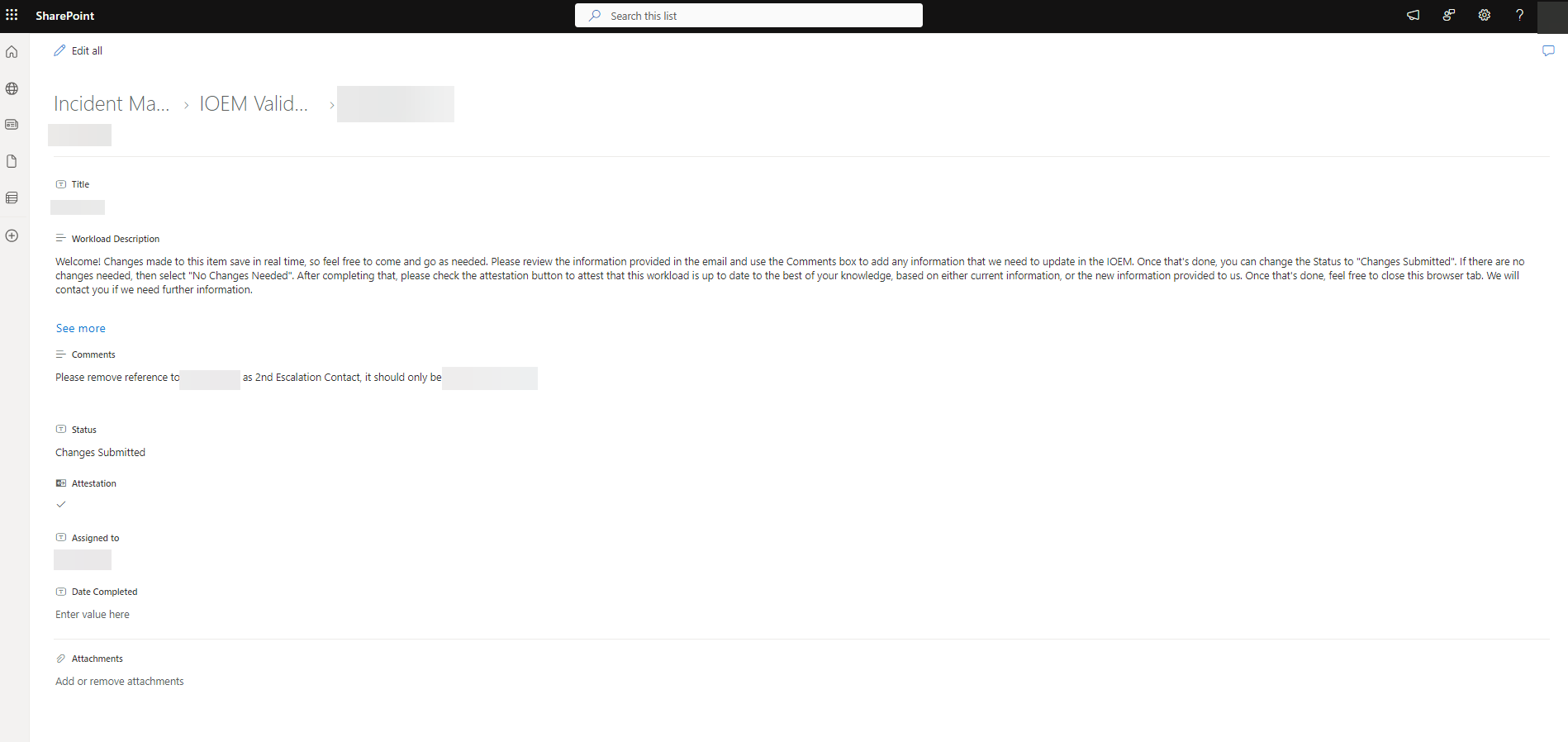
* SharePoint houses a list that contains one item for each aforementioned workload (~220 total) and provides a platform to be able to add/configure “views” in SharePoint (SP). Views are just ways to see data, a lot like filtering and sorting. Technically, the list was created by Microsoft Lists (another application), but I omitted that from this document for the purpose of simplicity, since SP shows and provides everything you need.
* Power Automate is the tool that carries out the actions of manipulating the data from SP, Outlook, Forms, and the list of O365 users. This manipulation allows the flow (series of actions with an end result) to send out emails to one individual for each workload, about 180 emails or so (multiple buckets are owned by the same management chain). In this effort, PA performs two flows
* Outlook is merely a vessel used to ship the emails that PA generates. It is controlled and manipulated by PA in our case.
* Forms is the tool used to aid the effort with all the workloads that were omitted from the initial rollout of emails. It was designed for more control throughout the process. It contains one question, which asks for a bucket name to send an email off for, and that kickstarts the second flow in this effort.

User Flow:

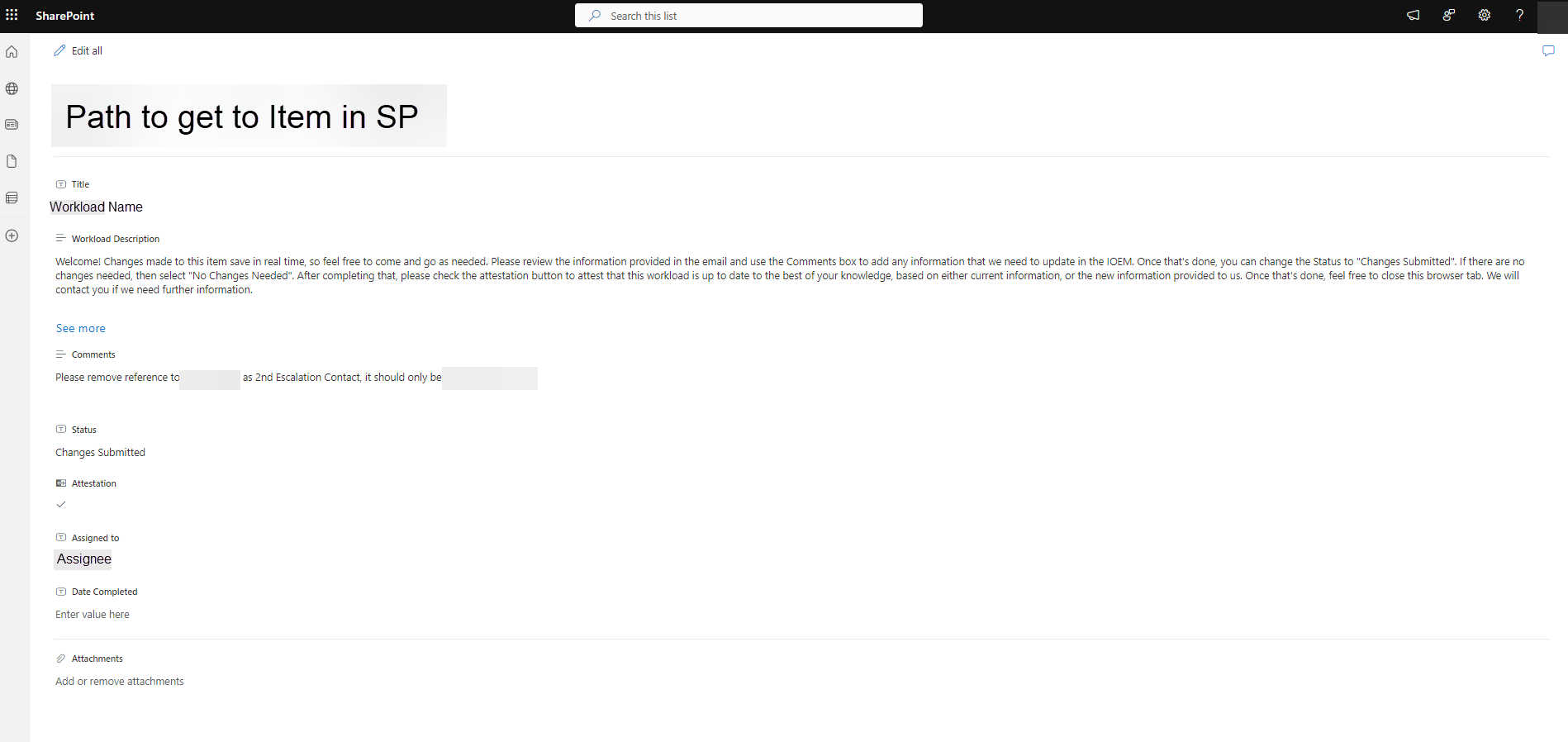
The user will get an email notifying them of the effort, instructions/expectations and a link to their assigned item:



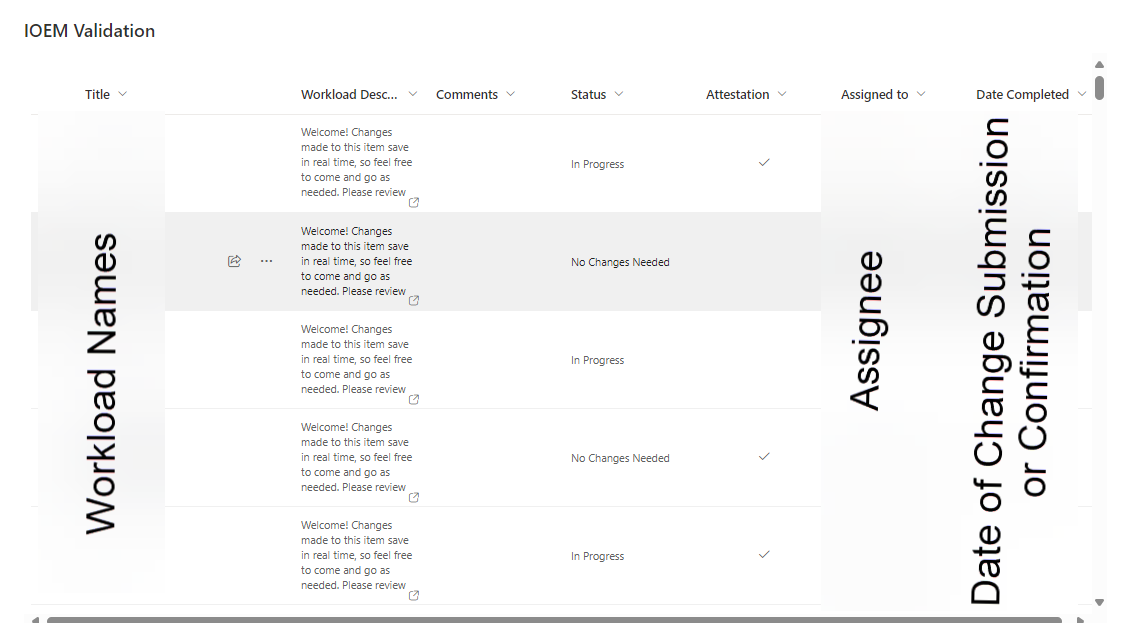
The users will click on the link and will see the following editable fields:



The Workload Description field provides instructions on how to proceed for the user, and what fields to fill out, and what is expected from each field based off the circumstances surrounding that workload. In a perfect world, every item will look like this:



The below image is how IMS can view the status of each item in this list to ensure things are moving forward to 100% validation:

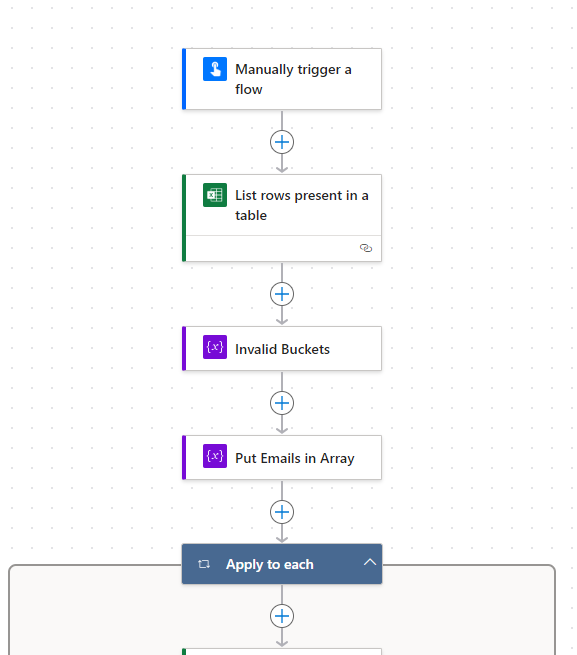


It displays all the information needed for IMS to keep track of the effort and allows us to reach out to any assignees who are falling behind on their task.

Process Flow:

The process flow for this will start with PA and all the data it needs to perform the first (and primary) flow. A list is created beforehand to house all the data. I titled mine IOEM Validation and housed it in the Incident Management SP page.

All PA flows must have some sort of trigger to execute the actions that follow, so I used the ‘Manually trigger a flow’ trigger, which allows me to kick off the flow at the click of a button. The next action, ‘List rows present in a table’, grabs all the rows (and the data in those rows) from a copy of a recently updated IOEM, and stores it, allowing for retrieval of that data by any other action(s). I then initialize two variables, ‘Invalid Buckets’ and ‘Put Emails in Array’, seen below:



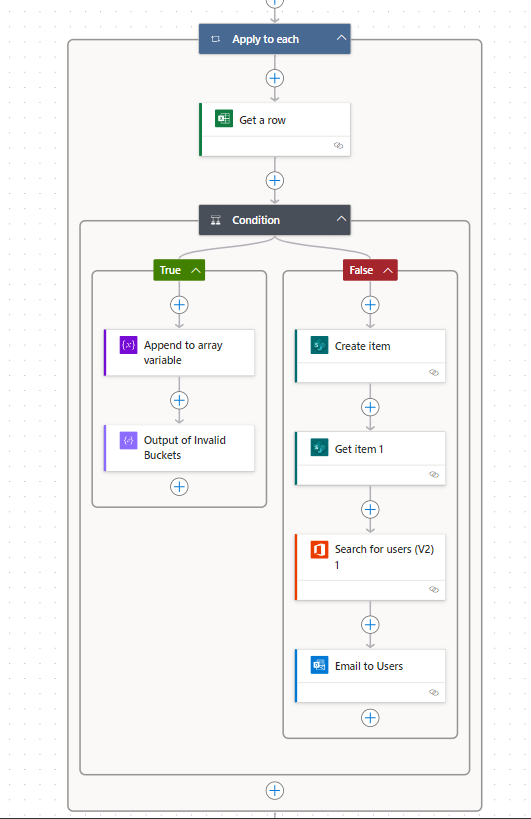
‘Invalid Buckets’ is an array variable that will hold the names of buckets that pass either one of the conditions coded in for the Conditional action. This variable did not have any initial value added to it during its creation. ‘Put Emails in Array’ was not set with an initial value and was designed to take all known emails with each user and put them in an array for filtering later in the flow.

An ‘Apply to each’ action is automatically added to the rest of the flow. This is due to the next action (‘Get a row’) referencing all the rows in the ‘List rows present in a table’ action. This gets every row in the spreadsheet and has the data in those rows ready to go. While this action and the ‘List rows present in a table’ action seem to do the same thing, they do not. The next action, a Conditional, is an “If” statement, with the following conditions:

If the [value of the] first escalation contact is equal to “Blank”, then the flow will add the bucket name to the ‘Invalid Buckets’ variable.

If the [value of the] first escalation contact exceeds 20 characters in length, then the flow will add the bucket name to the ‘Invalid Buckets’ variable.

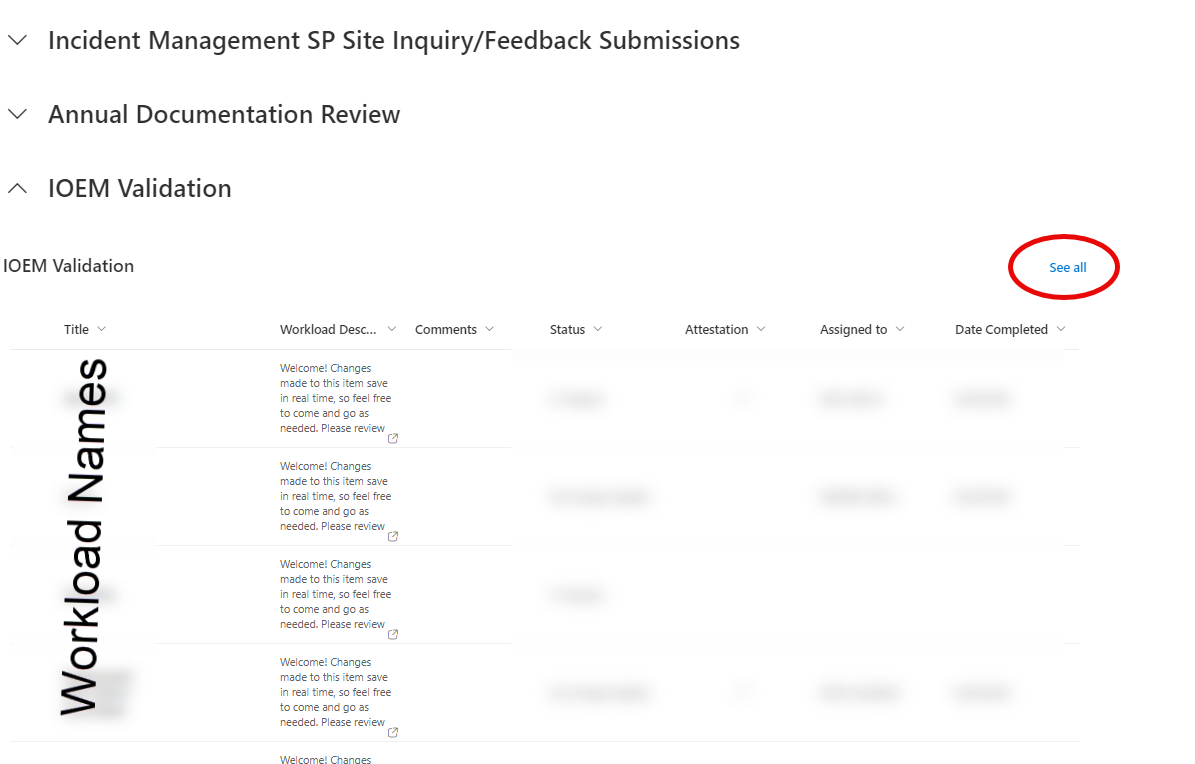
The two actions under ‘True’ are what adds the bucket names to the ‘Invalid Buckets’ variable and displays them for me to make sure those look right. If the row passes as false in the conditional, then it will make a SP item, assign it, and send an email to the assignee notifying them of this validation effort, provides them with instructions, and a link to access the item on SP.



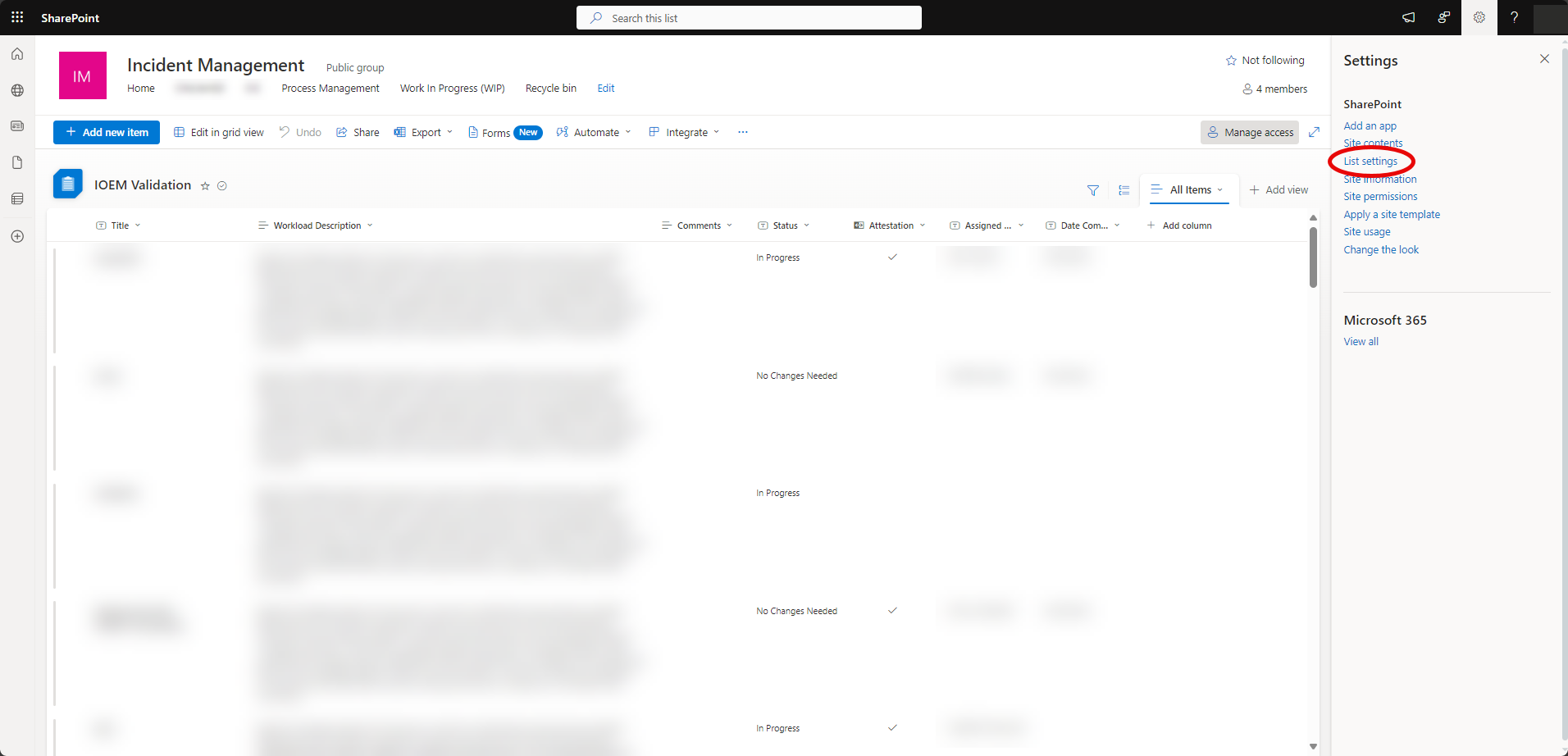
It is also worth noting that during the rollout, it was discovered that any assignee will need to be added to a SP group with Edit or Contribute access to this List within the IMS SP site.

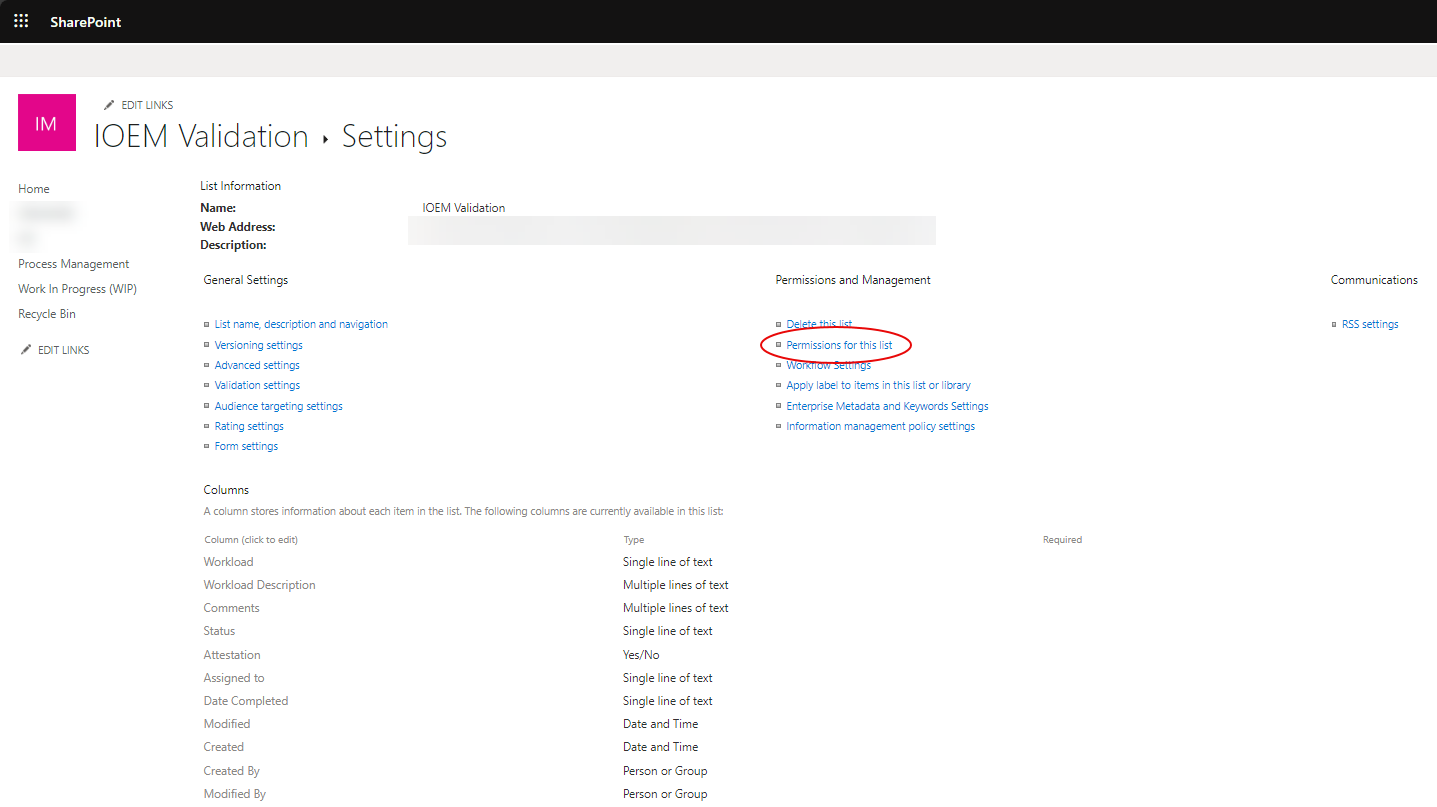
SharePoint Group:

Navigate to the Project Management page and click See All:

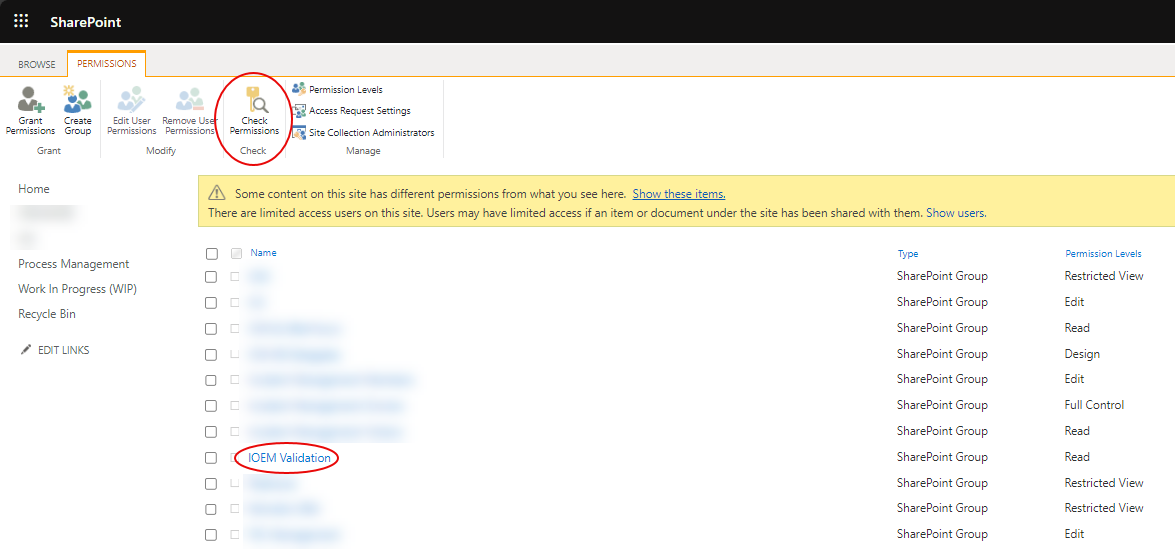


Then click the cog, List settings, then Permissions for this list:

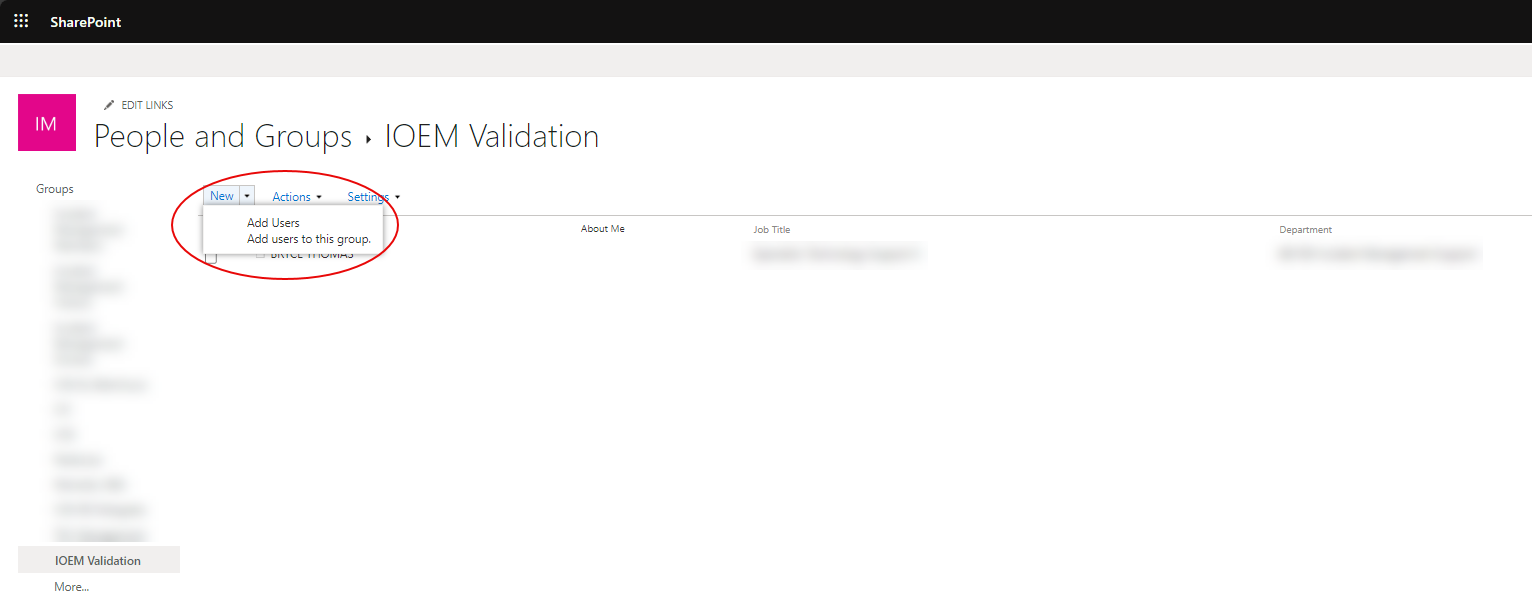




In the image below, click on the IOEM Validation Group. You can also check the permissions of a specific user to aid in troubleshooting, if applicable, by selecting Check Permissions.

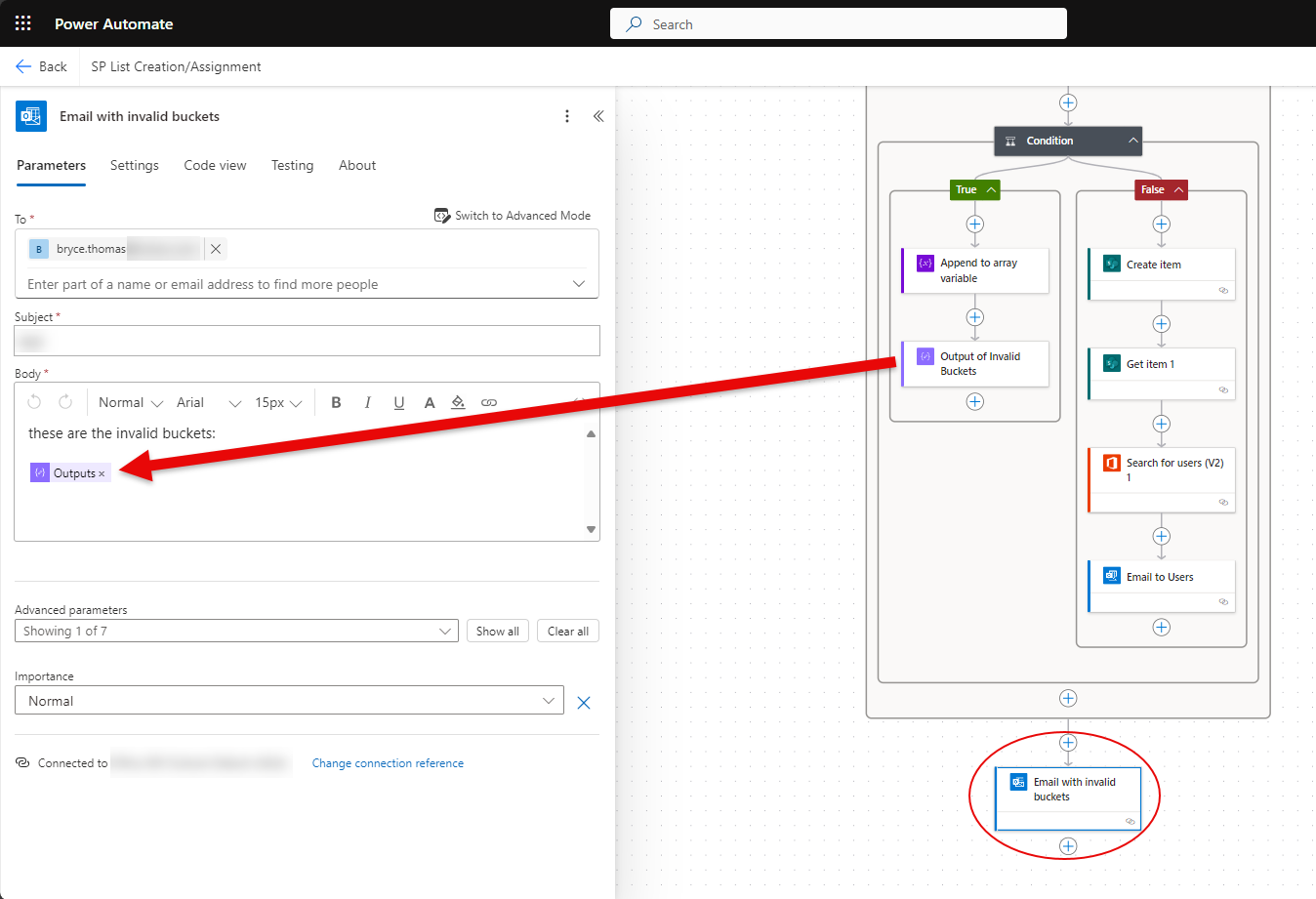


Click on New > Add Users:



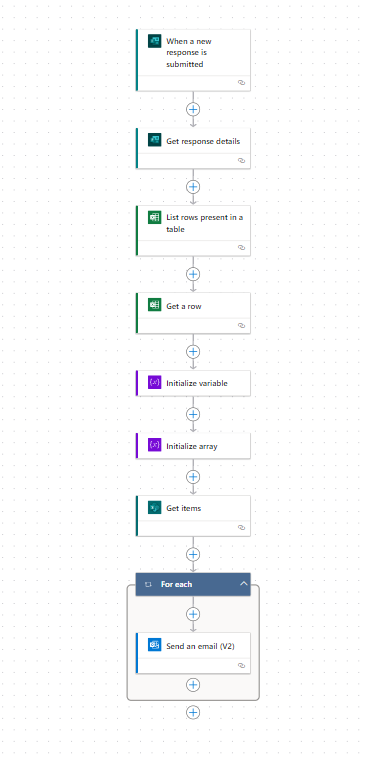
When adding users, be sure to un-check the ‘Send an email invitation’ box under Options. This will prevent an email being sent out to each user notifying them that they have been added to this group.

The last piece of the primary flow is an email that is sent to the appropriate resource with a list of all the invalid buckets:

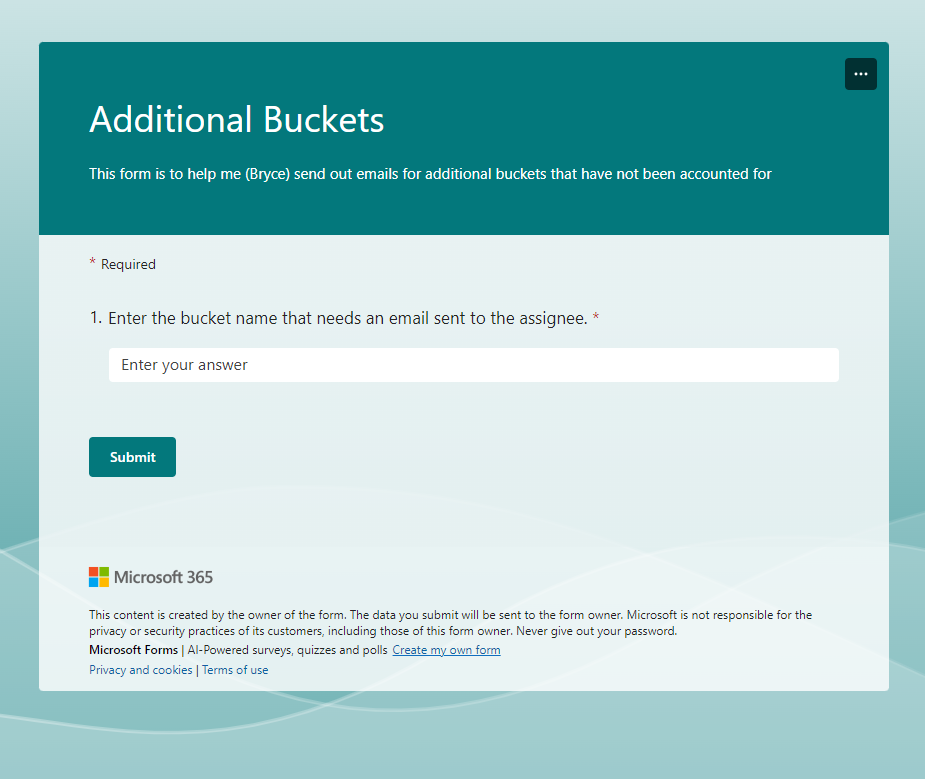


Secondary Flow:

Below is the flowchart used to design the second flow in this effort, which was designed to address all the buckets that errored out during the initial rollout:



This flow is designed to send an email to the assignee of the bucket name that is submitted via a Form:



The flow will take the bucket name, grab the assigned person for the item that has that bucket as the title, and send them an email notifying them of assignment, instructions, and a link to the item. Similar to the first flow but configured differently where an email is sent one at a time rather than all at once, like the primary flow. This allows for better error handling in this flow and better quality control.