Hands-on Lab: Building Joule Skills in SAP Build

Aleksandrs Antonuks, Vass, aleksandrs.antonuks@vasscompany.com

Michael Pytel, Vass, michael.pytel@vasscompany.com

Sheldon Lipshitz, Vass, sheldon.lipshitz@vasscompany.com

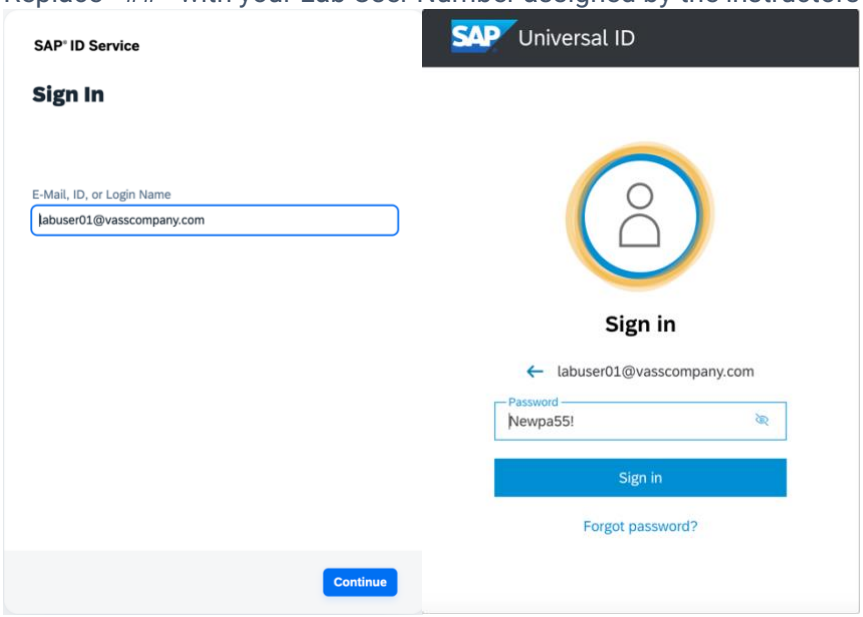
https://vasscompany.com

Index

Table of Contents

Environment Information

1. Please use GOOGLE CHROME as your browser
2. SAP Build App URLs
   - SAP Build Apps
3. Lab User Information
   - User: **labuser<##>@vasscompany.com**
   - Password: Newpa55!
4. Replace <##> with your Lab User Number assigned by the instructors

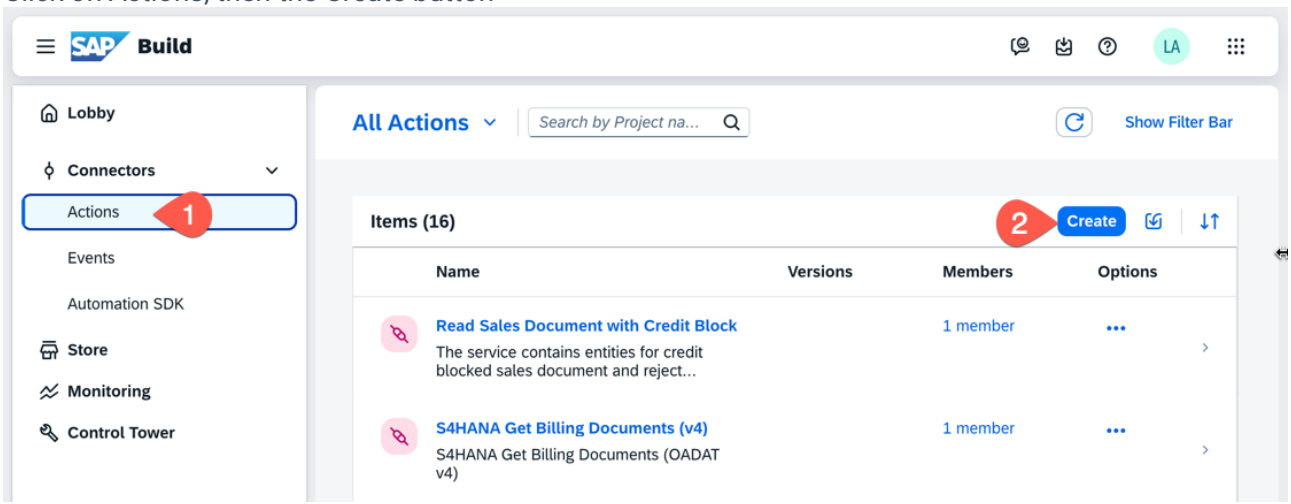


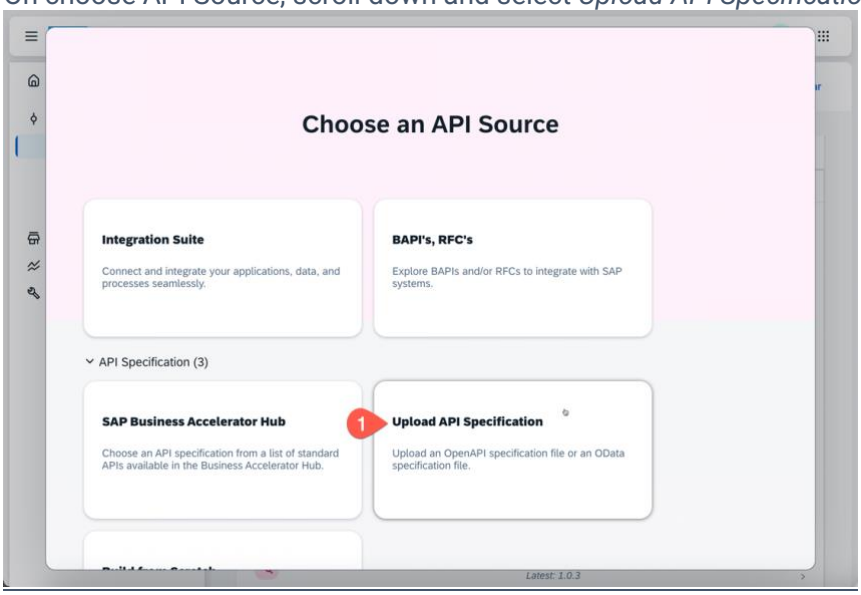
Lab Goal

We will be building a Joule Skill that can review Sales Orders with the status 'Credit Hold' and then help the business user draft an email to notify the customer of their open Invoices to be paid before the Sales Order will be processed.

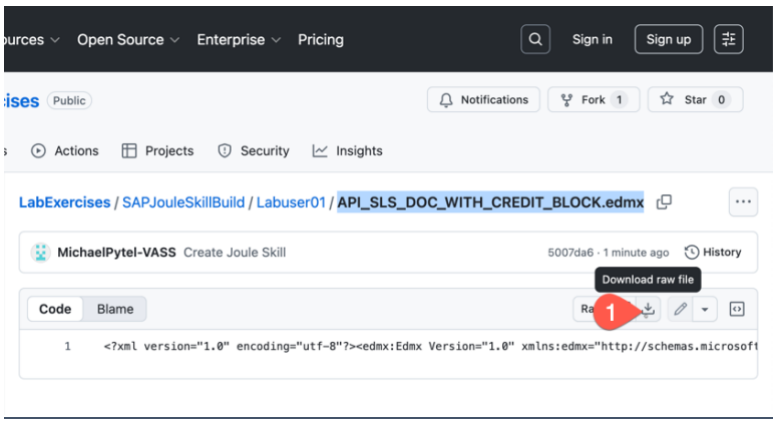
Create Actions for Joule

Our first step is to create Actions for Joule to use this lab. An action connects to an ODATA API.

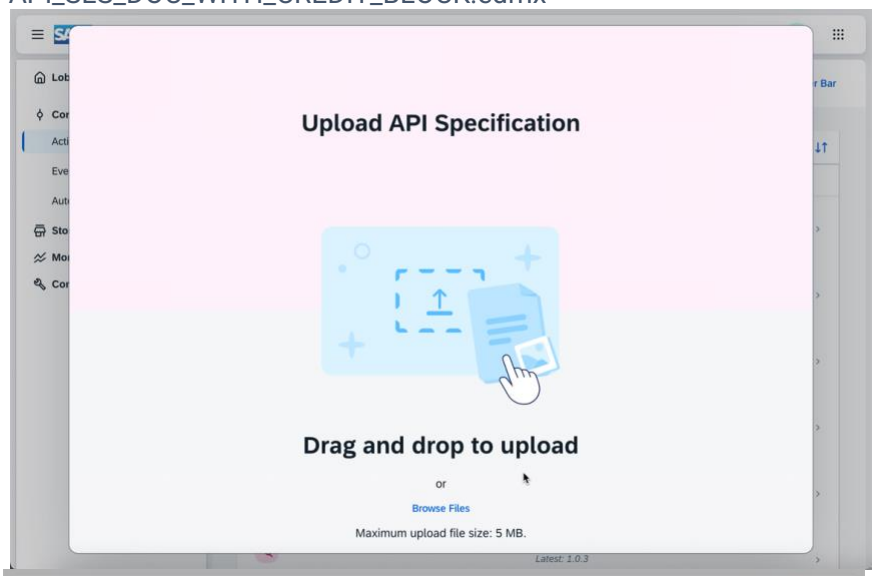
1. Open an Chrome Browser and navigate to SAP Build Apps
2. Enter your Lab UserID and Password
3. Click on Actions, then the Create button
4. 

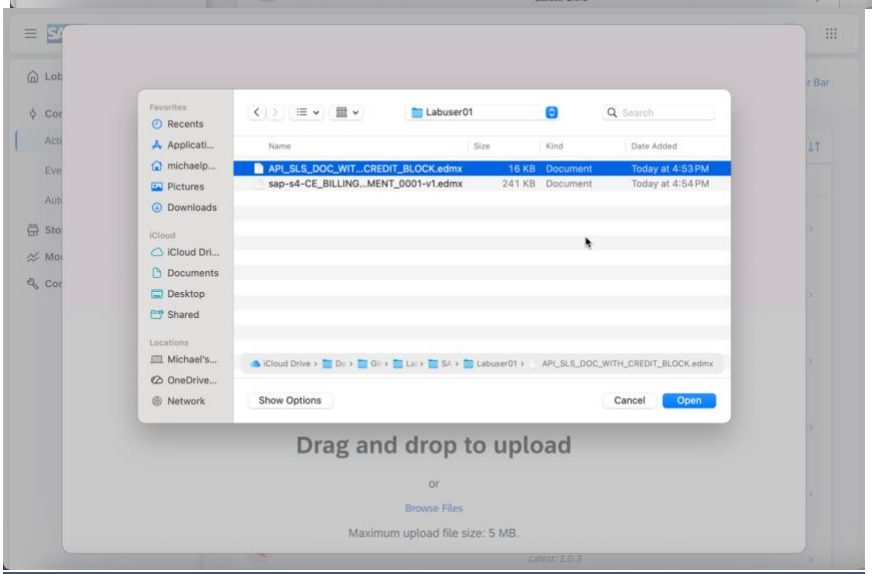
5. On choose API Source, scroll down and select *Upload API Specification*
6. 

7. Browse to your Labuser## folder on Github to download the two EDMX files
    a. https://github.com/MichaelPytel-VASS/LabExercises/tree/main/SAPJouleSkillBuild
    b. Open you labuserXX folder
    c. Download the EDMX files to your local computer

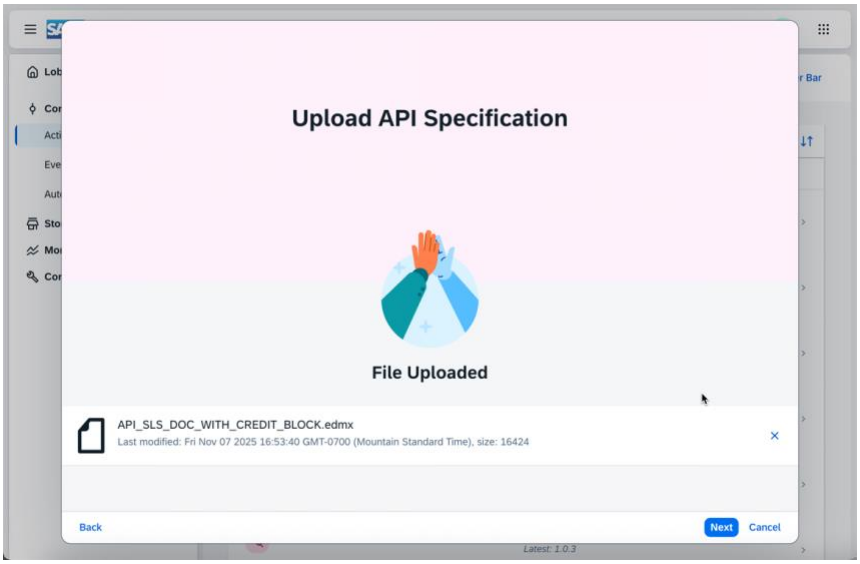
d. 

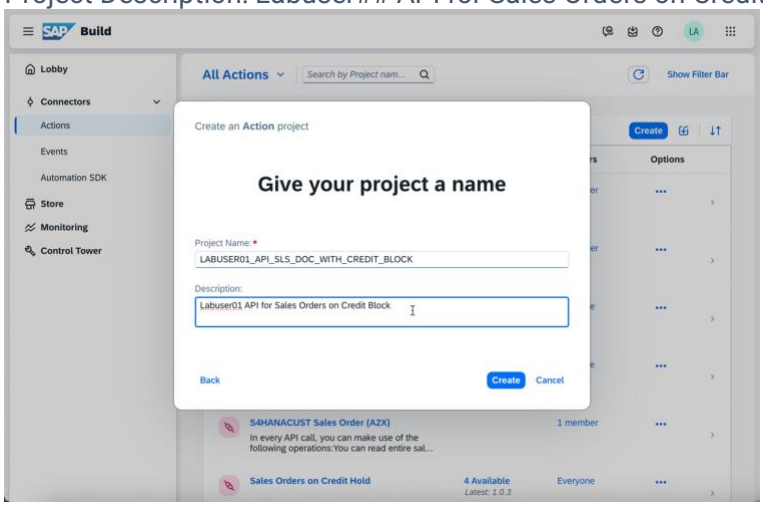
8. When prompted upload the first API we will activate ‘API_SLS_DOC_WITH_CREDIT_BLOCK.edmx’

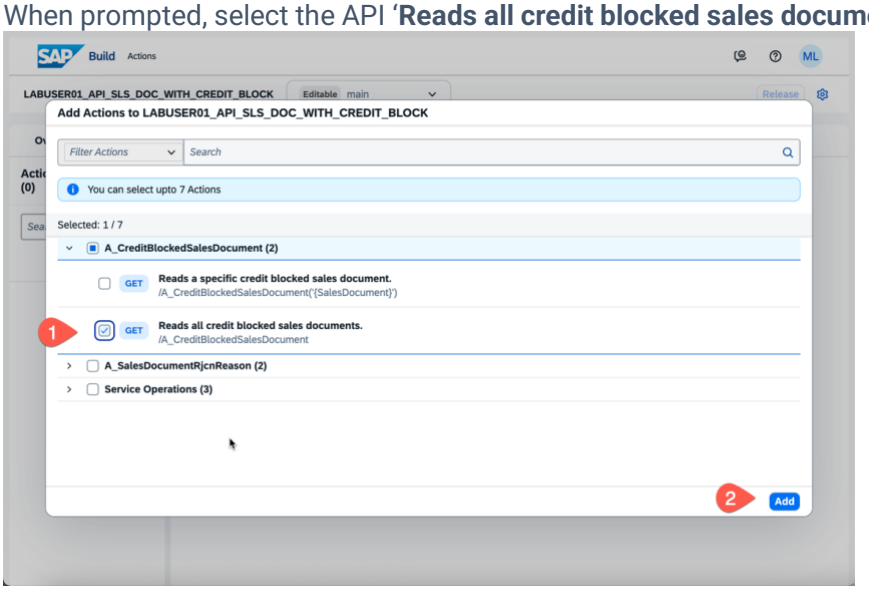
9. 

10. 

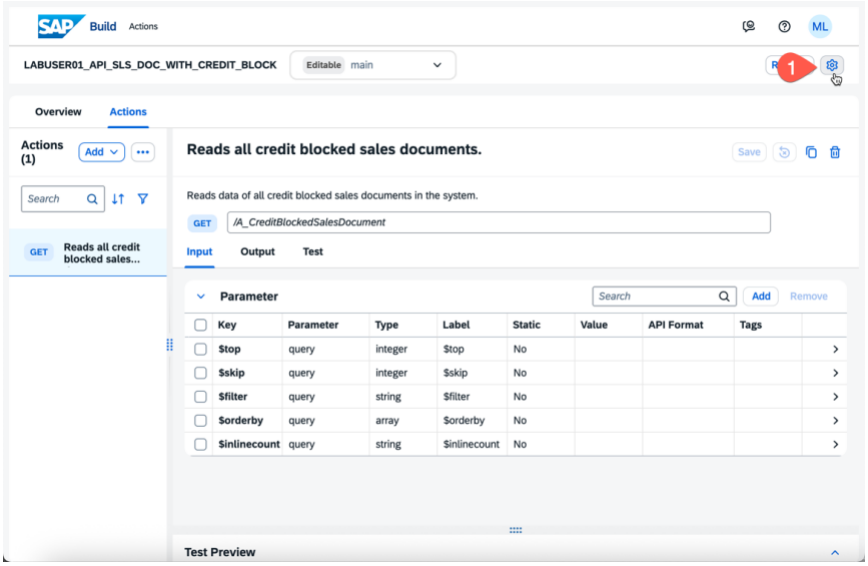
11. When you complete the upload, review and click Next

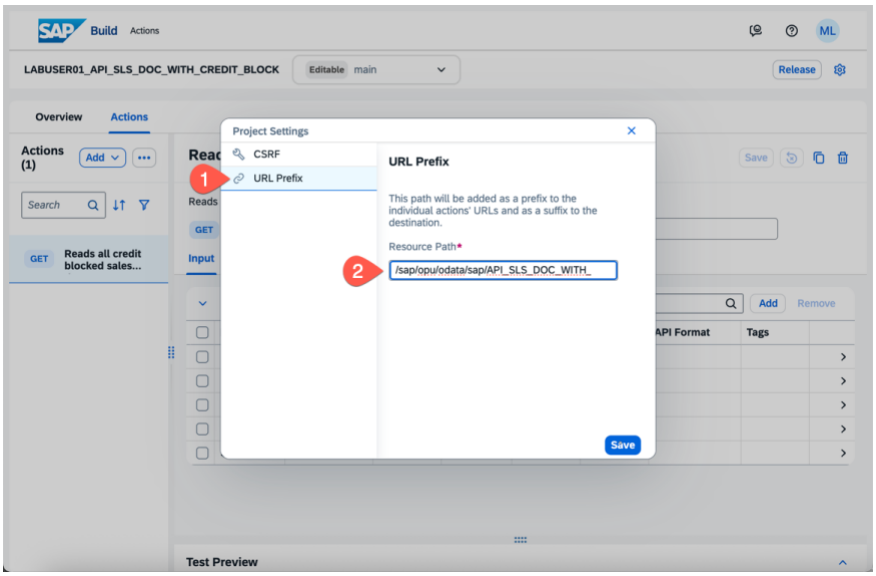
12. 

13. Enter the project name as shown below
    a. Project Name: Labuser##_API_SLS_DOC_WITH_CREDIT_BLOCK
    b. Project Description: Labuser## API for Sales Orders on Credit Block
    c. 

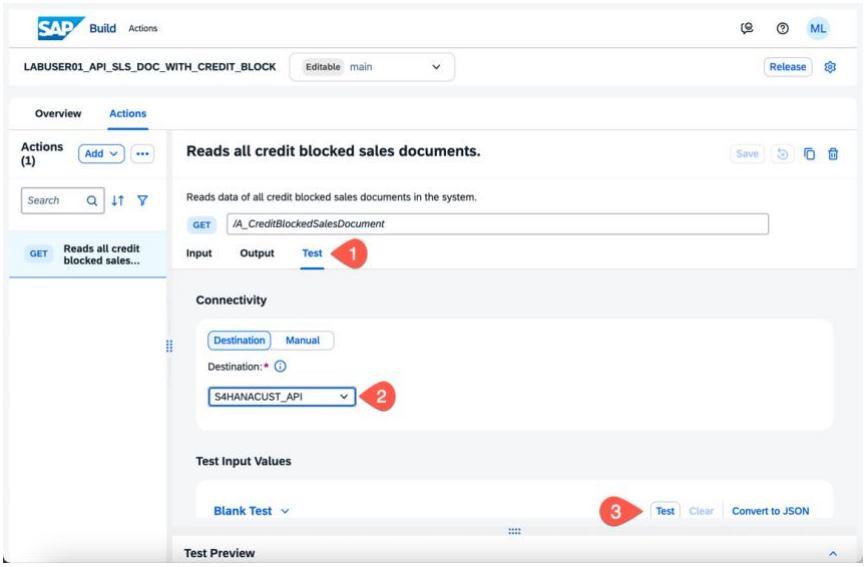
    d. Click Create after entering a name
14. A new browser tab will open; check your pop-up blocker.
15. When prompted, select the API '**Reads all credit blocked sales documents**.' And Click Add.
16. 

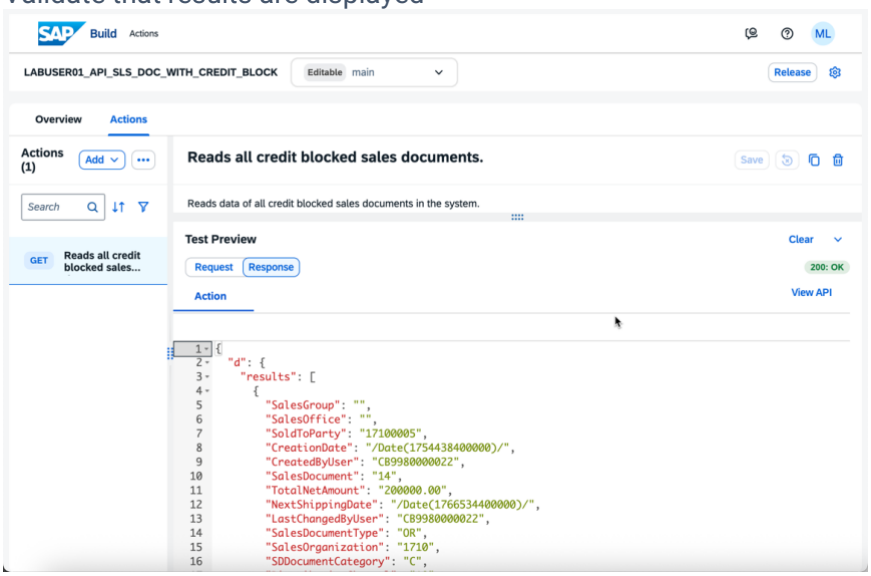
17. Click the Settings Button

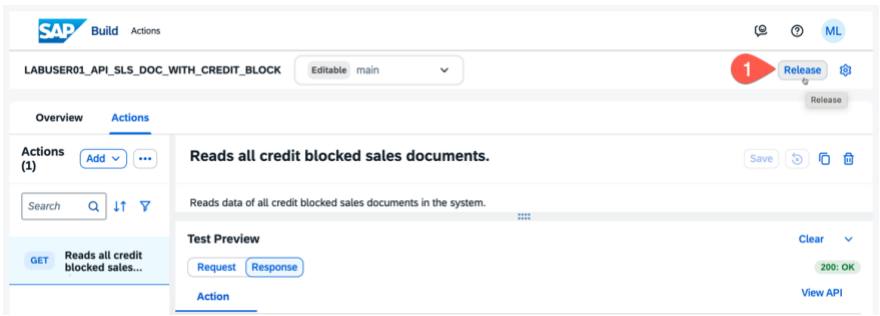
18. 

19. Select *URL Prefix* and then Resource Path
20. 

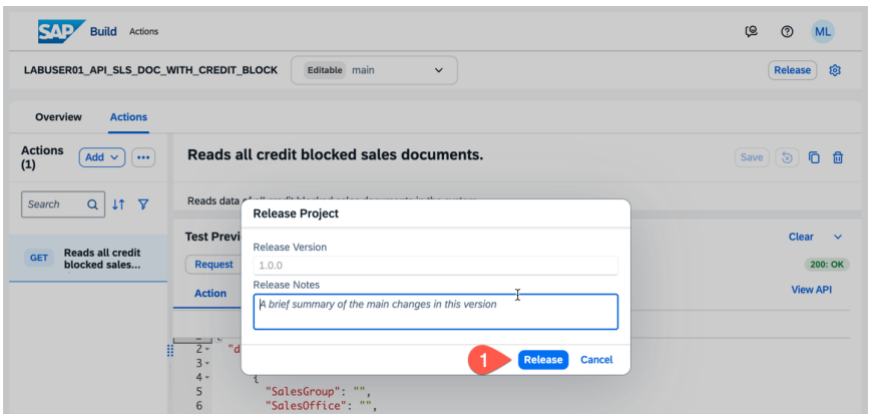
21. Enter in the URL prefix for the API
    a. Path: /sap/opu/odata/sap/API_SLS_DOC_WITH_CREDIT_BLOCK
22. Click Save
23. Next, we're going to test the API.
24. Click the tab for *Test*
25. Select the Destination: S4HANACUST_API
26. And then select the *Test* button

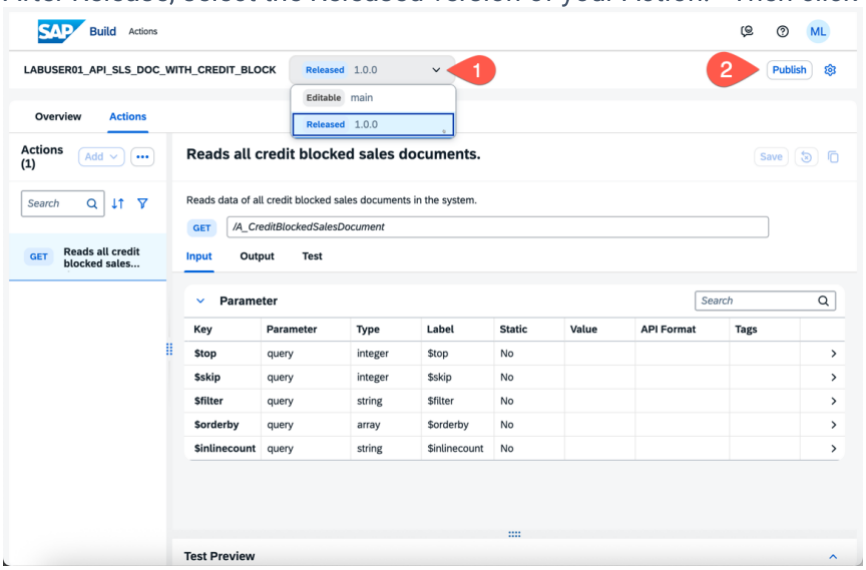
27. 

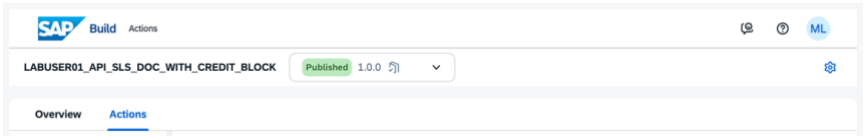
28. Validate that results are displayed
29. 

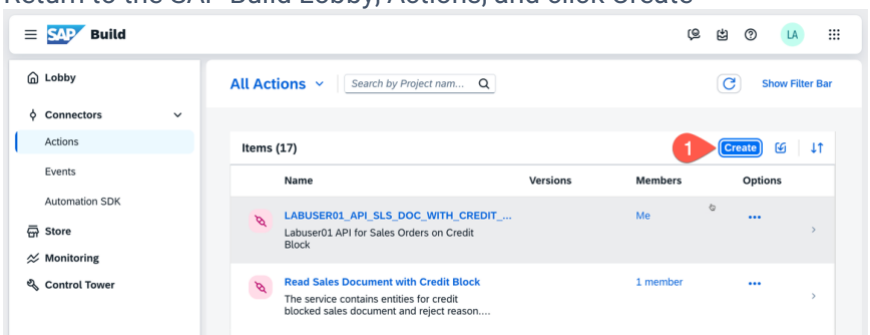
30. Next, we’re going to Release and Publish the Action.
31. Click on Release
32. 

33. Confirm the Release

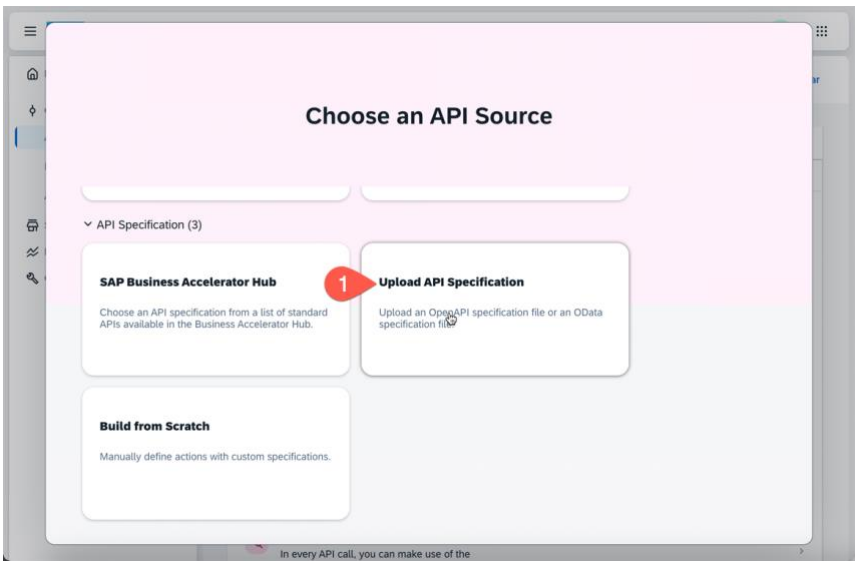
34. 

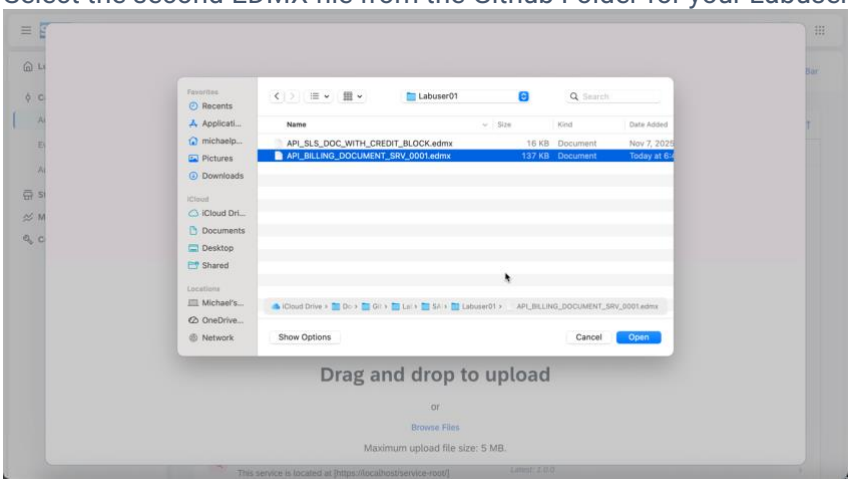
35. After Release, select the Released version of your Action. Then click Publish.
36. 

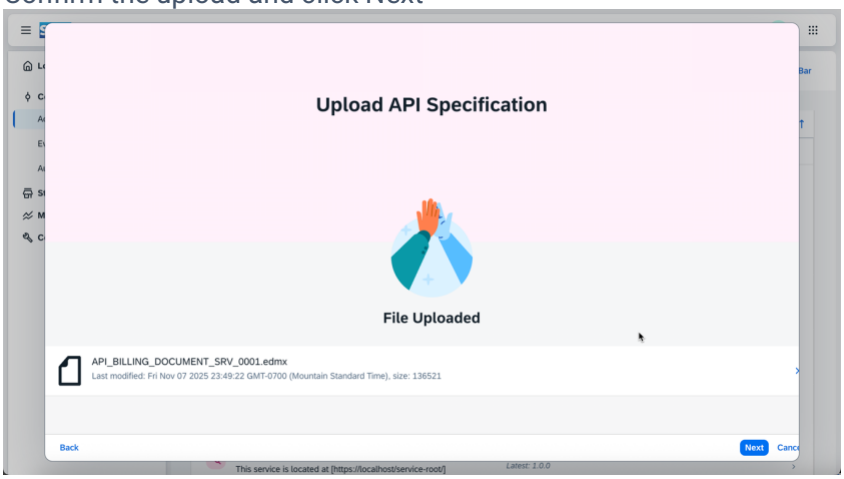
37. Confirm the Action is Published
38. 

39. Next, we're going to create the second Action to read open Customer Invoices
40. Return to the SAP Build Lobby, Actions, and click Create
41. 

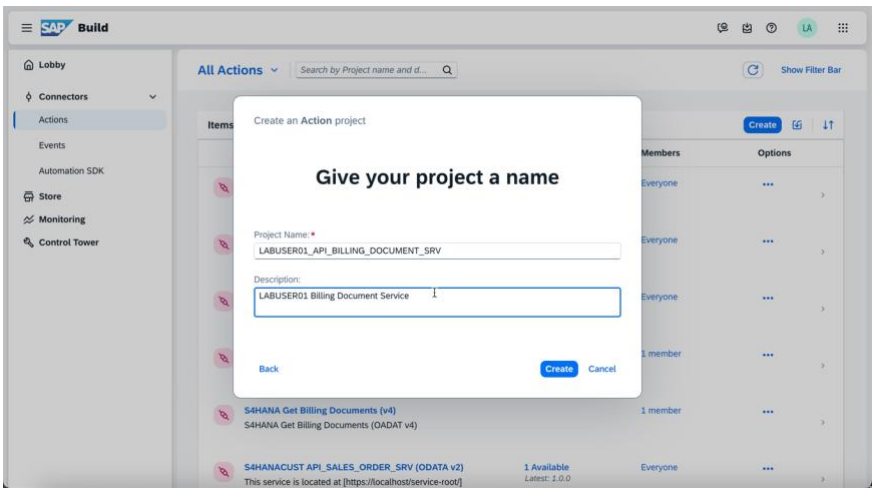
42. Choose *Upload API Specification*

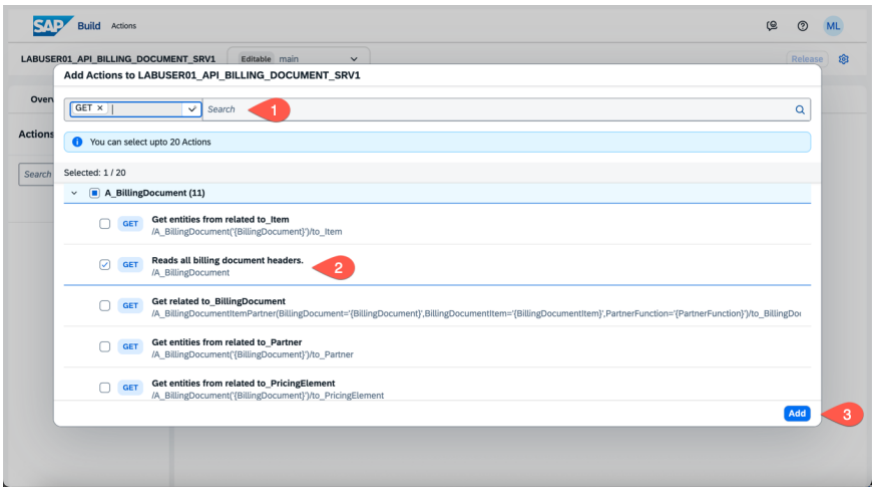
43. 

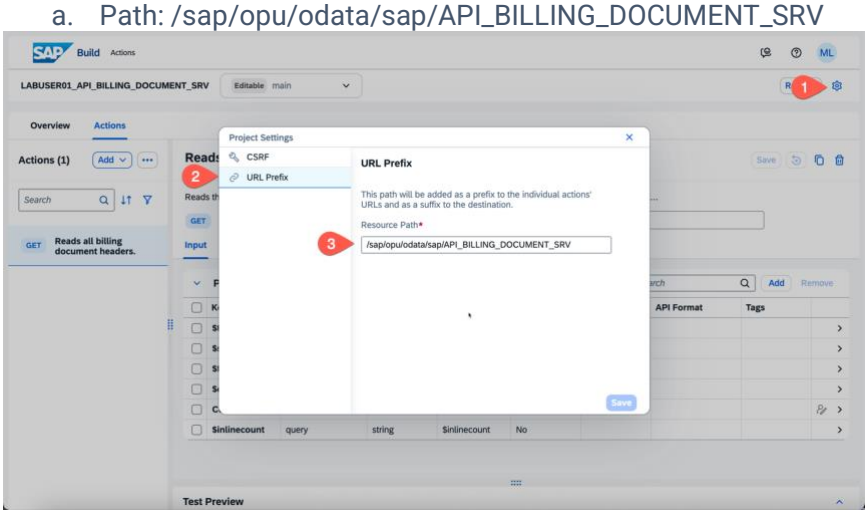
44. Select the second EDMX file from the Github Folder for your Labuser##
45. 

46. Confirm the upload and click Next
47. 

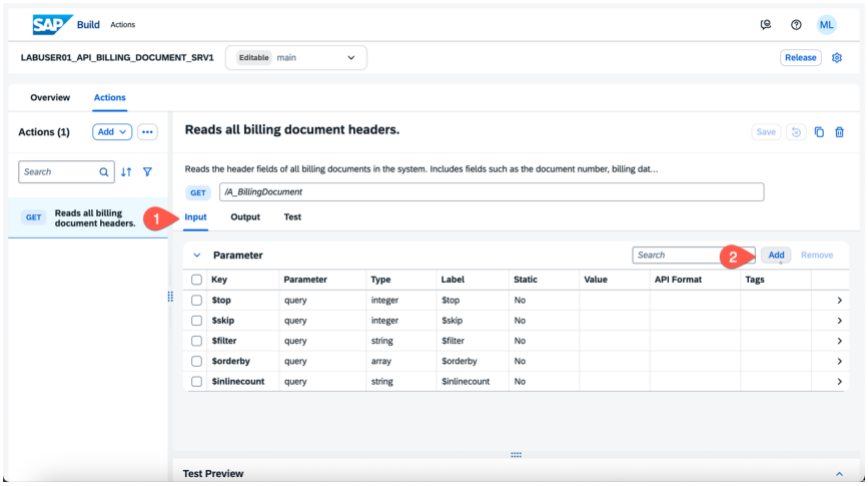
48. Enter a Project Name and Project Description
    a. Project Name: LABUSER##_API_BILLING_DOCUMENT_SRV
    b. (replace ## with your labuser number)

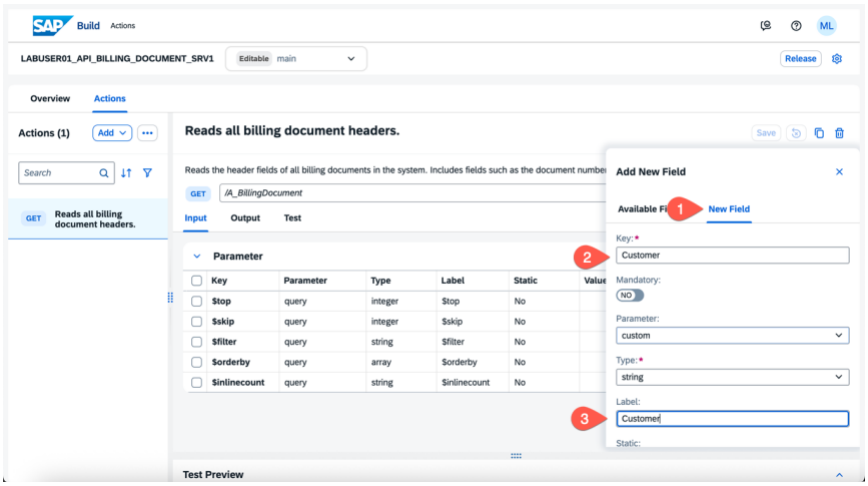
49. 

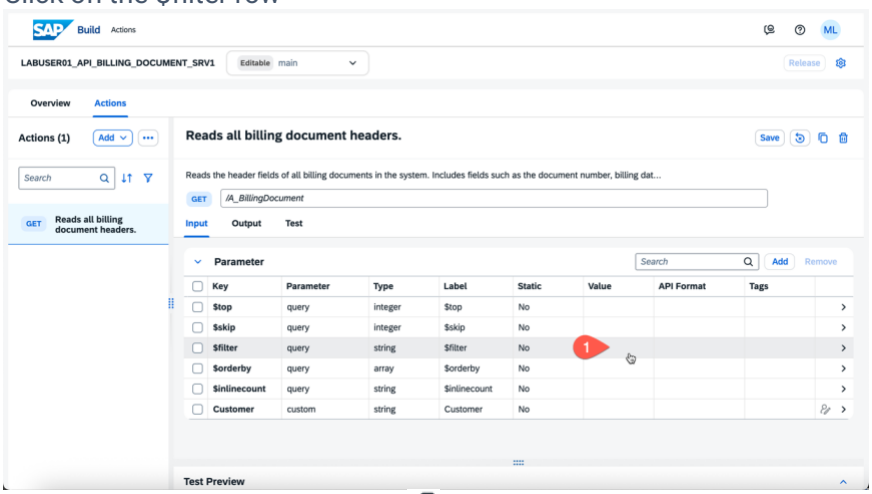
50. Click Create
51. A new browser tab will open (confirm your popup blocked is disabled)
52. When prompted to add Actions, set a filter for ‘GET’ actions.
53. Scroll down and select the API ‘Reads all billing document headers’
54. Click Add
55. 

56. Click the Settings icon on the upper right
57. Select URL Prefix then Enter the Resource Path below:
    a. Path: /sap/opu/odata/sap/API_BILLING_DOCUMENT_SRV
58. 

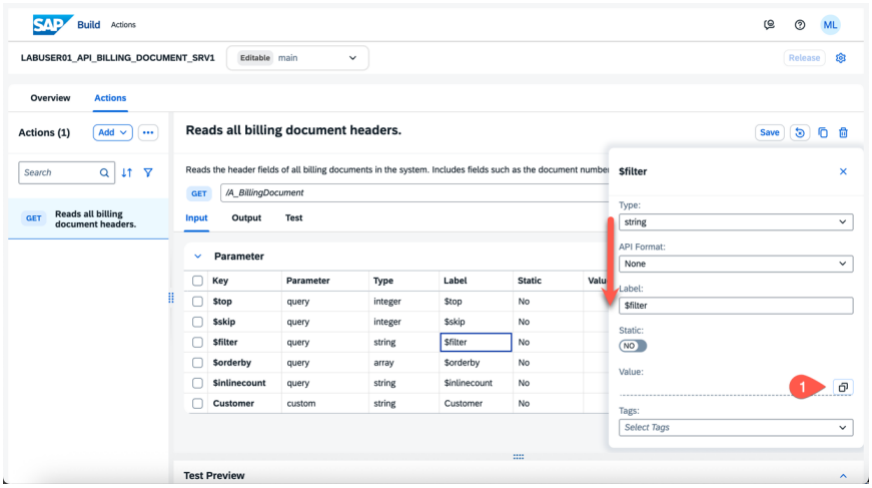
59. Click Save
60. Next, we’re going to add a field to help us build the filter for the ODATA Service.
61. On the Input Tab, Click Add

62. 

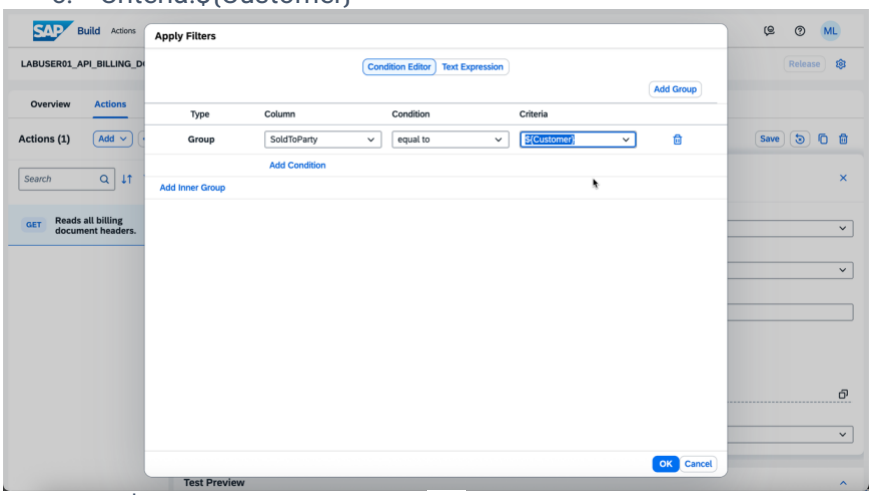
63. Click New Field and then enter the following
    a. Key: Customer
    b. Type: string
    c. Label: Customer
    d. Max Length: 20
64. Scroll down in the Add New Filed modal to click Add
65. 

66. Next, we're going to apply the filter.
67. Click on the $filter row
68. 

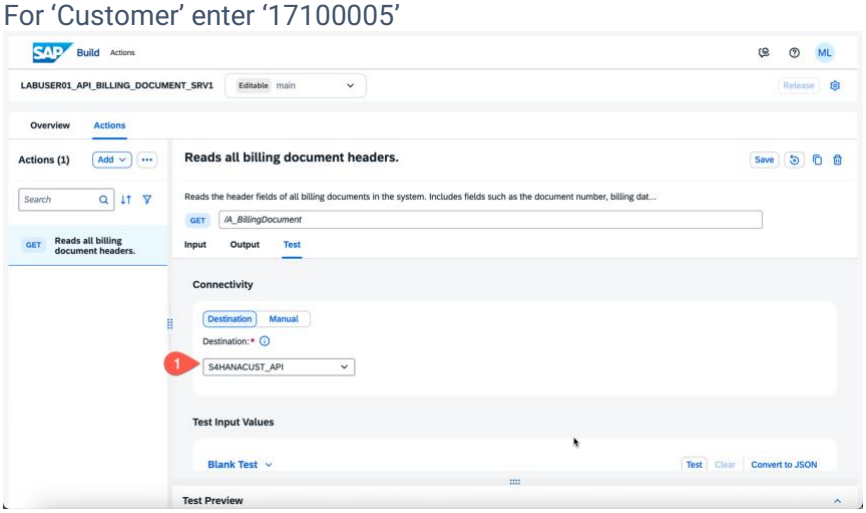
69. Scroll down and click the button in the Value: field

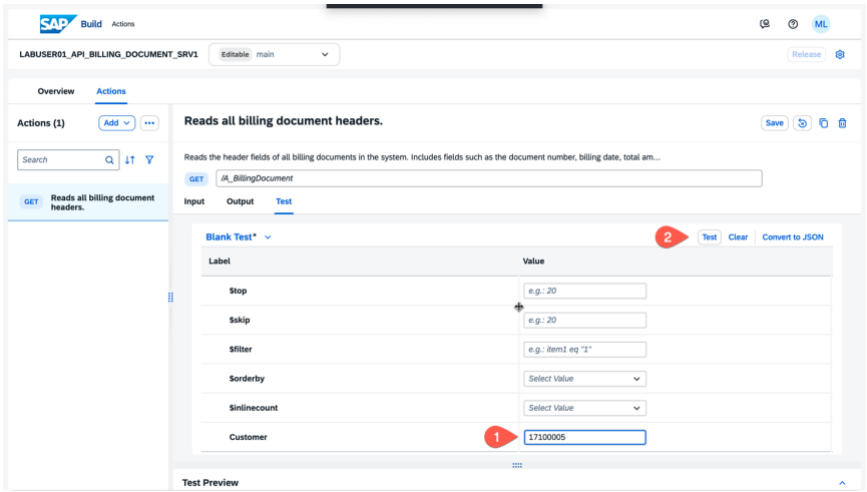
70. 

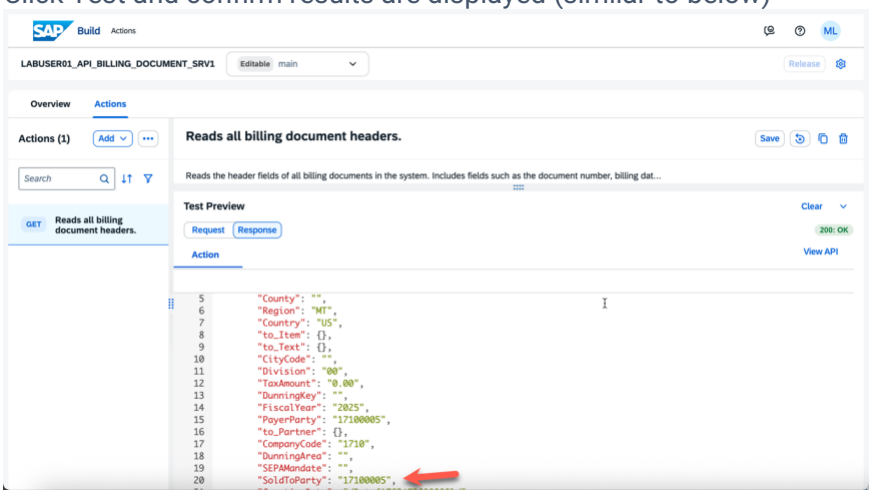
71. Add a Condition as shown below. Then click OK
    a. Column: SoldToParty
    b. Condition: equal to
    c. Criteria:${Customer}

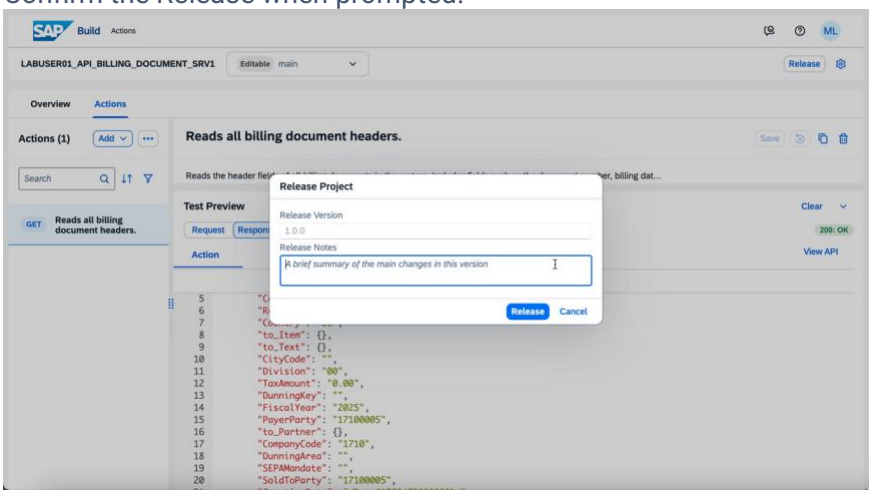
72. 

73. Close the $filter panel by click the ×
74. Select the *Test* tab
75. Select the Destination: S4HANACUST_API; scroll down to test values
76. For ‘Customer’ enter ‘17100005’

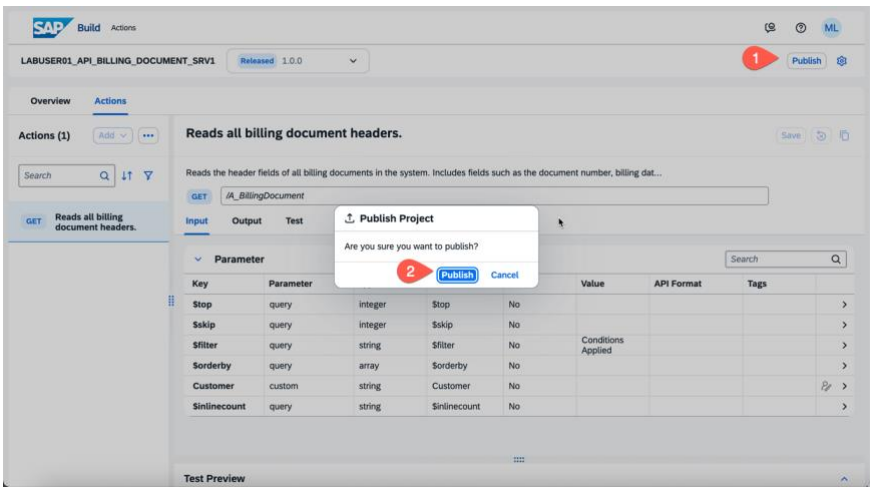
77. 

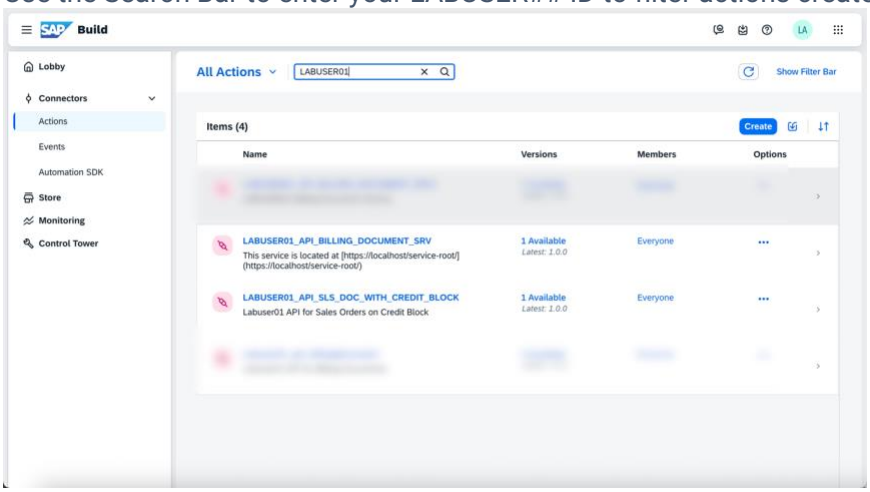
78. 

79. Click Test and confirm results are displayed (similar to below)
80. 

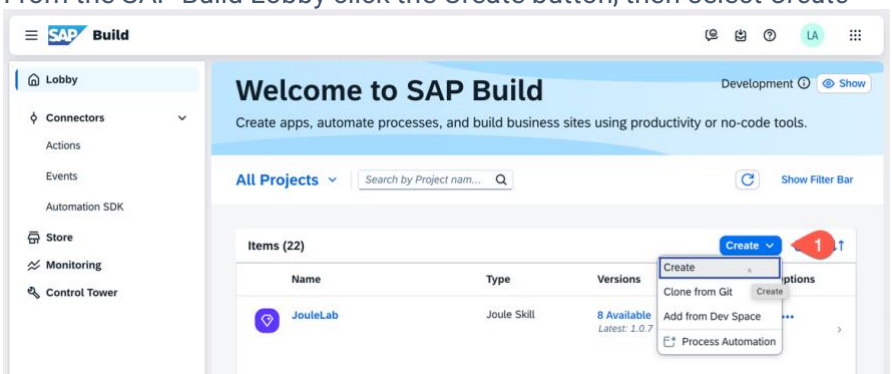
81. After confirm results, click the Save Button.
82. Click the Release button.
83. Confirm the Release when prompted.
84. 

85. Next, we're going to click Publish and then confirm.

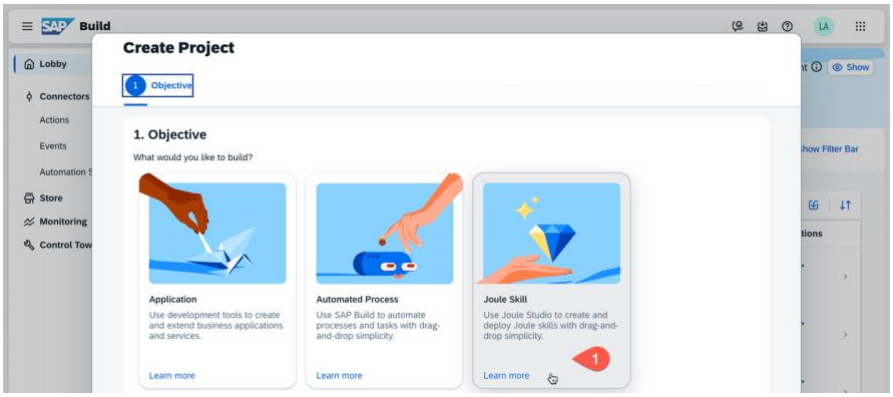
86. 

87. We've now publish two actions we can leverage within a Joule Skill.
88. Close the browser tab for the Action and return to the SAP Build Lobby
89. Use the Search Bar to enter your LABUSER## ID to filter actions created.
90. 

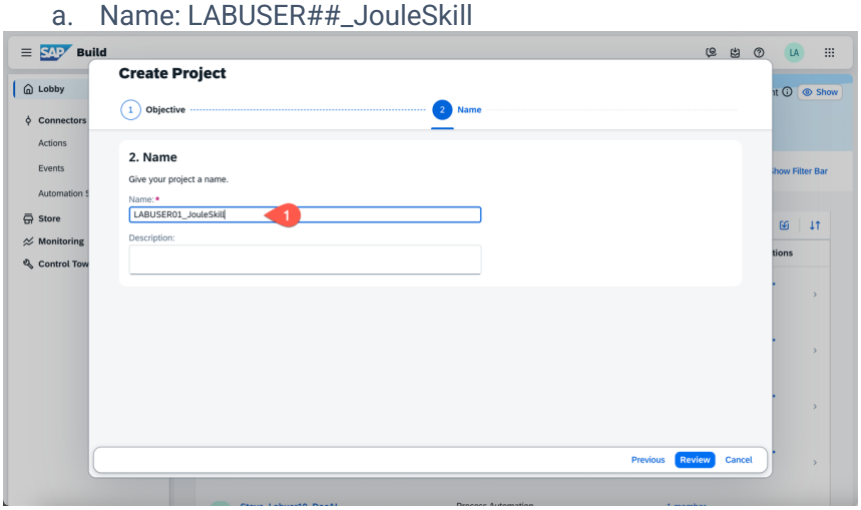
Create a Joule Skill

1. From the SAP Build Lobby click the Create button, then select *Create*
2. 

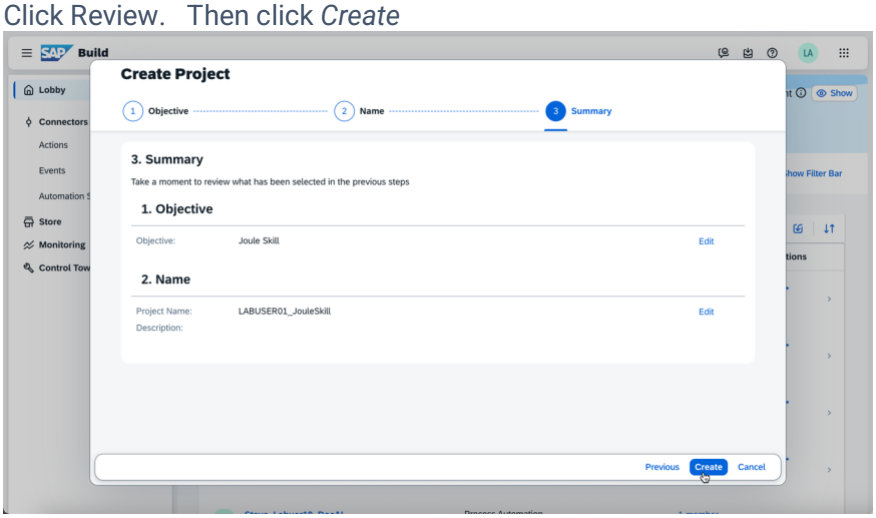
3. Select *Joule Skill*

4. 

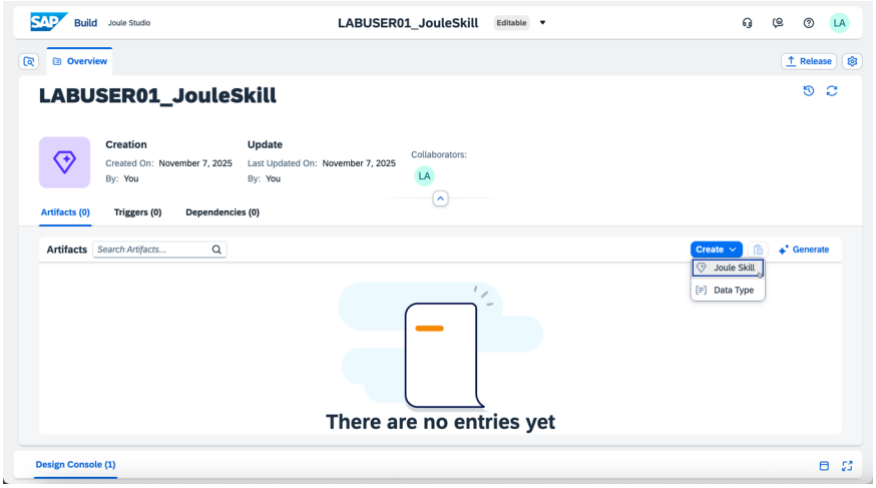
5. Enter a name for your Joule Skill in the following format
    a. Name: LABUSER##_JouleSkill

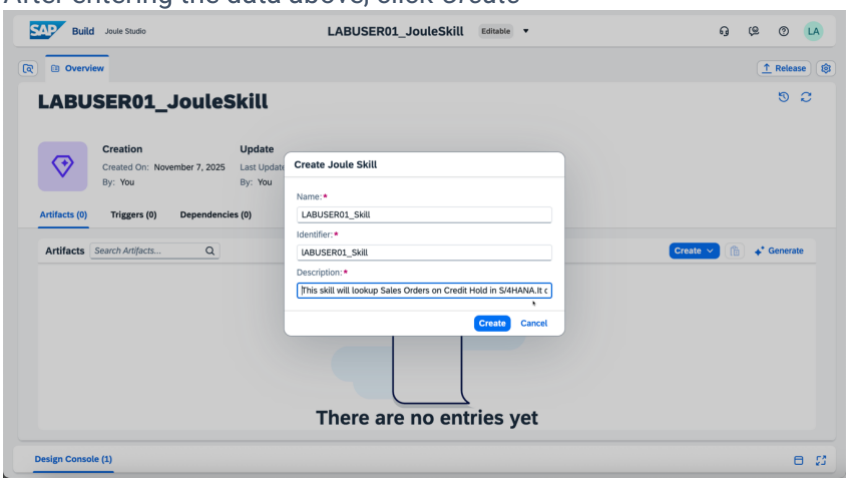
6. 

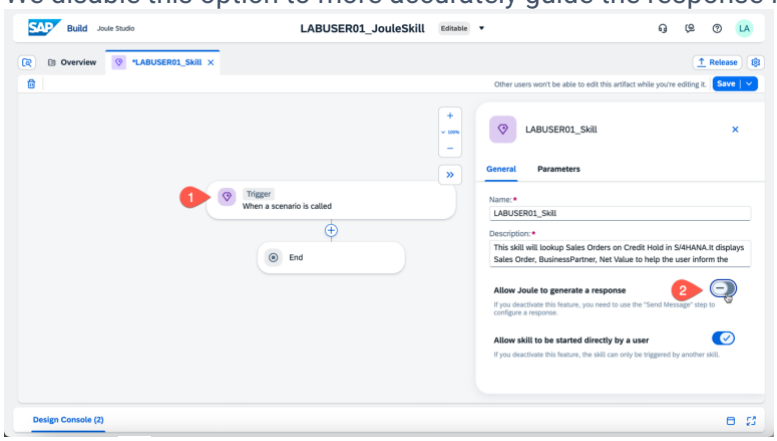
7. Click Review. Then click *Create*

8. 

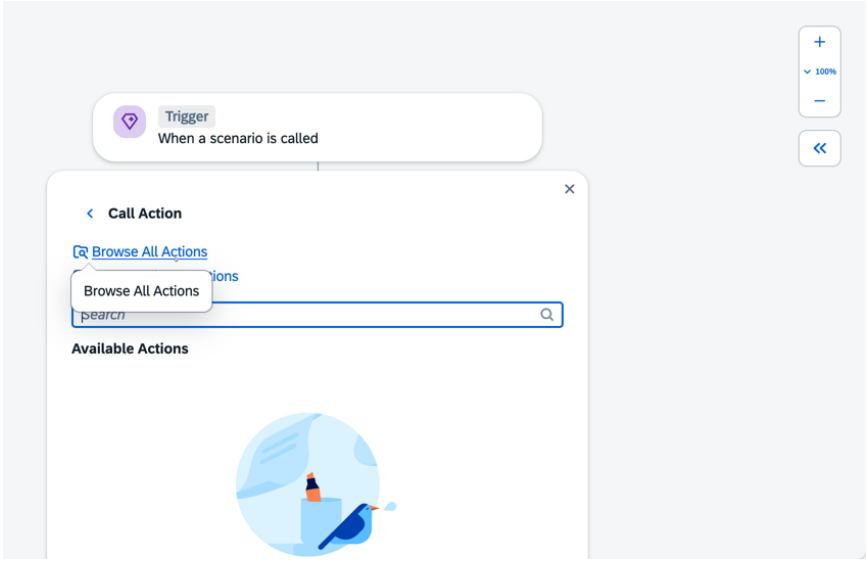
9. A new browser tab will open (suppress the pop-up blocker)
10. Click *Create > Joule Skill*

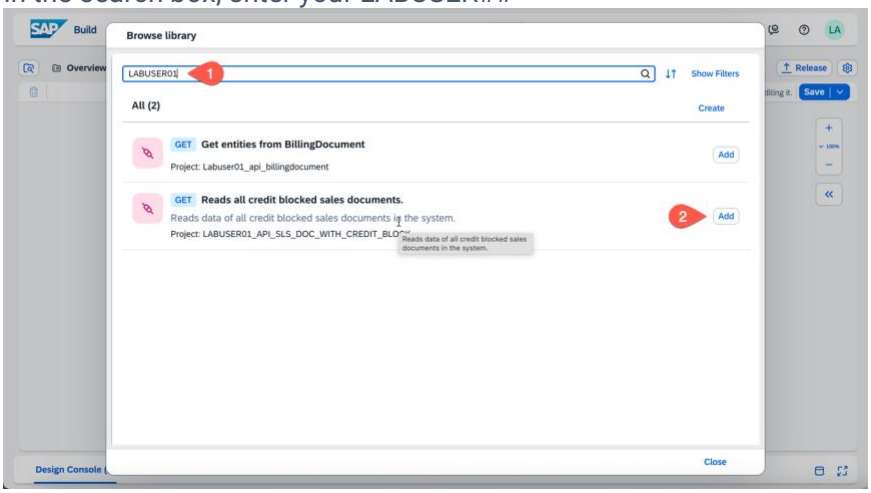
11. 

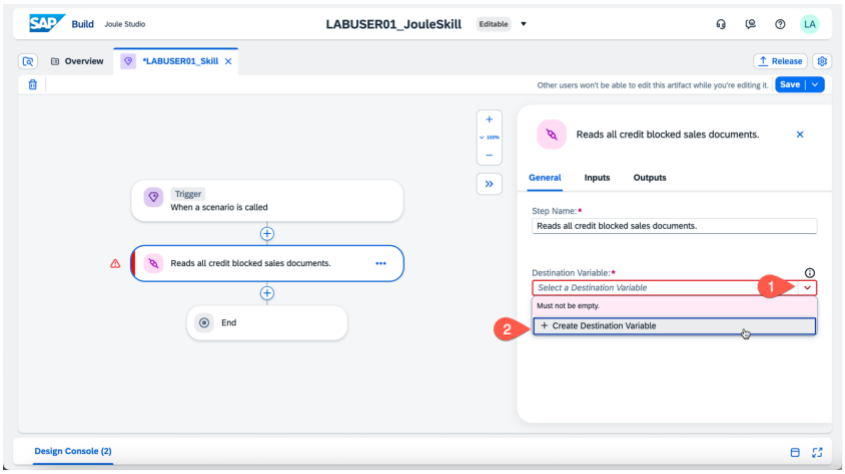
12. On the *Create Joule Skill* modal, enter the following data
    a. Name: LABUSER##_Skill
    b. Identifier: <Generated by SAP>
    c. Description: This skill will look up Sales Orders on Credit Hold in S/4HANA.It displays Sales Order, Business Partner, Net Value to help the user inform the customer of their order status.
13. IMPORTANT: The 'Description' is interpreted by the LLM to understand the task to be completed. The 'Description' is very important and should be informative.
14. After entering the data above, click *Create*
15. 

16. When the canvas opens, select the Trigger and disable "Allow Joule to Generate a Response"
    a. We disable this option to more accurately guide the response from Joule.
    b. 

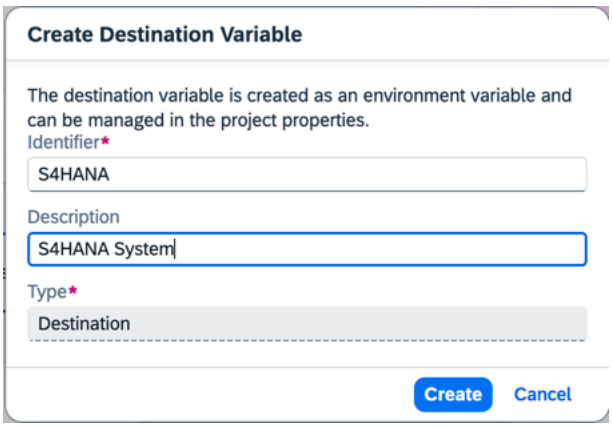
    c. Click the ✕ to close the details
17. Click the plus sign under trigger to add an Action
18. Click Action
19. Click Browse All Actions

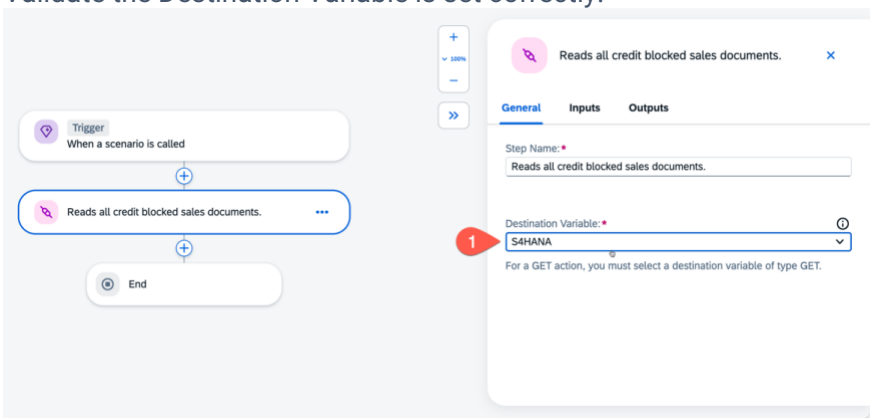
20. 

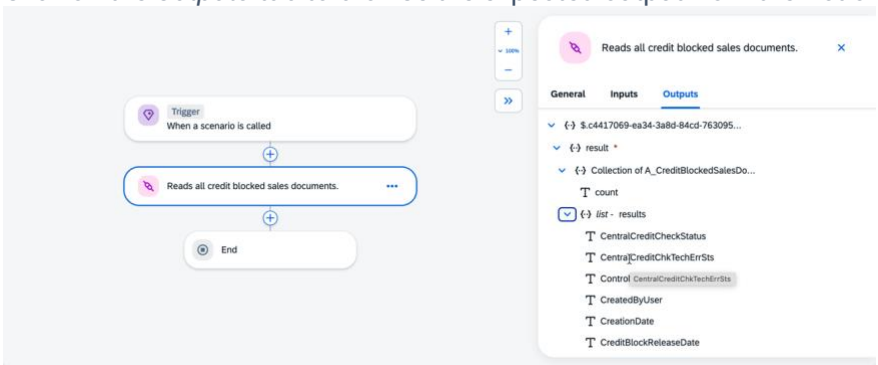
21. In the search box, enter your LABUSER##
22. 

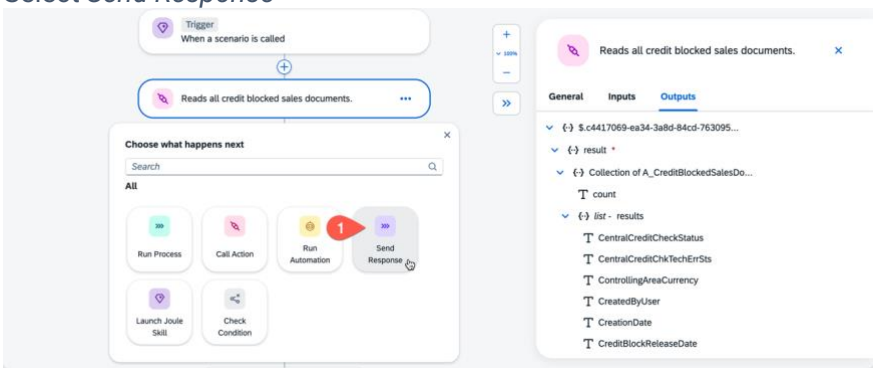
23. Click Add next to the Action ‘Reads all credit blocked documents’
24. After adding the Action, click the dropdown for *Select a Destination Variable*
25. Click *Create Destination Variable*
26. 

27. Enter the Identifier and Description as shown below. Click *Create*

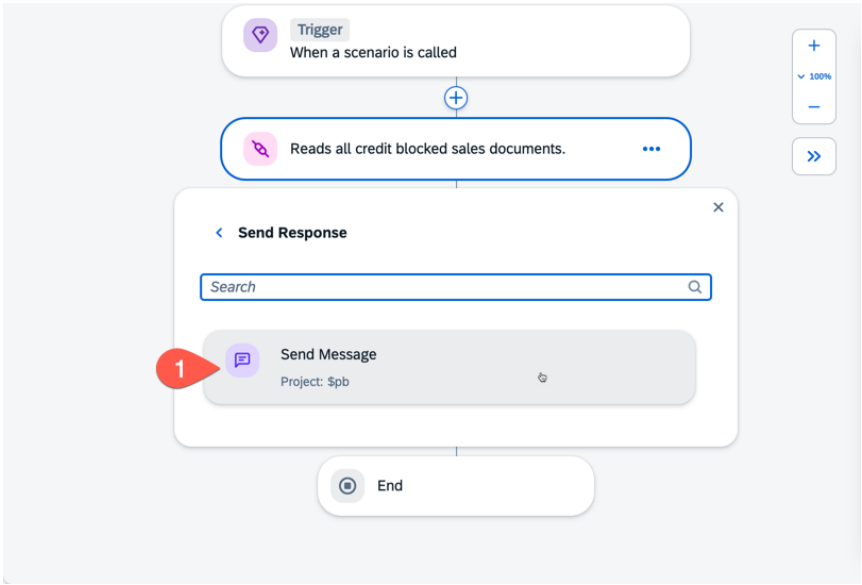
28. 

29. Validate the Destination Variable is set correctly.
30. 

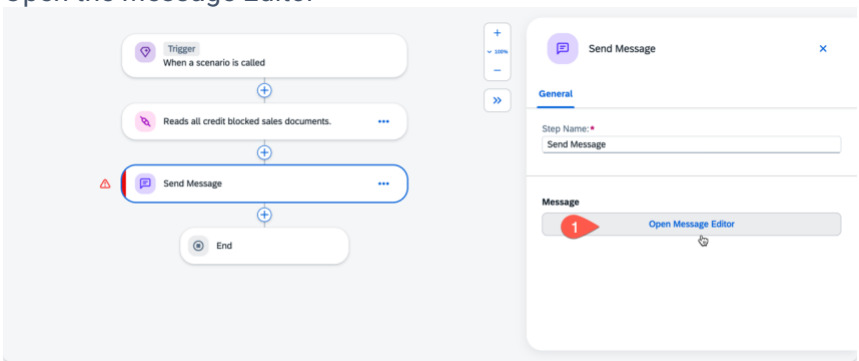
31. Click on the *Outputs* tab to browse the expected output from the Action
32. 

33. Return to the canvas and click the plus sign under the Action
34. Select *Send Response*
35. 

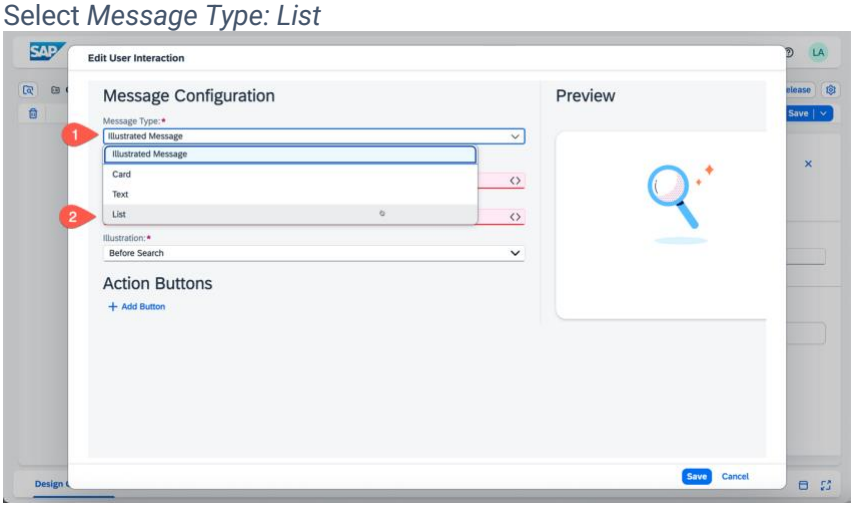
36. Select *Send Message*



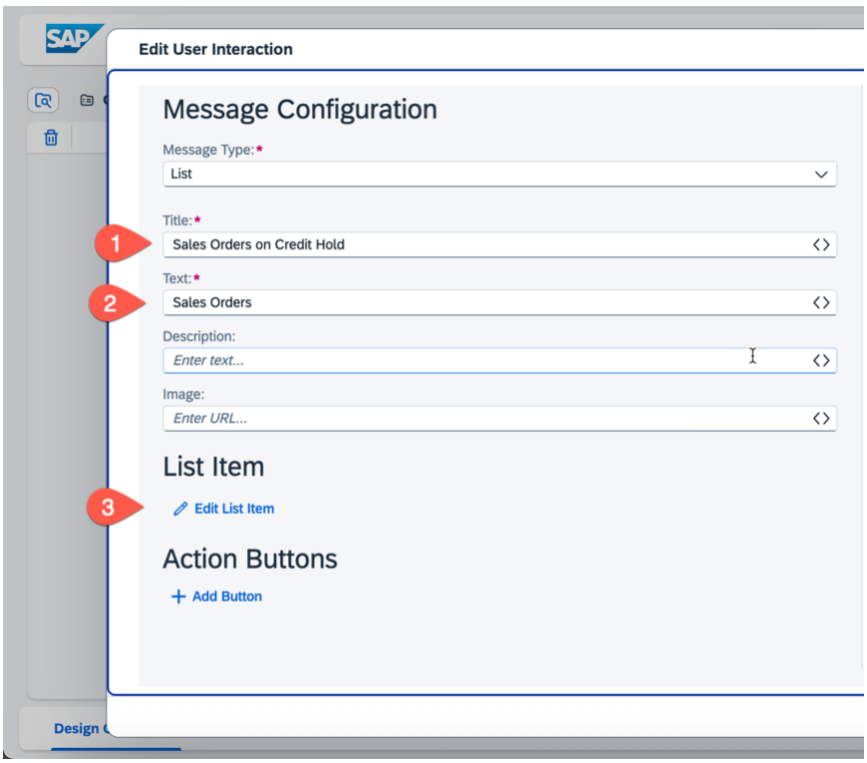
37.
38. Open the Message Editor

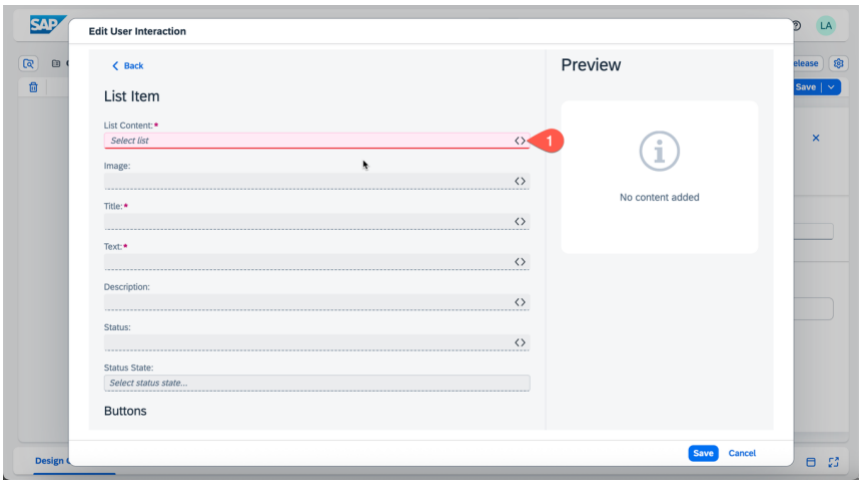


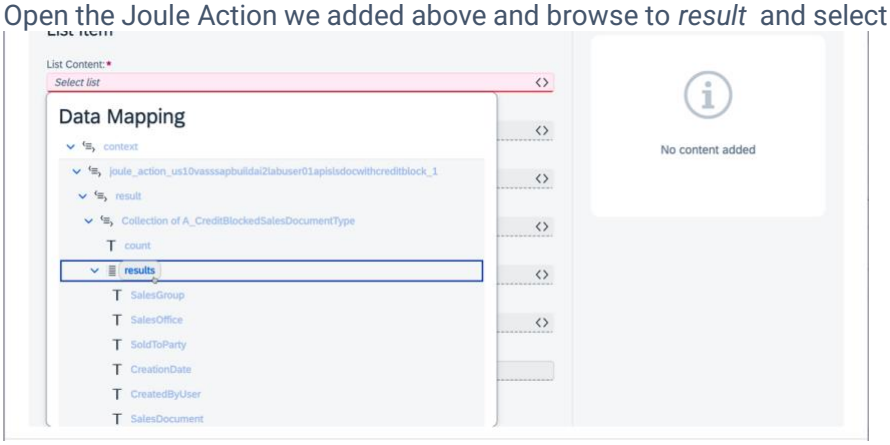
39.
40. Select *Message Type: List*



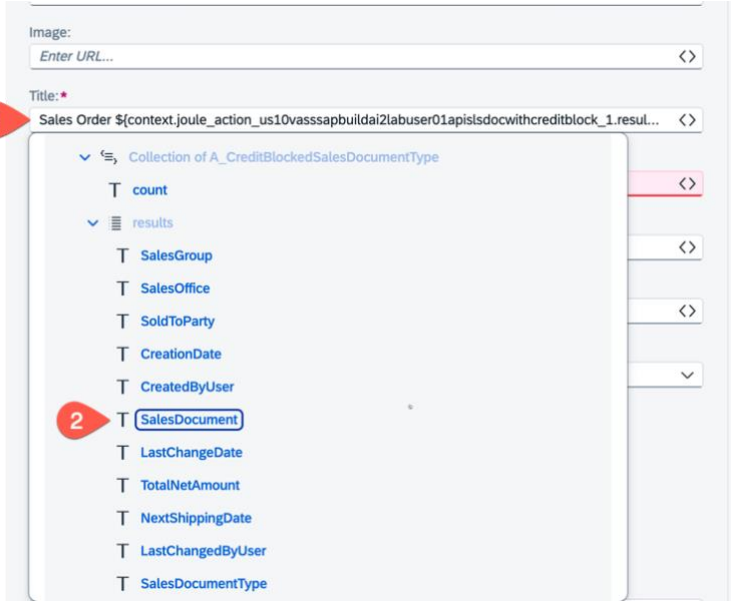
41.
42. Enter the Title: Sales Orders on Credit Hold

43. 

44. Select *Edit List Item*
45. On the *List Item* screen click the variable <> button for *List Content:*
46. 

47. Open the Joule Action we added above and browse to *result* and select it
48. 

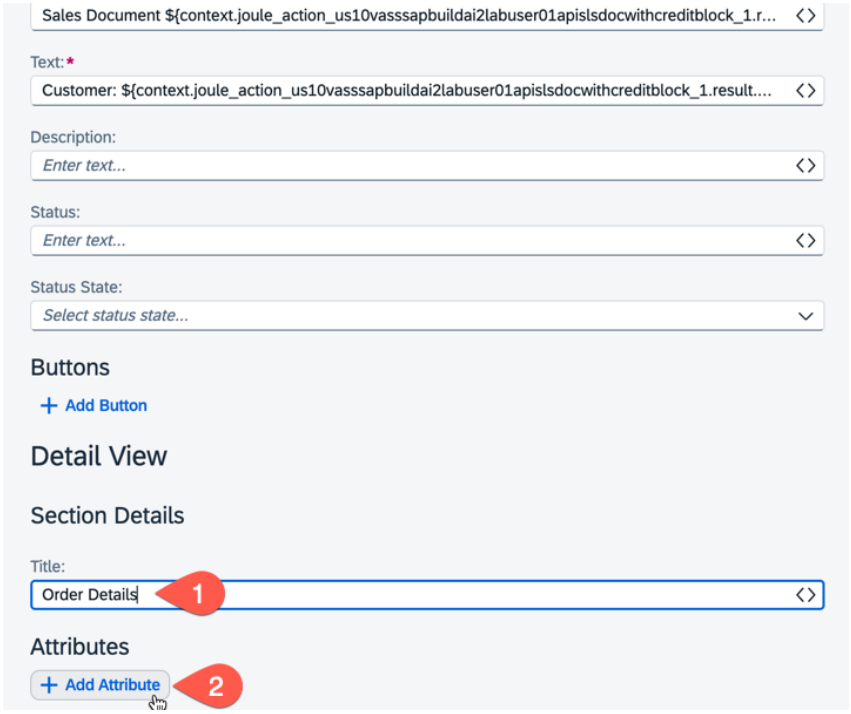
49. Enter the following data
    a. Title: **Sales Order** and open the variable results to SalesDocument

b. 

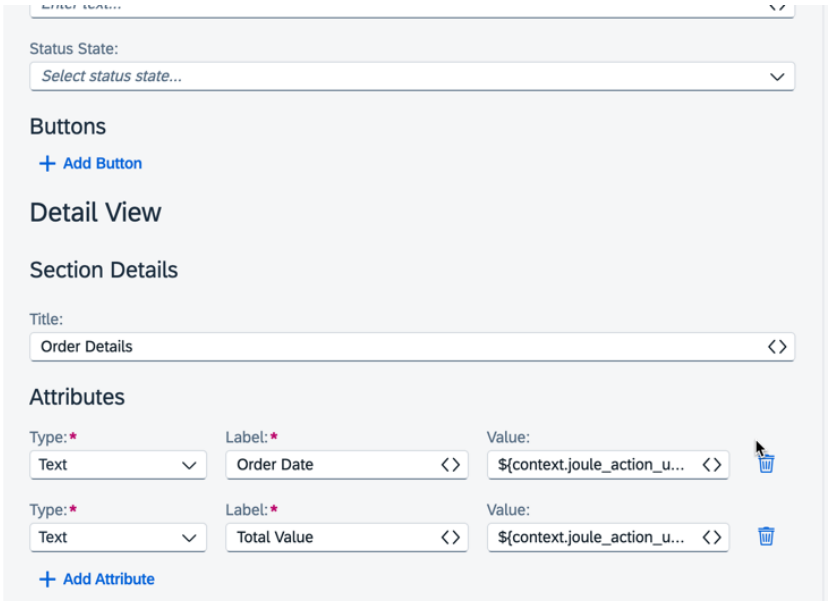
c. In the Text: field enter **Customer** and select the field SoldToParty

d. 

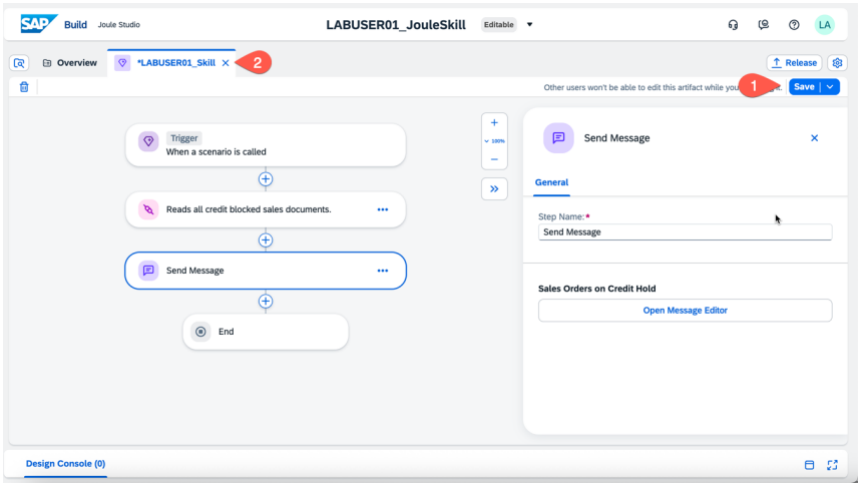
50. Scroll down to *Section Details*
51. Enter the Title: *Order Details* and click *Add Attribute*

52. 

53. Add two attributes
    a. Label: Order Date
        i. Value: Creation Date from Action T CreationDate
    b. Label: Total Value
        i. Value: TotalNetAmount from Action T TotalNetAmount

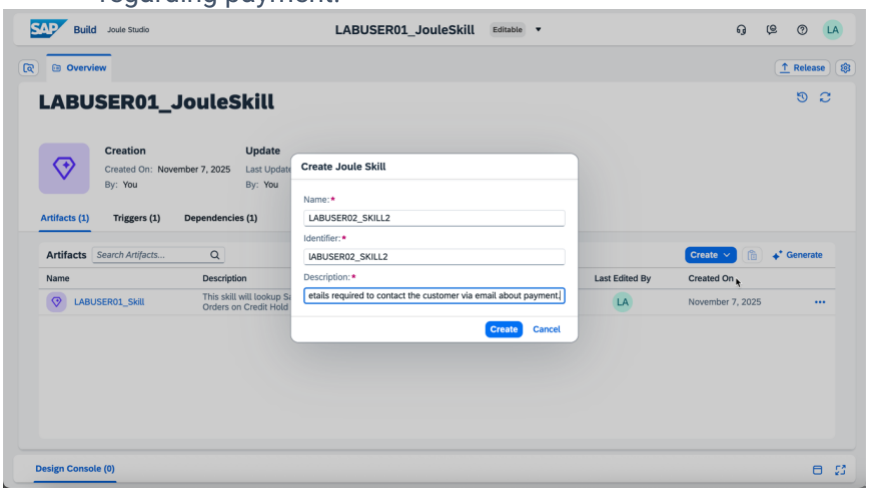


54.
55. Click the *Save* button when complete
56. Click the *Save* button again on the canvas.
57. Close the canvas for this skill.

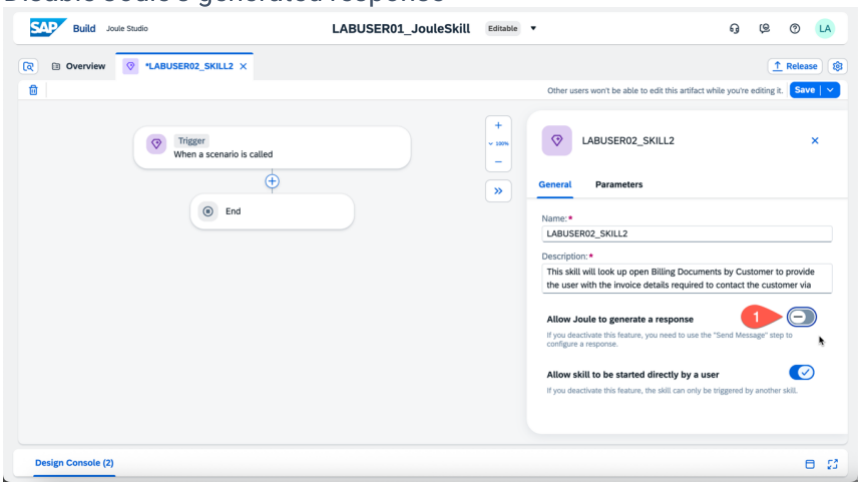
58. 

Next, we're going to create another skill to view open invoices. From the Overview, click *Create* > *Joule Skill* again

1. Enter the following details on Create Joule Skill
    a. Name: LABUSER##_SKILL2
    b. Description: This skill will look up open Billing Documents by Customer. Billing Documents are Invoices sent to the Customer awaiting payment. The output will provide the user with the invoice details required to contact the customer via email regarding payment.

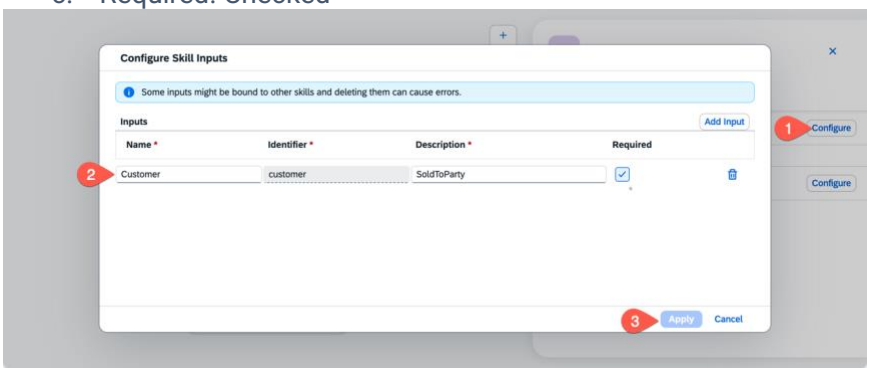
2. 

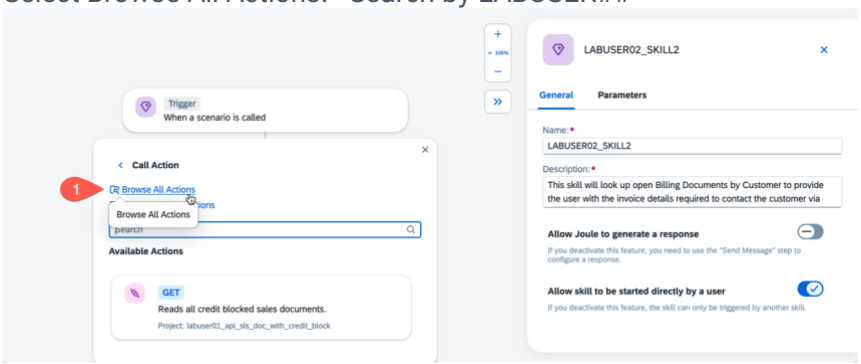
3. Disable Joule's generated response

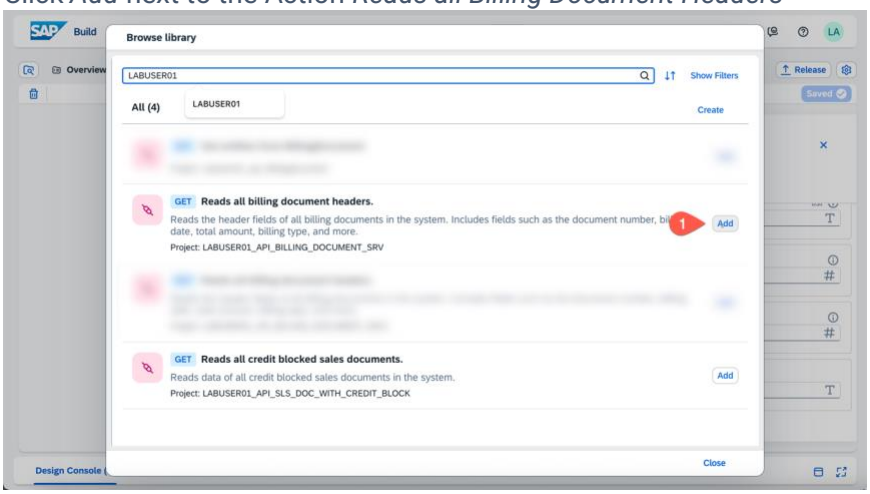
4. 

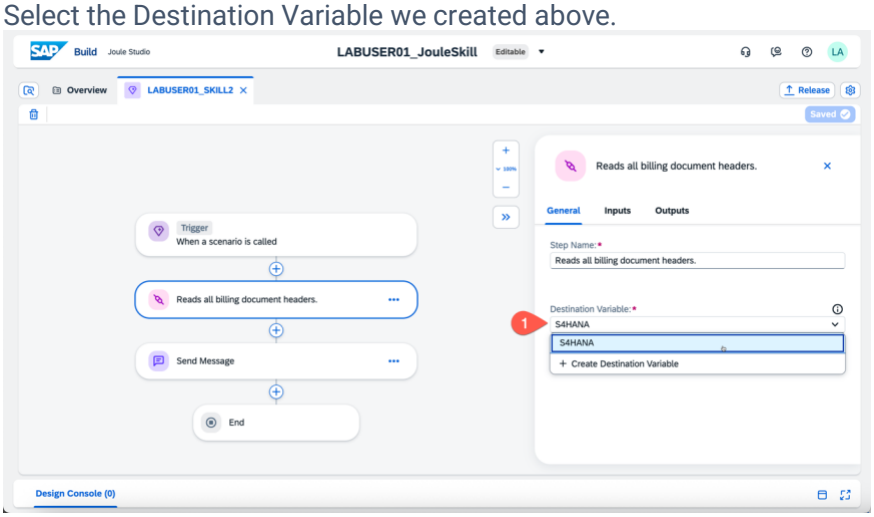
5. Click the Parameters Tab
6. Click Configure for Skill Inputs
7. Enter the following details

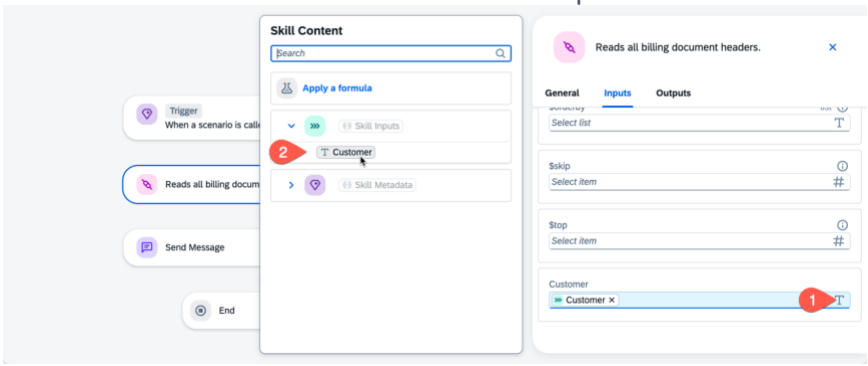
a. Name: Customer
b. Description: SoldToParty
c. Required: Checked

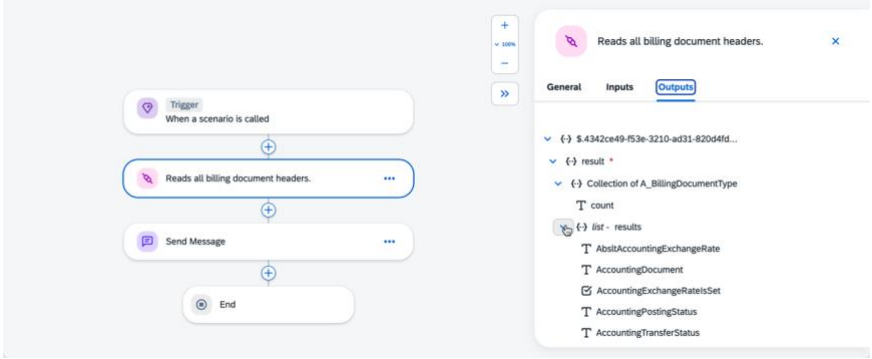
8. 

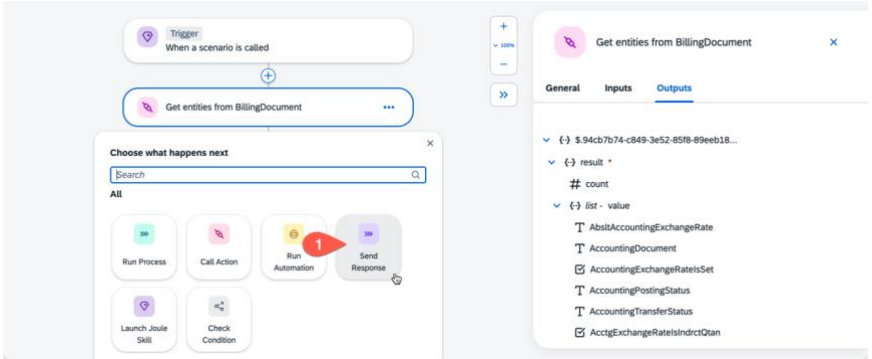
9. Click Apply
10. Return to the canvas, Click the plus sign to add a Call Action under Trigger
11. Select Browse All Actions. Search by LABUSER##
12. 

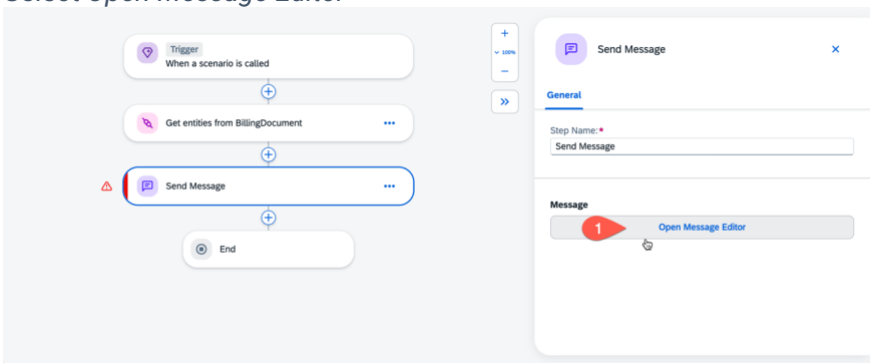
13. Click *Add* next to the Action *Reads all Billing Document Headers*
14. 

15. Select the Destination Variable we created above.
16. 

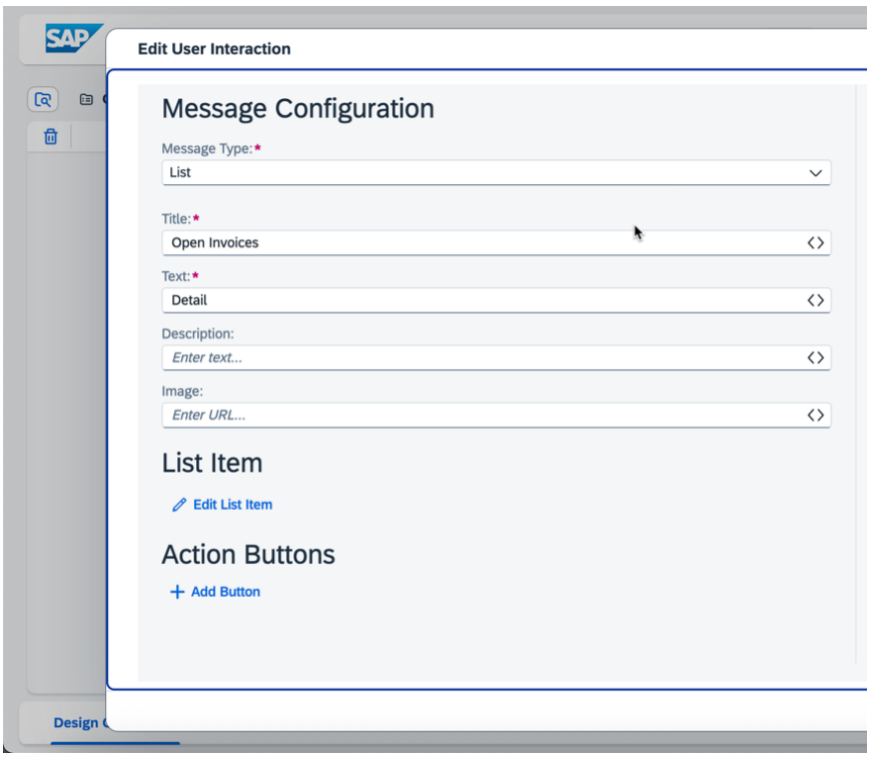
17. Next, we're going to add the Input Variable
18. Click the Inputs tab, scroll down to 'Customer' (the field we added to the API call)
19. Click into the field and select the 'Customer' input variable we added.
20. 

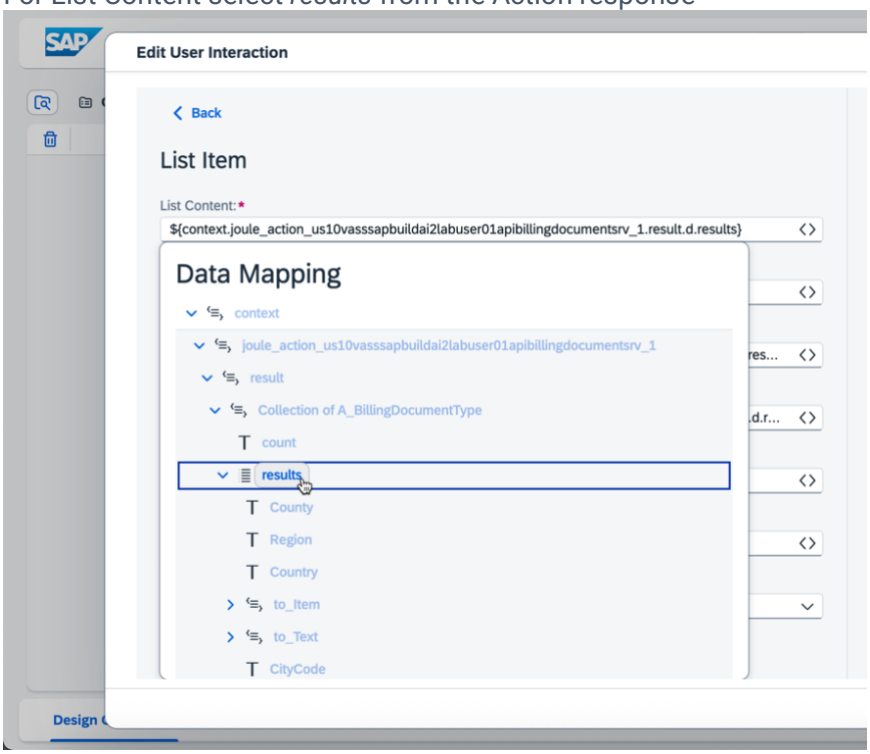
21. Browse the *Outputs* tab to review a potential response from the API
22. 

23. Click the plus sign under the Action we just added.
24. Select *Send Response*
25. 

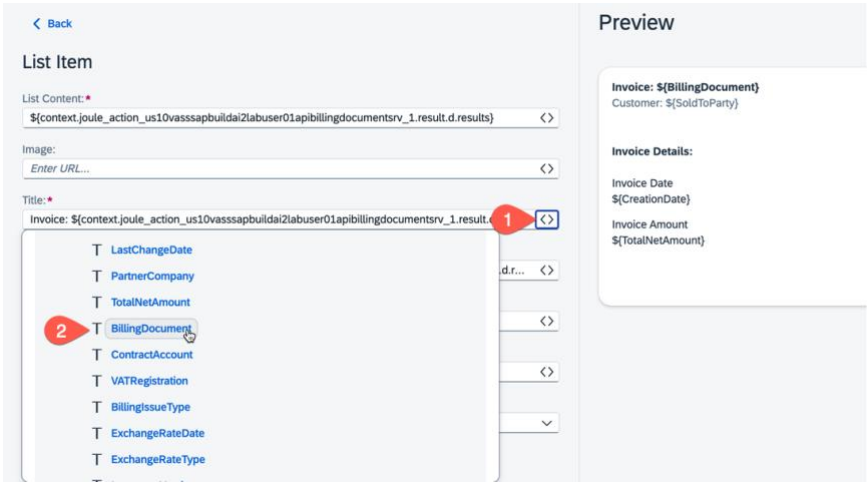
26. Select *Send Message*
27. Select *Open Message Editor*
28. 

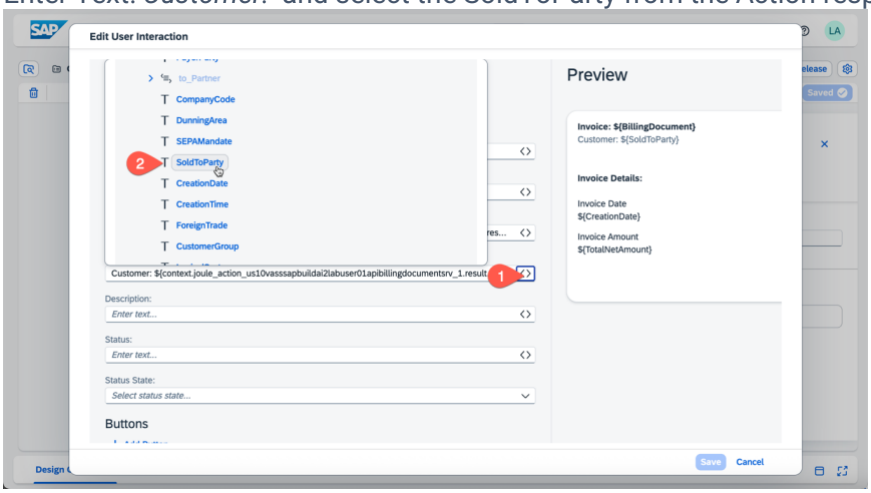
29. Select Message Type: *List*
30. Enter Title: *Open Invoices*
31. Enter Text: *Detail*

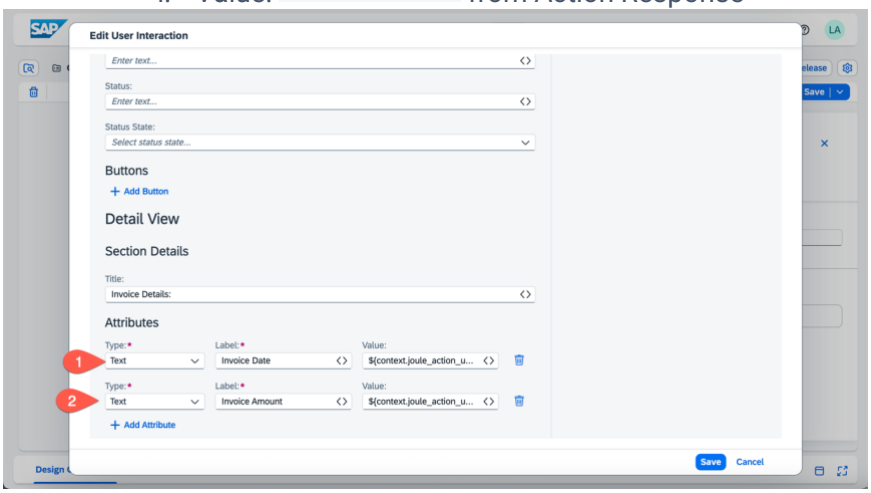
32. 

33. Click *Edit List Item* on screen
34. For List Content select *results* from the Action response
35. 

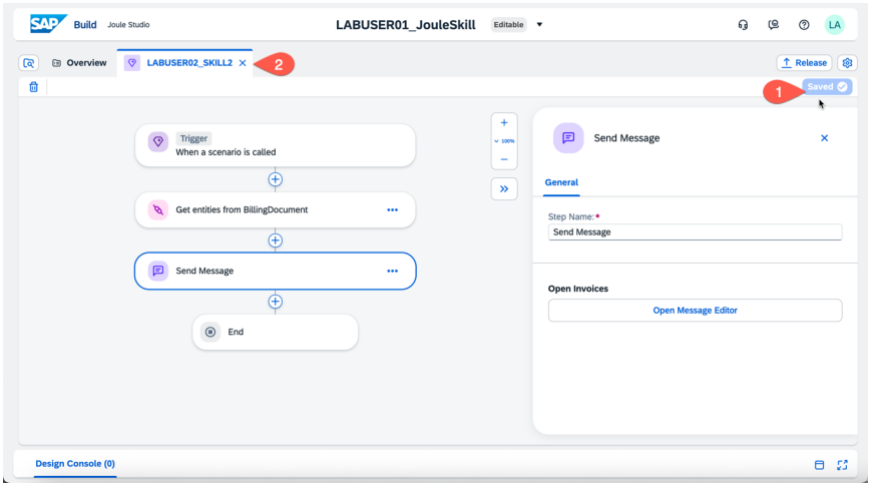
36. Enter Title: *Invoice:* and select the BillingDocument value from Action response.

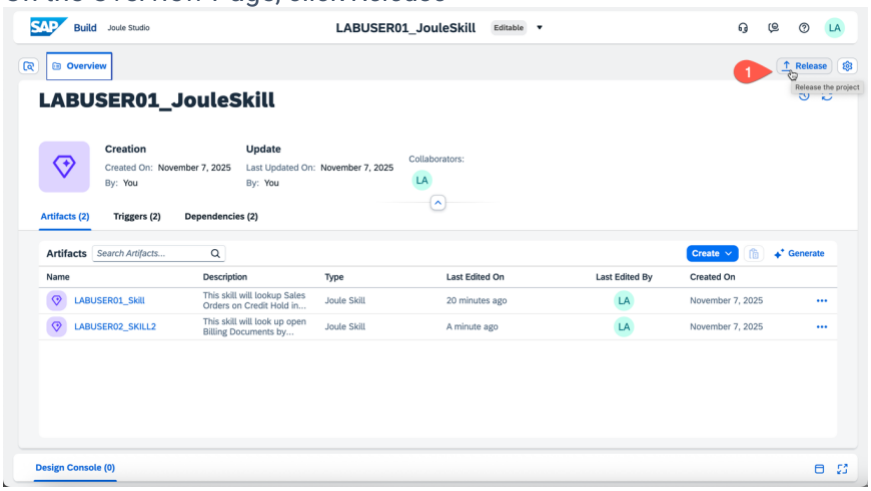
37. 

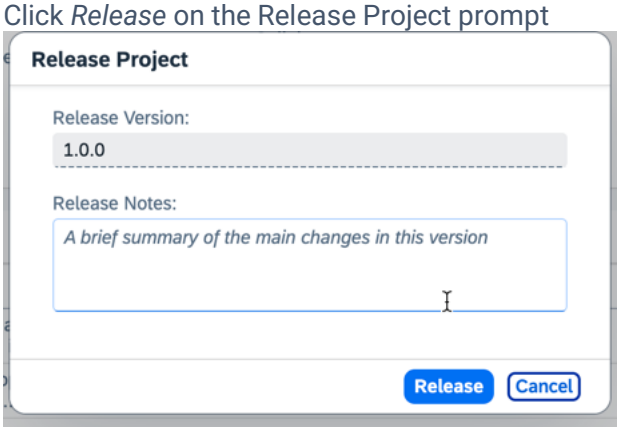
38. Enter Text: *Customer:* and select the SoldToParty from the Action response
39. 

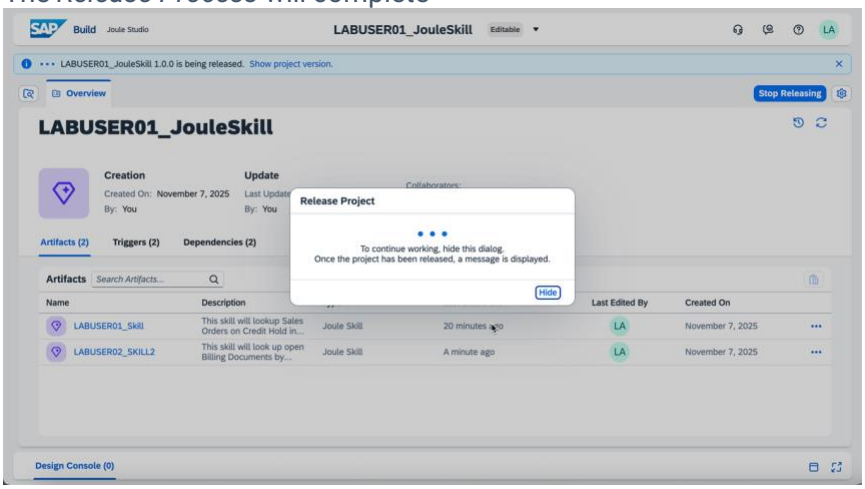
40. Scroll down to Detail View
41. Add two Attributes
    a. Label: Invoice Date
        i. Value: T CreationDate from Action Response
    b. Label: Invoice Amount
        i. Value: T TotalNetAmount from Action Response
42. 

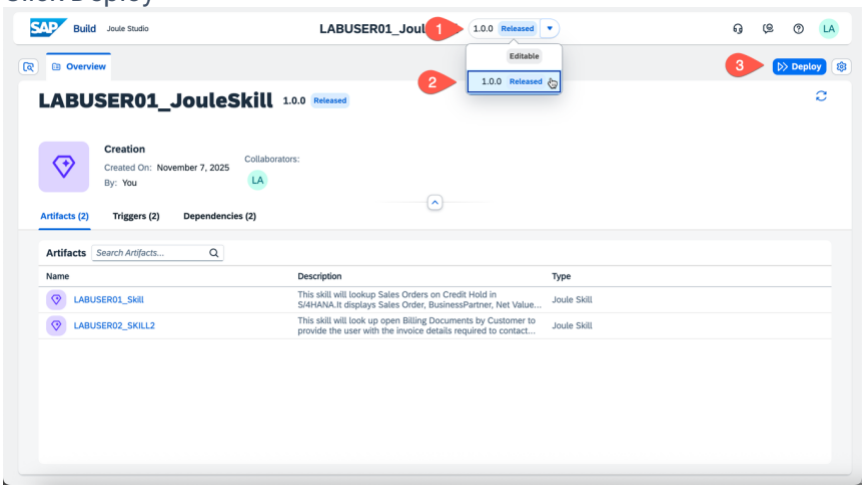
43. Click *Save*
44. Click *Save* again and close the Skill

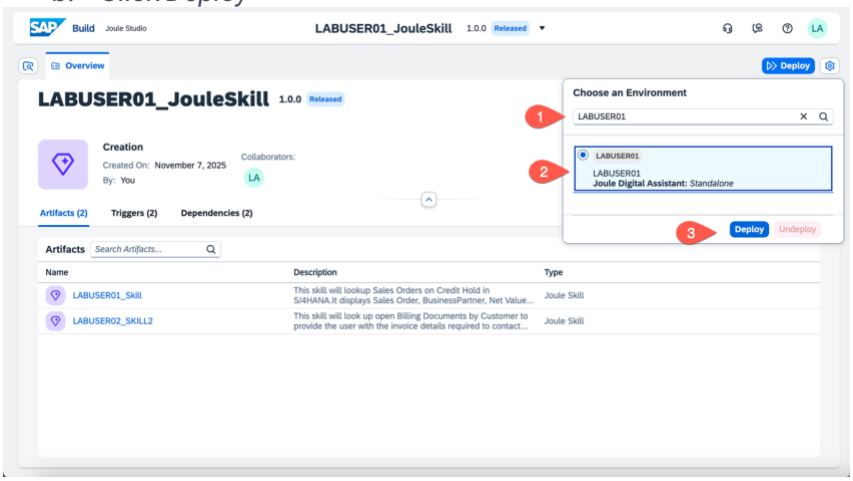
45. 

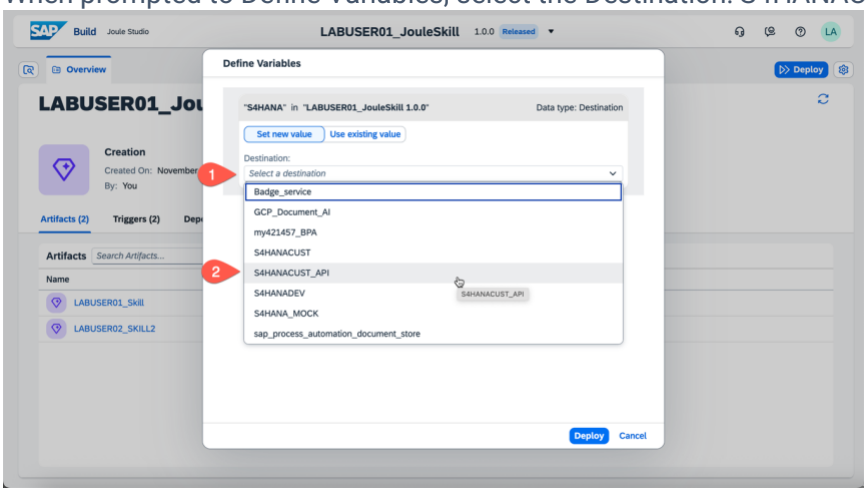
46. On the Overview Page, click *Release*
47. 

48. Click *Release* on the Release Project prompt
49. 

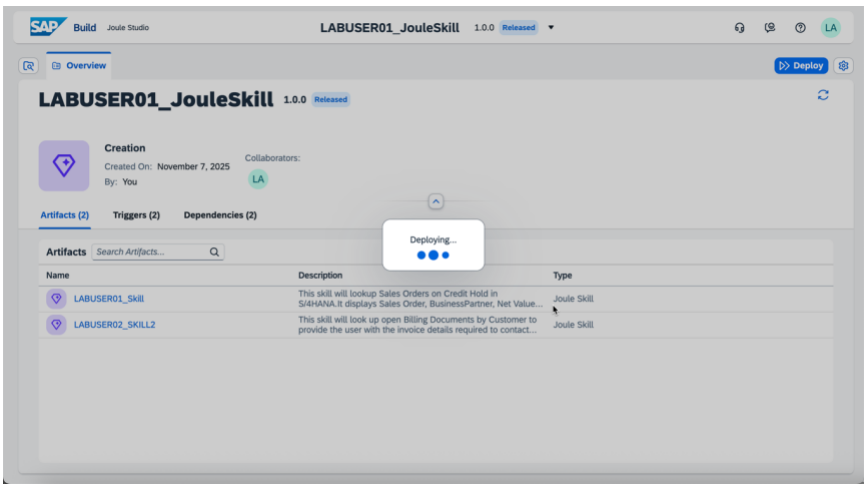
50. The *Release Process* will complete
51. 

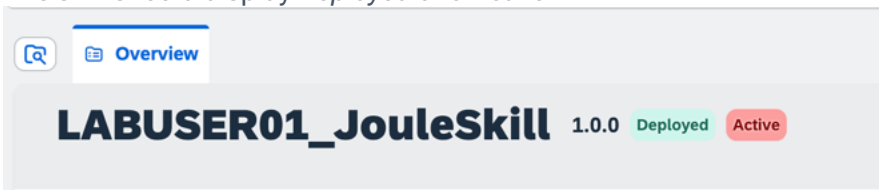
52. From the upper middle drop down, select your *Released* version
53. Click Deploy
54. 

55. Select the Environment for your LABUSER##
    a. *Search LABUSER##*
    b. Click *Deploy*
56. 

57. When prompted to Define Variables, select the Destination: S4HANACUST_API
58. 

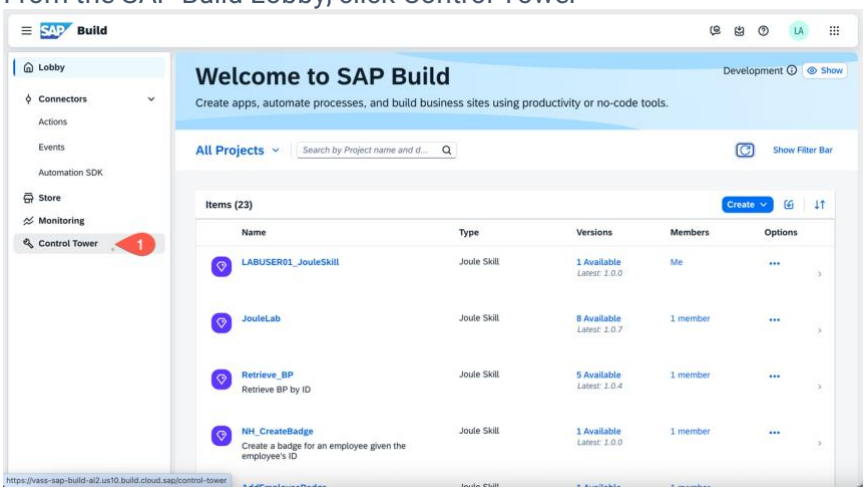
59. Click *Deploy* after selecting the destination

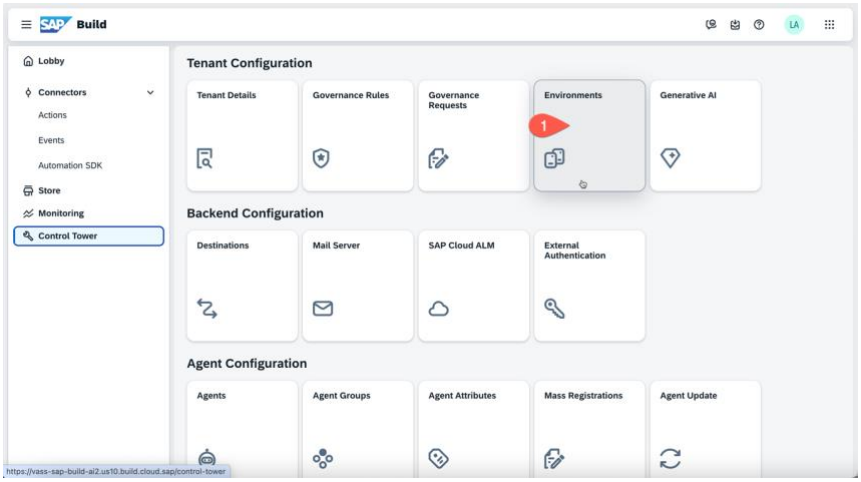
60. 

61. The skill should display *Deployed* and *Active*
62. 

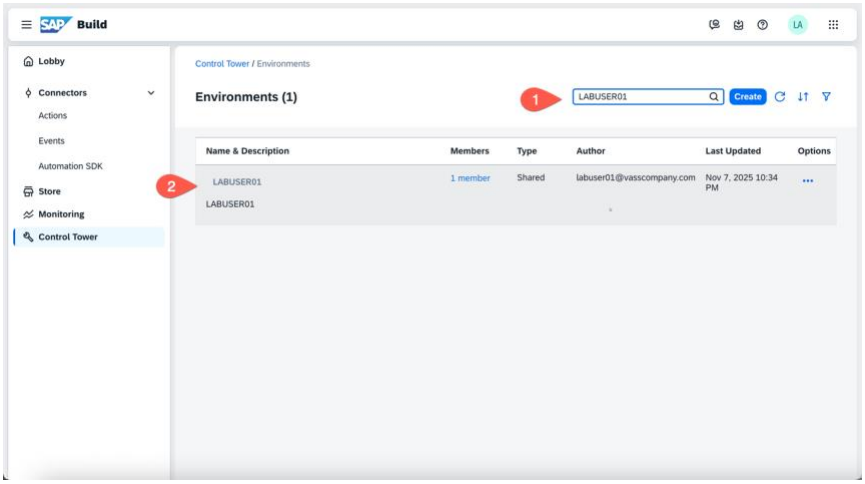
63. Close this Browser Tab and return to the SAP Build Lobby

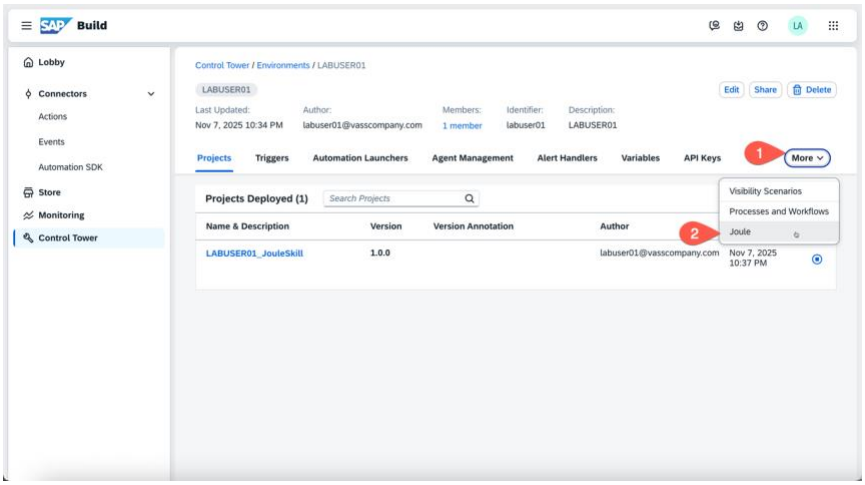
Test your Joule Skill

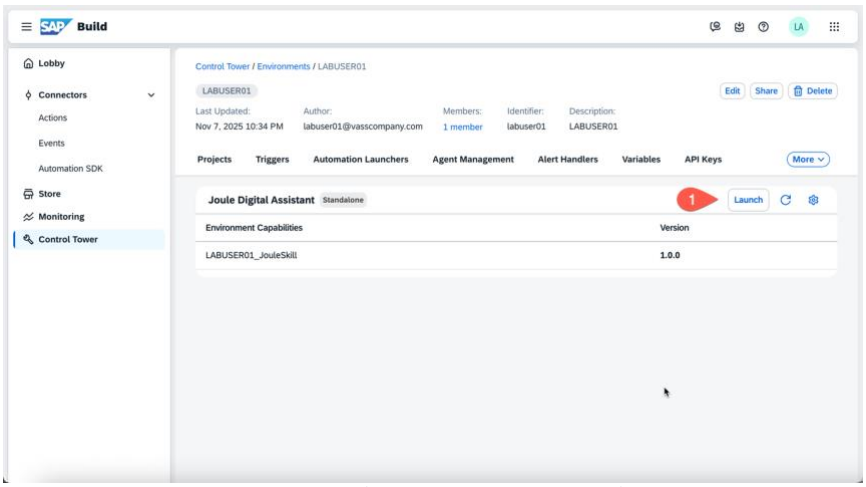
1. From the SAP Build Lobby, click Control Tower
2. 

3. Click on Environments
4. 

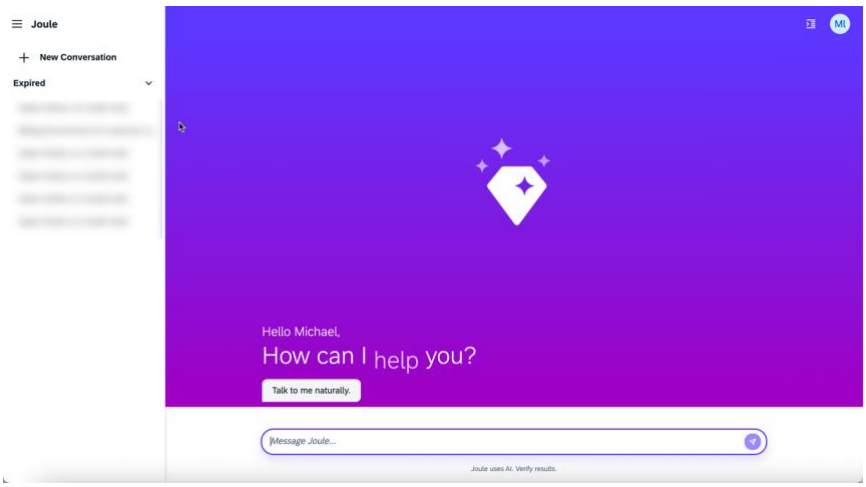
5. Search LABUSER## in the Search field

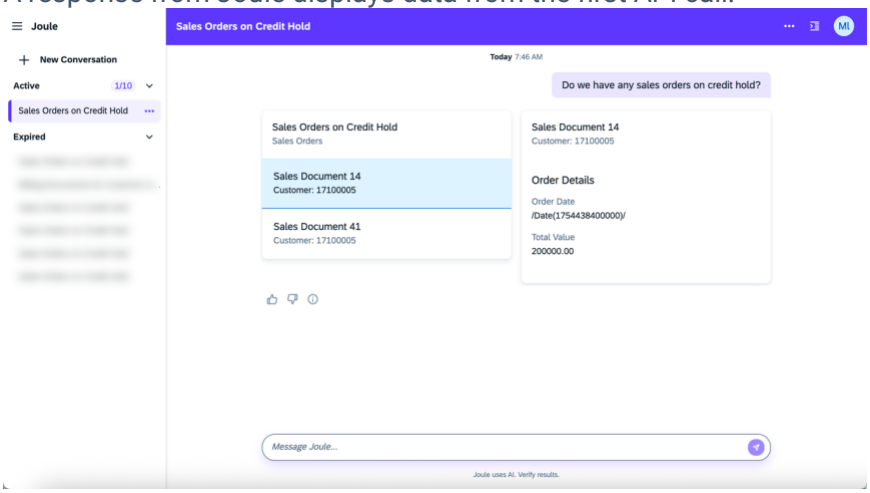
6. 

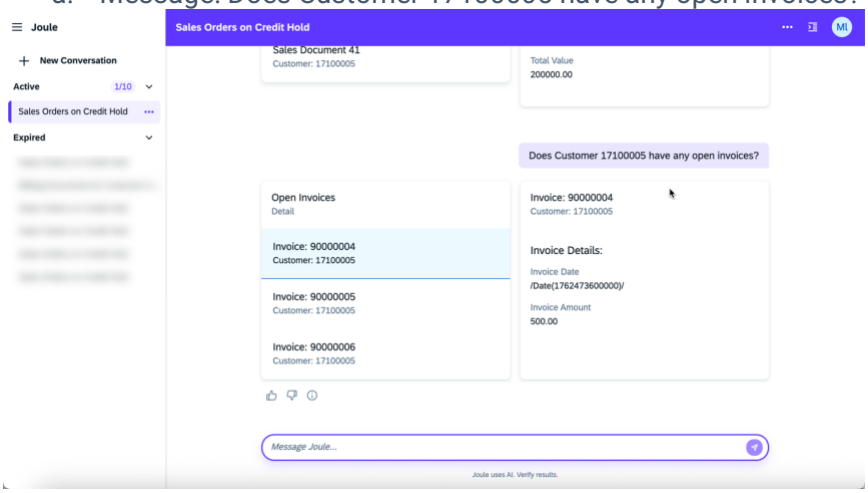
7. Select your Environment
8. Click *More > Joule*
9. 

10. Click *Launch*
11. 

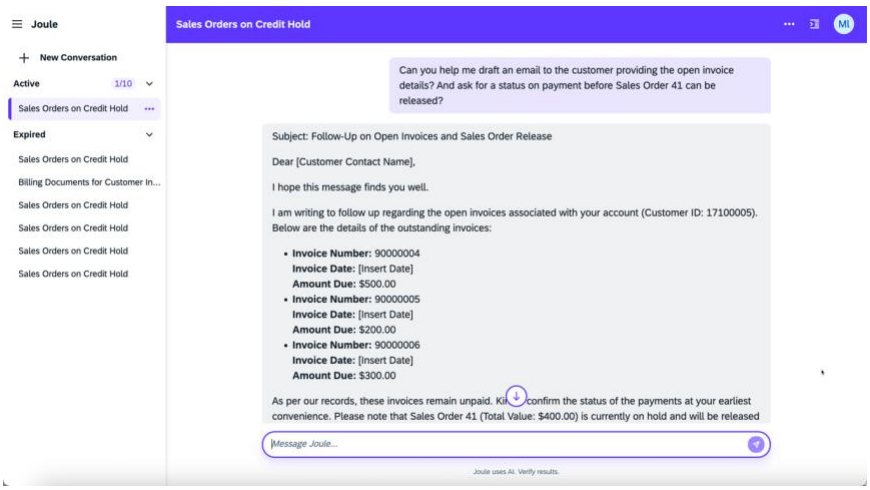
12. A new browser tab will open (verify popup blocker)
13. A new conversation will be started. You can also click New Conversation to start a new dialog.

14. 

15. Enter the following prompt for Joule
    a. Message: Do we have any sales orders on credit hold?
16. A response from Joule displays data from the first API call.
17. 

18. Note the Order Date field. It's the unmodified field from the JSON response
    a. Trivia: Where would you fix that field?
    b. Hint: It's in the Action configuration
19. Next, message Joule about any open invoices for the customer.
    a. Message: Does Customer 17100005 have any open invoices?
20. 

21. You can click on an invoice to view details from the API response.
22. Next, we're going to ask Joule to draft a email asking the customer for payment.
    a. Message: Can you help me draft an email to the customer providing the open invoice details? And ask for a status on payment before Sales Order 41 can be released?
23. Validate the response provided by Joule.

24. 

Where can we go from here?

1. Enrich the responses with more data, like Customer Name, Contact Name!
2. Resolve the date display issue in Joule response.
3. Link this Skill to an Automation in SAP Build to send emails!
4. For example, we could automate this skill to draft emails at the beginning of every week
5. Add buttons with links to S/4HANA Fiori Apps for Invoices or Sales Orders
6. Retrieve PDF copies of Invoices and attach to an email
7. And more!

Thank you for completing our Hands-On Lab! Please email if you have any questions!