O365/Azure FLow – URI [Linked] SharePOint-Documents

# AS A – I WANT – SO THAT

1. **AS A**: User/Admin of third-party-line-of-business software – **I WANT**: To click on a link containing custom parameters pointing to a SharePoint page so that I can reach the appropriate list-item/folder/document-set if already exists/created otherwise/else *creates new* *automagically* (workflow-activity) - with the metadata/details/information of the specific record in my ‘third party software’ – **SO THAT:** I can have an entity record in my Third-party-software/ERP-Line-Of-Business-App/Database integrated/linked with SharePoint list-item/folder/document-set to upload documents associated with such entity-record … kind’a – CRM integration with SharePoint?

\*\* **Disclaimer**: Above use-case is too technical and binding to solution!

# Solution ApProach

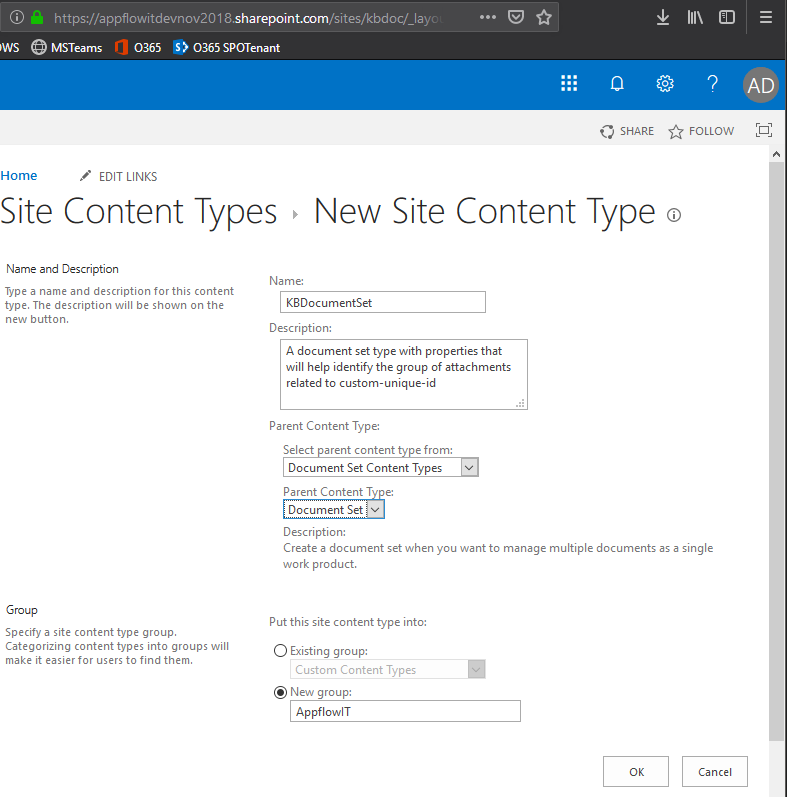
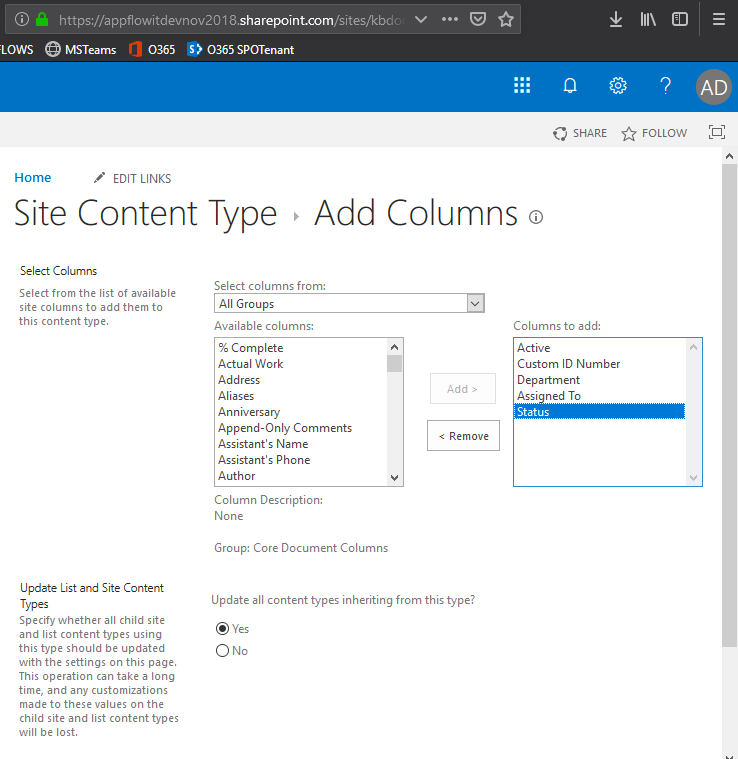
1. Third party System/Software/Program/Script forms a URL link to SharePoint page with agreed query string parameters.
2. When user clicks on such link – they are brought to a generic SharePoint page – which triggers a “Http Request” MS-Flow with parameters from the SharePoint page URL.
3. The MS-Flow will search based on a unique field/parameter value for list-item/folder/document-set in the specific Document Library/List
   1. If it does not find one – the Flow actions to create one with all metadata appropriately updated.
4. In either case, it forms the URL-Path to the item/folder/document-set and returns as a “Http Response”.
5. On receiving such response, the SharePoint page – redirects to such URL.
6. Further enhancements and complexities can be considered in terms of validations, input-forms/confirmations, security etc.

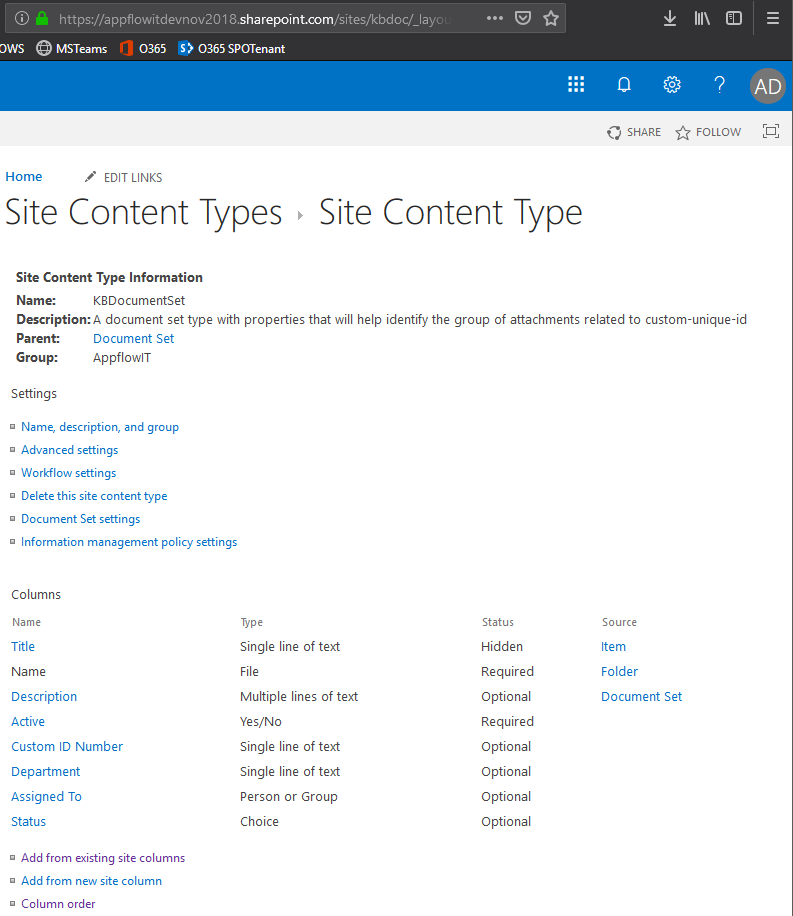
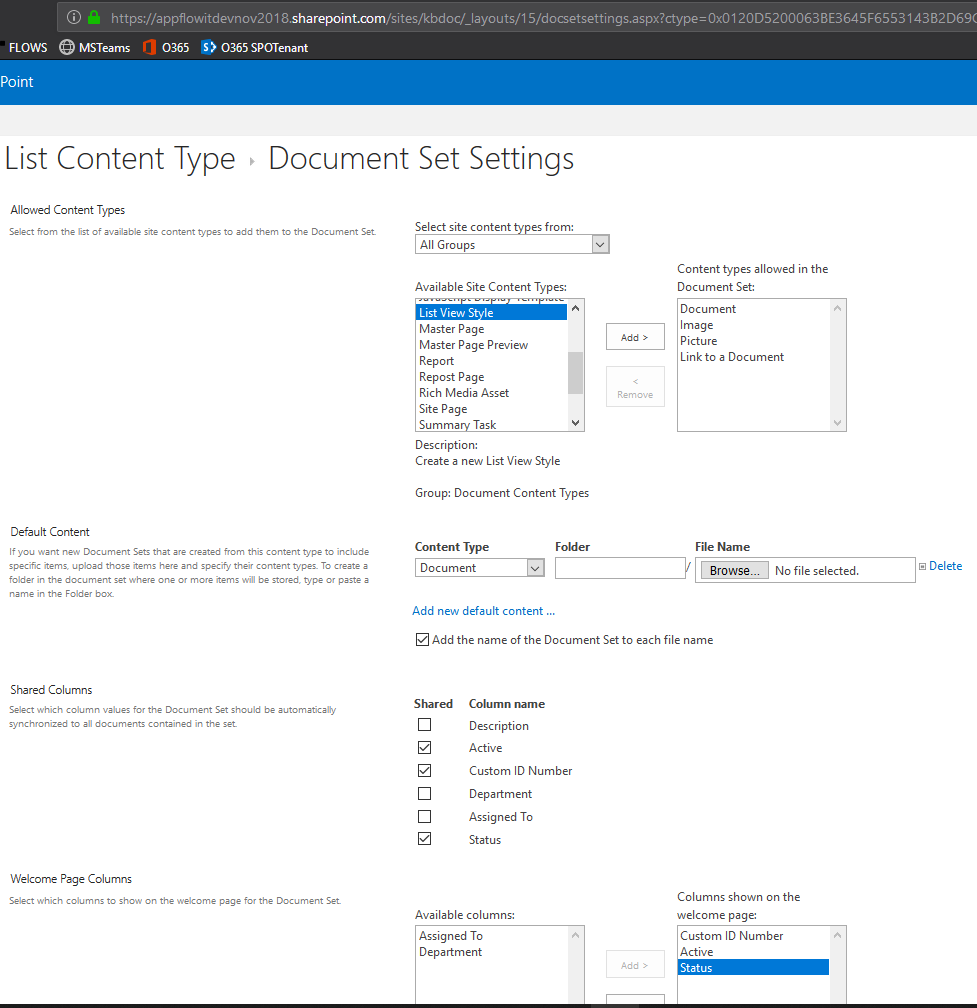
\*\* **Disclaimer**: Using Document Set in this implementation below - is not ultimately-necessary or the only option and alternative simpler-solutions like creating a simple SharePoint List with Key field(s) that then points to a Folder in any/other Library – Would even be more desirable (and simpler) for many scenarios. In this implementation, I stuck around with Document Set option to explore and work with SharePoint REST-API and Flow integration.

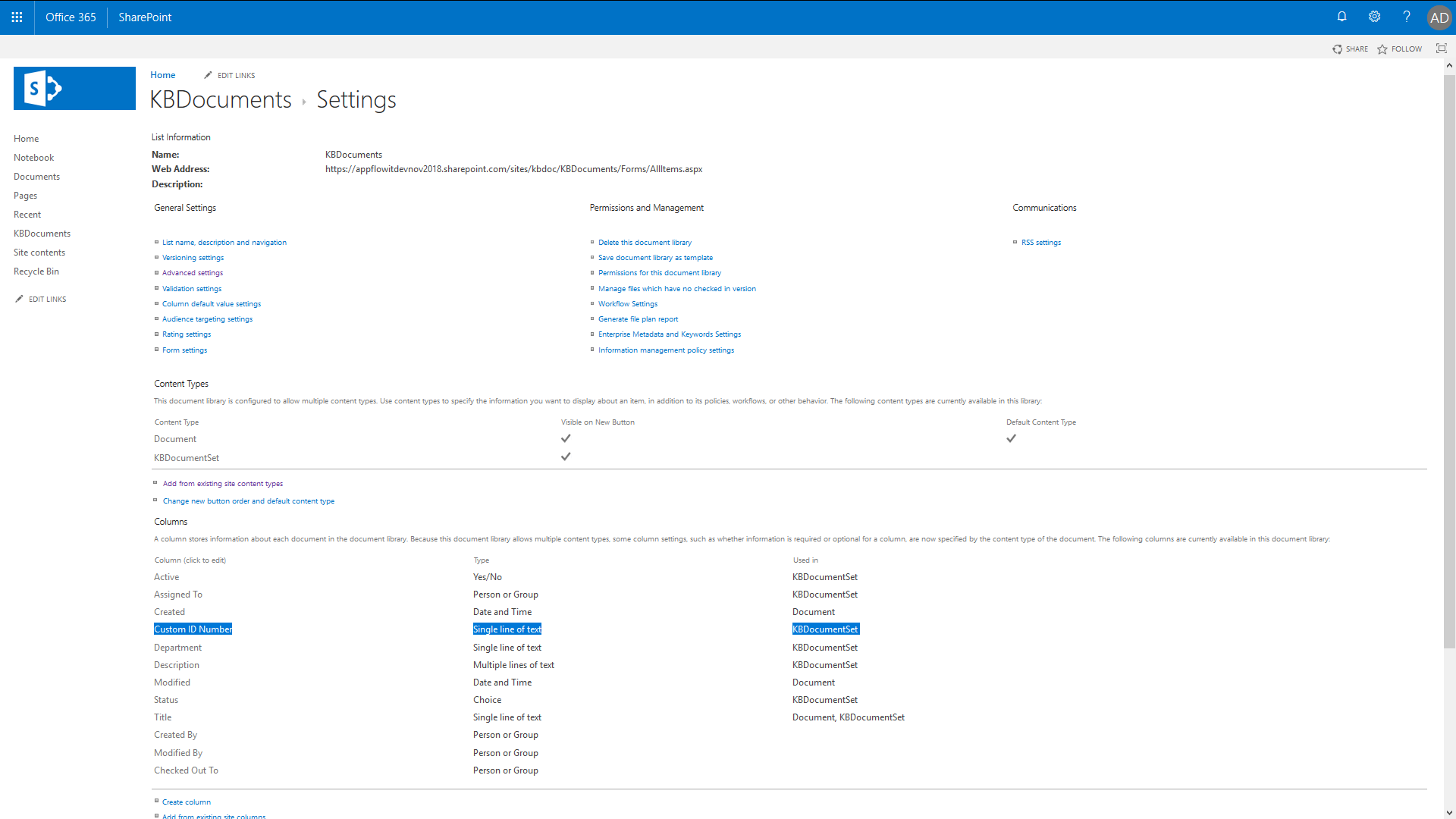
# sSOLUTION DESIGN

## Infomration Architecture

**Content Type**: KBDocumentSet

**Document Library:** KBDocuments(KBDocumentSet content type added)

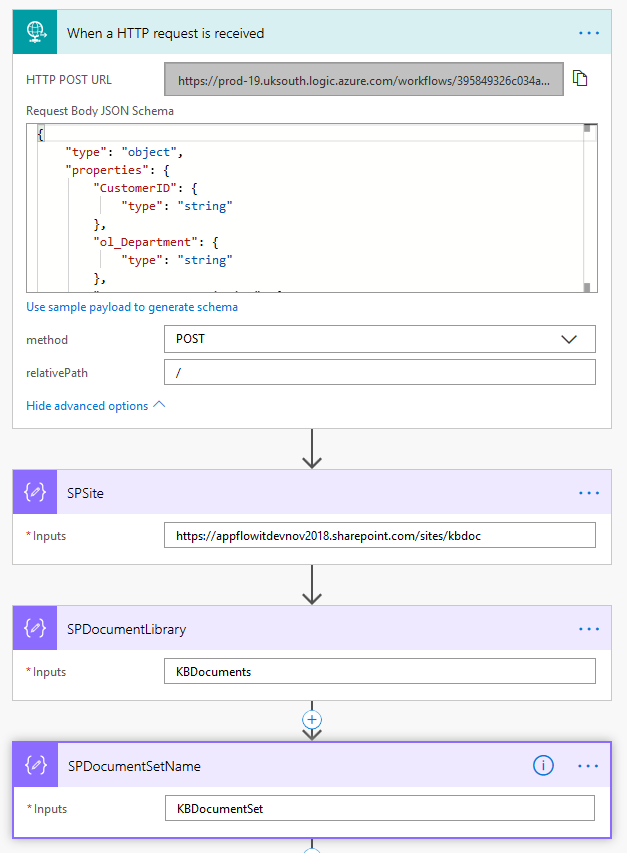
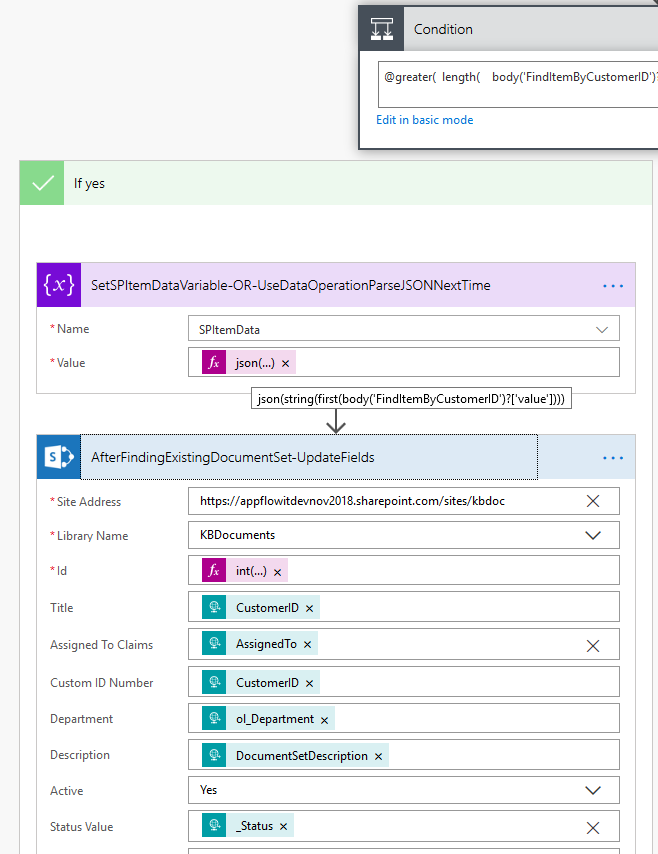
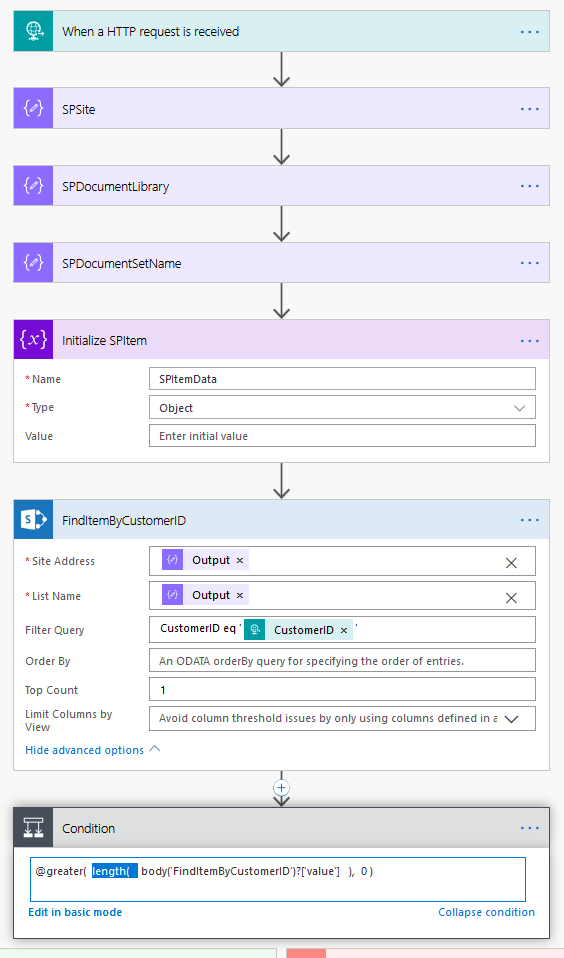
|  |  |  |
| --- | --- | --- |
| **Column (click to edit)** | **Type** | **Used in** |
| ***Custom ID Number*** | Single line of text | **KBDocumentSet** |
| Active | Yes/No | **KBDocumentSet** |
| Assigned To | Person or Group | **KBDocumentSet** |
| Department | Single line of text | **KBDocumentSet** |
| Description | Multiple lines of text | **KBDocumentSet** |
| Status | Choice | **KBDocumentSet** |
| Title | Single line of text | Document, **KBDocumentSet** |
| Created By / Modified By / Checked Out To | Person or Group |  |
| Created / Modified | Date and Time | Document |

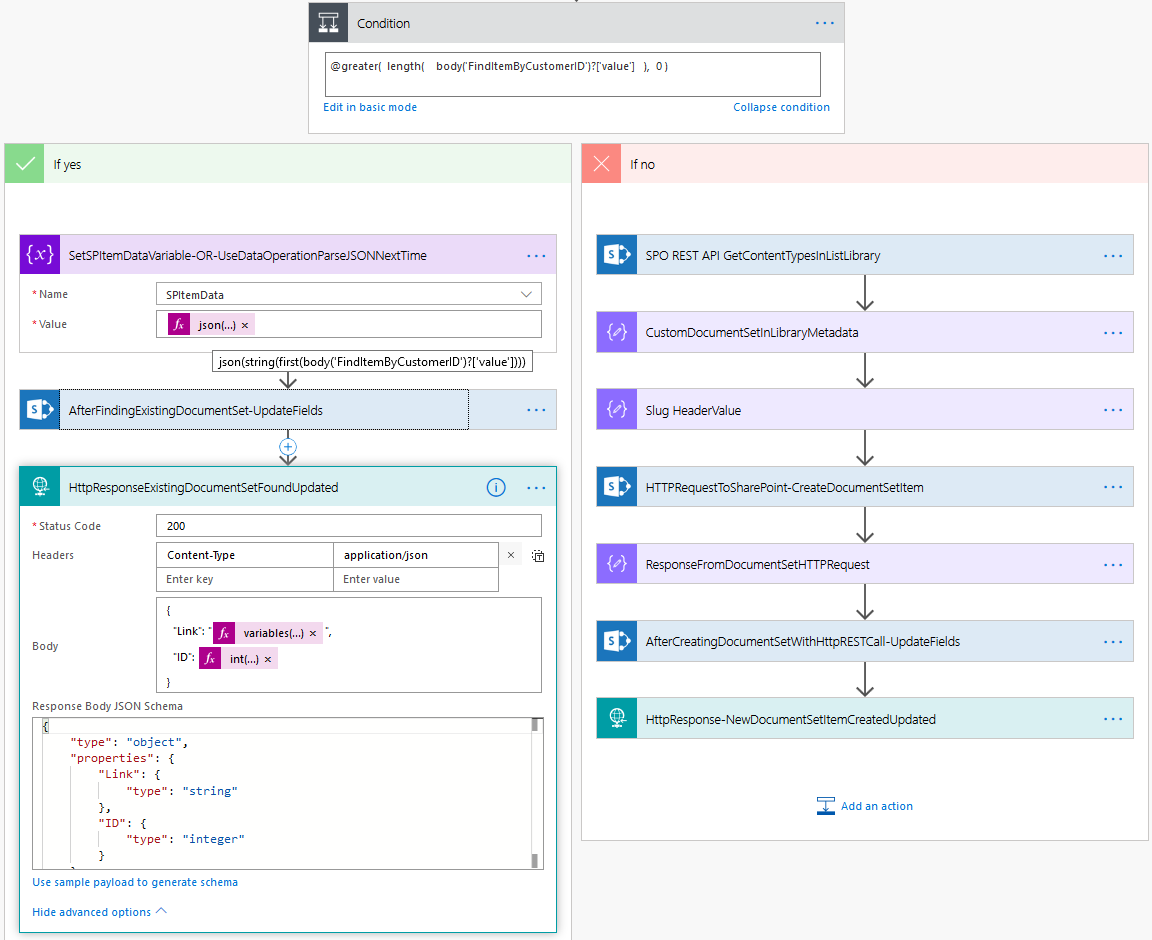
## FLOW DESIGNER

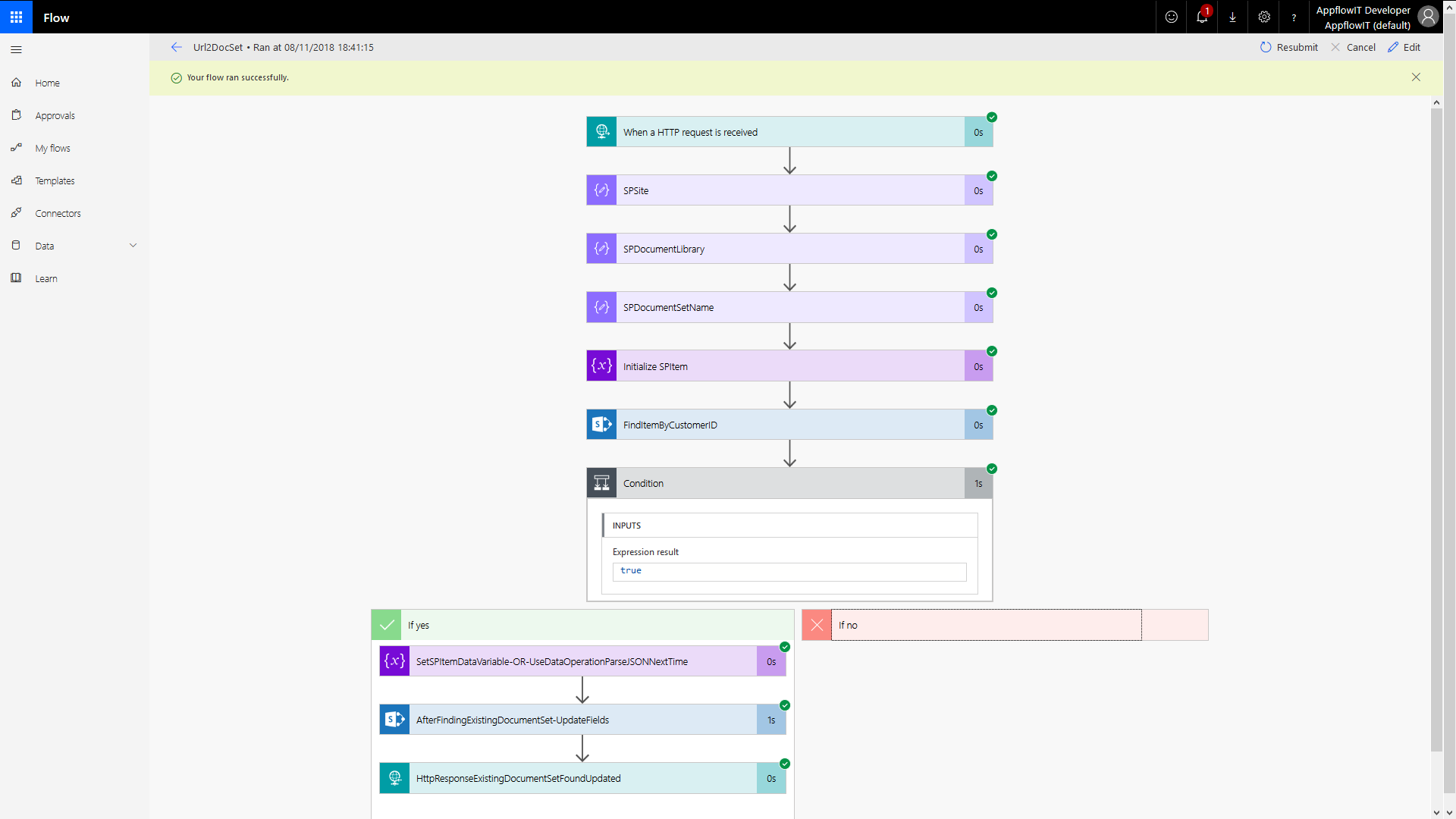
The flow is triggered with a HTTP (POST) Request with a JSON payload with parameters from the LineOfBusiness-URL Transformed-by-SharePoint-Page-Script (JQUERY/Ajax)!

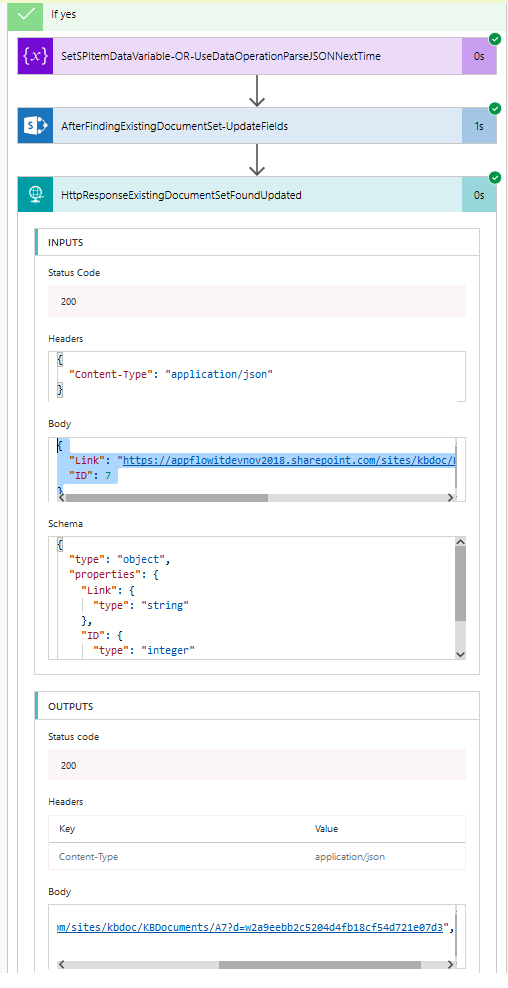
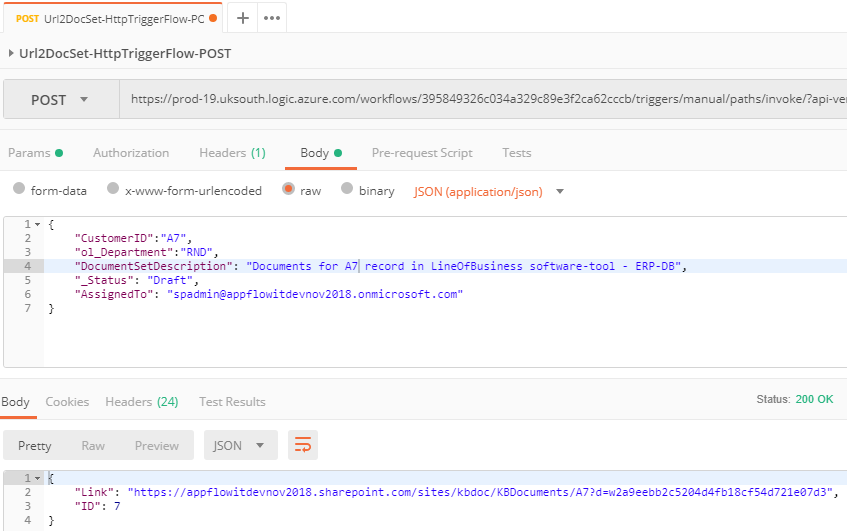
\*\* Tip when working with (HTTP) Flows 🡺 Do it in parts and output the Responses to peek in the results/schema/data.

## PART 1 🡺 REQUEST + UPDATE (DOCUMENT SET) + RESPONSE



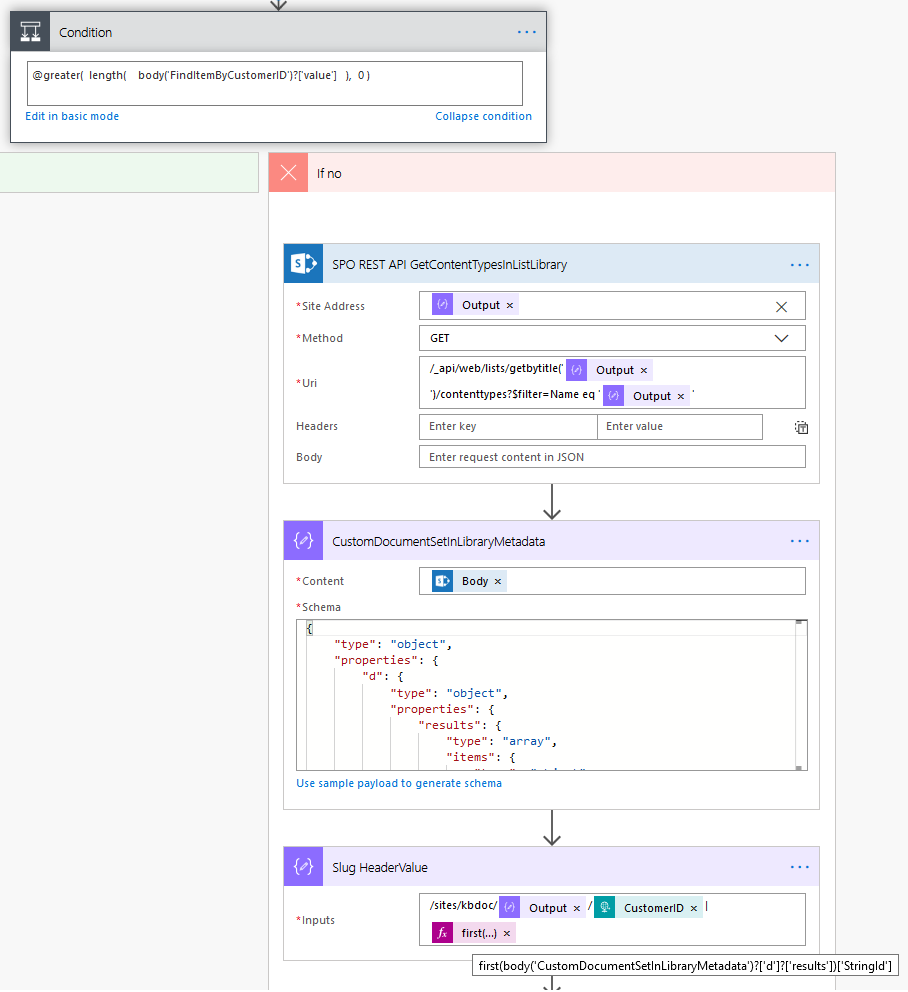


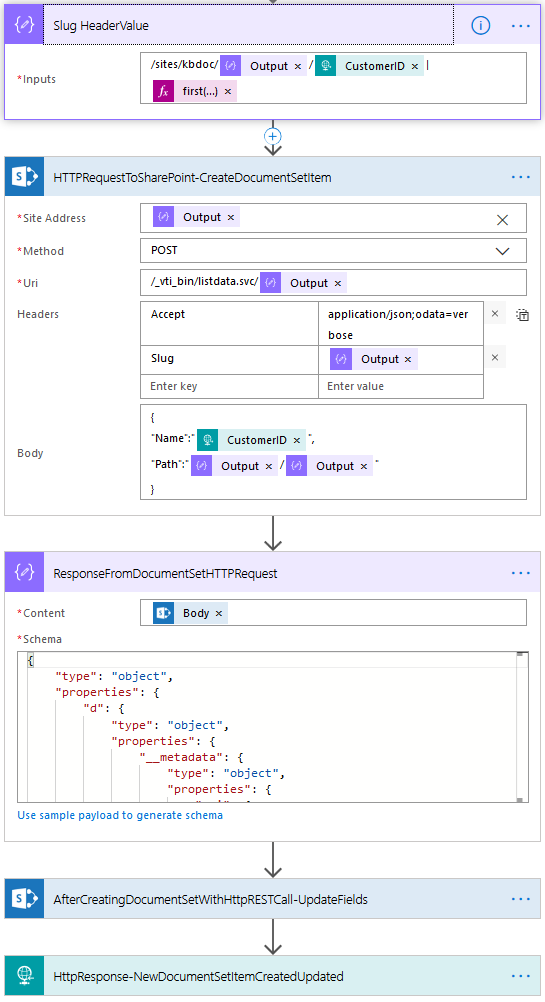
 

## PART 2 🡺 REQUEST (Same as Part-1) + CREATE NEW (DOCUMENT SET) + RESPONSE

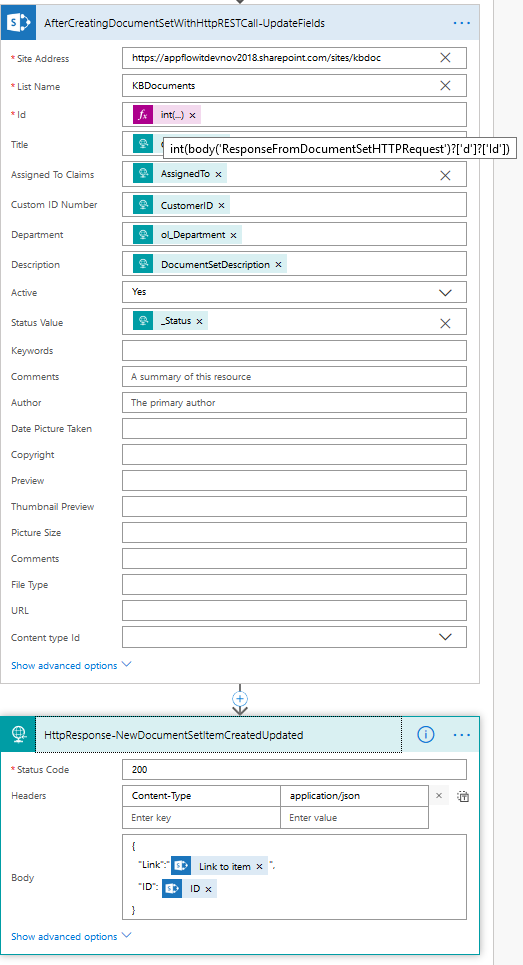
Ref-Only: <https://sharepointycl.wordpress.com/2018/05/03/creating-sharepoint-online-document-set-in-microsoft-flow/>

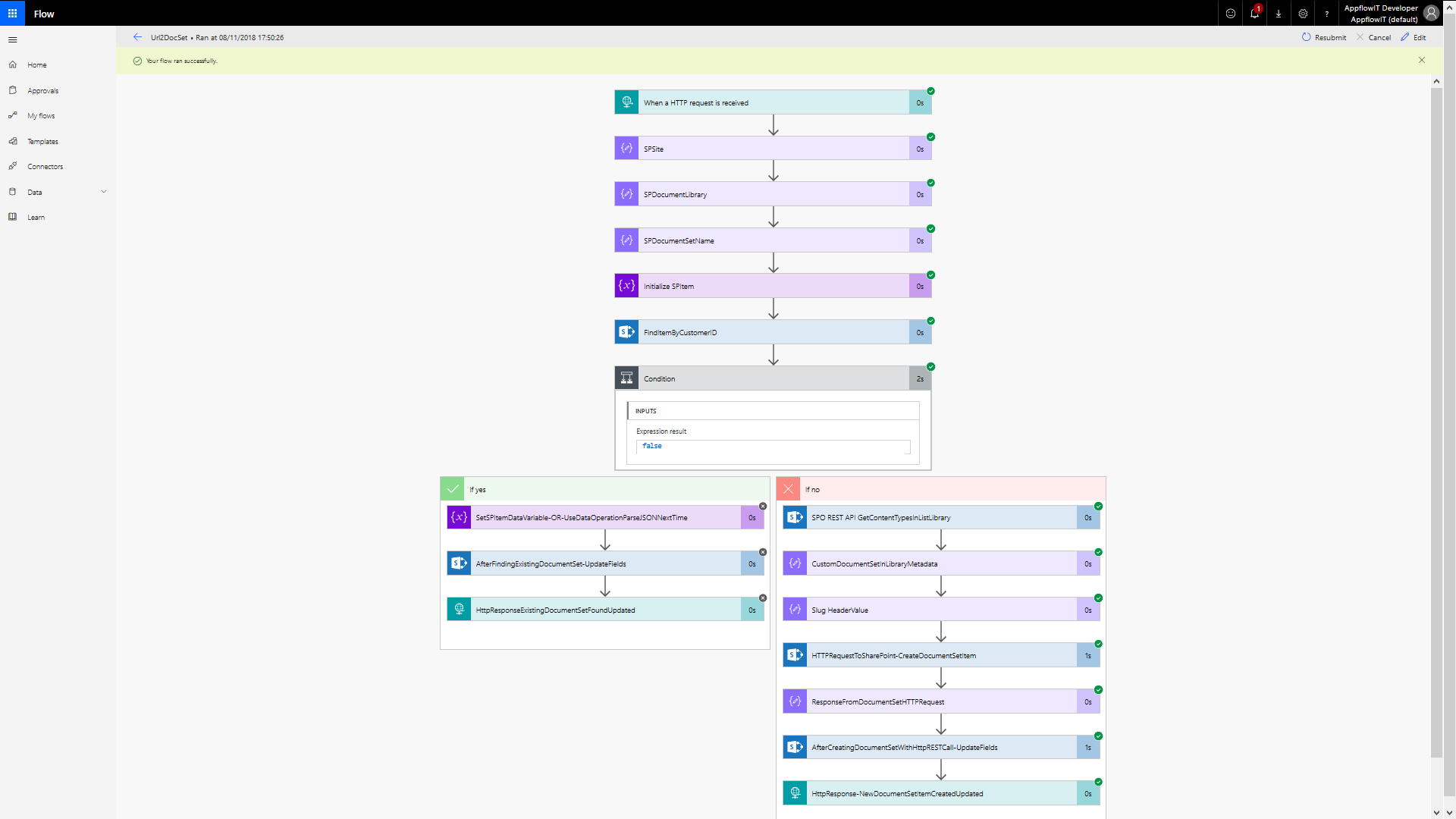
Ref-Only: JavaScript client call to REST API: 

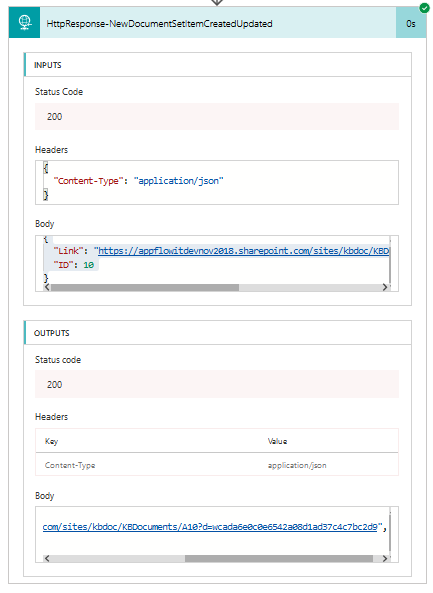
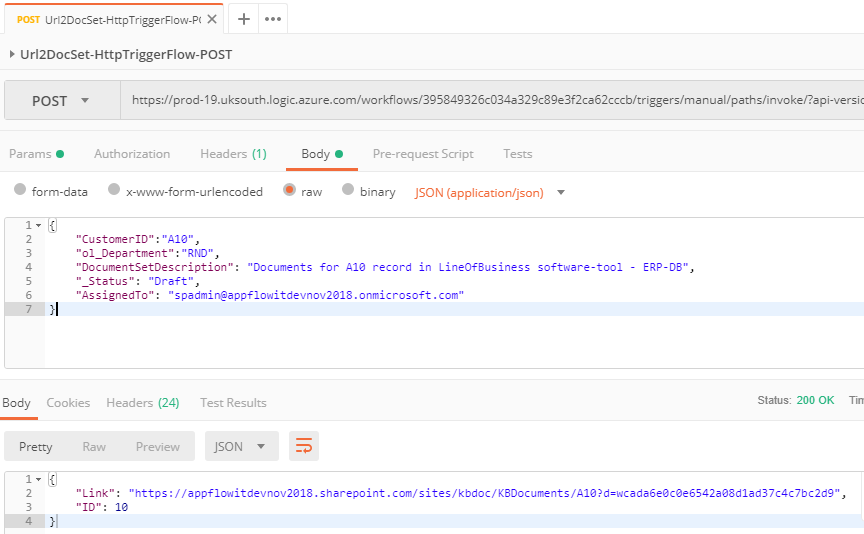




**first**(body('CustomDocumentSetInLibraryMetadata')?['d']?['results'])[**'StringId'**]





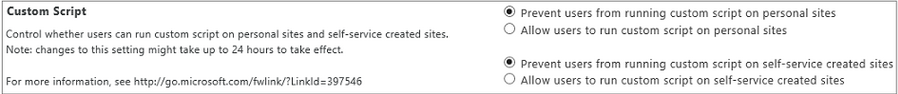
 

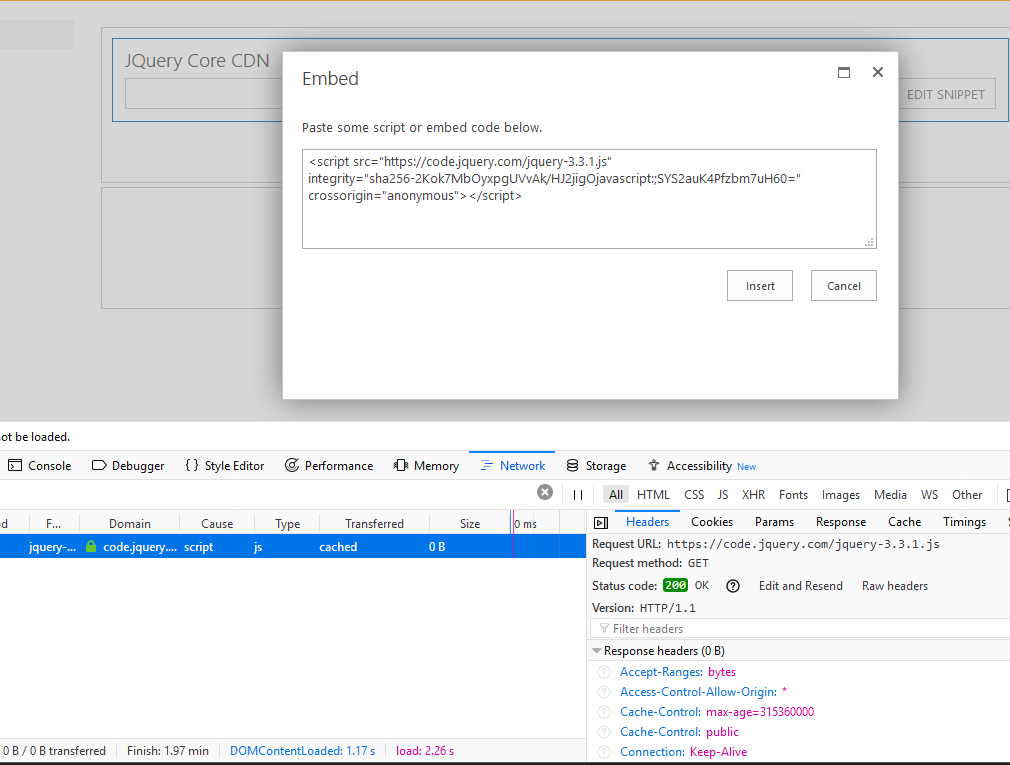
# SUPPORT – ENVIRONMENT – TOOLS – CONFIGURATION

## TOOLS: SOAPUI | POSTMAN (Chrome) | FIDDLER

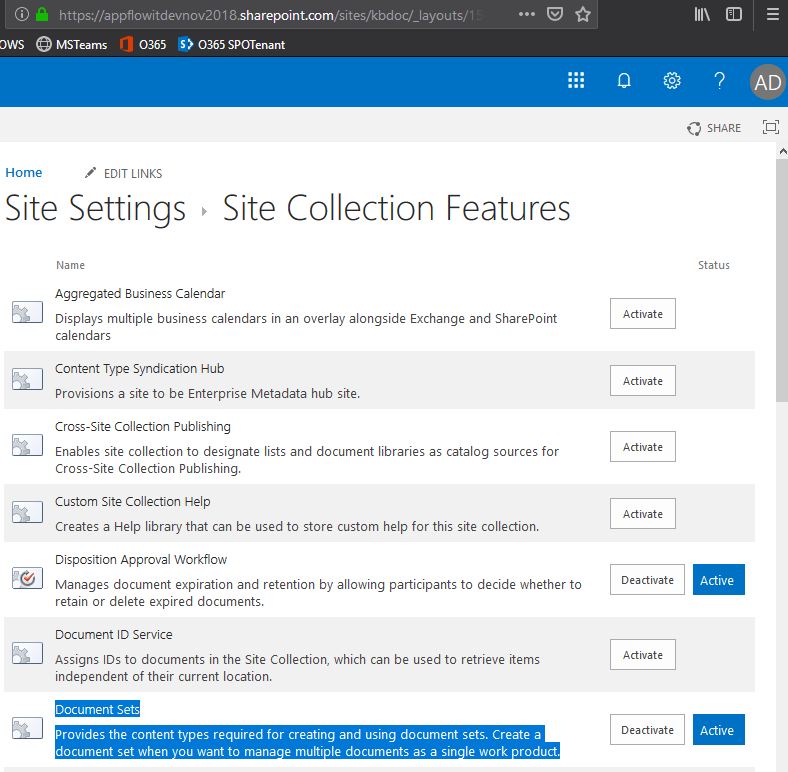
1. SOAPUI: Download and Install OSS(free) version from - <https://www.soapui.org/downloads/soapui.html>
2. **POSTMAN**: Download and install for Chrome from - <https://www.getpostman.com/download?platform=win64>
3. CAML-QUERY:
   1. <https://saketa.com/sharepoint-caml-query-builder/>
   2. [https://**www.biwug.be**/resources](https://www.biwug.be/resources)

## CONFIGURATION: **AllowING** custom script on sitecollection pages





## CONFIGURATION: Ensure Document Sets Site-Collection Feature enabled

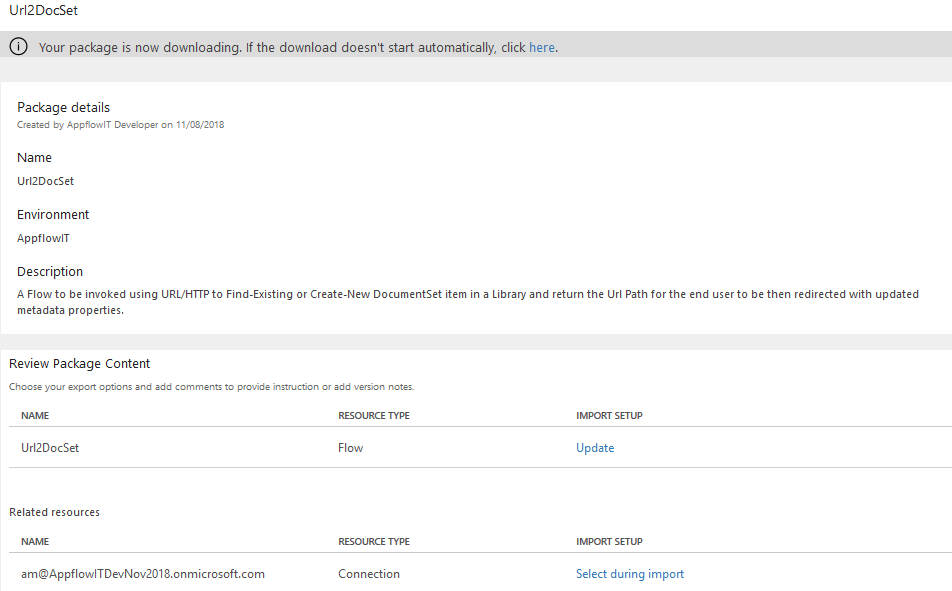


# HOPE THIS HELPED!?

Kindly feedback your thoughts and suggestions and ideas, that would be magical to get them😊

Cheers, Aaseaf Mahkahti | [sharepointjob@outlook.com](mailto:sharepointjob@outlook.com)

\* Disclaimer: The purpose of this write-up/document is partially for reference/blogging – partially for self-promotion, hence not everything is necessarily elaborated step-by-step and word-by-word.



# DISCLAIMER

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples depicted herein are provided for illustration only and are fictitious.  No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Appflow IT Solutions Ltd. products. You may copy and use this document for your internal, reference purposes.

© 2018 Appflow IT Solutions Ltd. All rights reserved.