



IBM App Connect Enterprise

ACE Designer for Beginners

Featuring:
ACE Designer, Jira, Slack

1. INTRODUCTION.....	3
2. CONFIGURE SLACK	4
2.1 LOGIN TO SLACK AND CREATE A WORKSPACE	4
2.2 GET SLACK ACCESS TOKEN.....	8
2.3 OBTAIN SLACK CLIENT ID, CLIENT SECRET AND AUTHORIZATION CODE.....	12
3. CONFIGURE JIRA	15
3.1 CREATE TWO ISSUES	16
3.2 CREATE JIRA API TOKEN.....	30
4. WRITE DATA FROM JIRA TO CONSOLE LOG.....	34
4.1 START DESIGNER FLOW EDITOR.....	34
4.2 CREATE EVENT-DRIVEN FLOW	38
4.3 CONFIGURE JIRA TRIGGER EVENT	39
4.4 CONFIGURE LOG ACTION	44
4.5 TESTING	49
5. SEND JIRA ISSUE TO SLACK.....	52
5.1 TEST SENDING JIRA DATA TO SLACK.....	52
END OF LAB GUIDE	62

1. Introduction

The main objective of this hands-on lab is to show the new Designer Flow capability in the (software based) Designer Flow Editor in v13.

In this lab guide you will create a Designer Event-driven message flow to process Issues from Jira when they are created or updated and send information about them to a channel in Slack. You will require access to Jira Cloud and Slack. For the purposes of this guide, we registered a Gmail account and used this to acquire access to Jira Cloud and Slack.

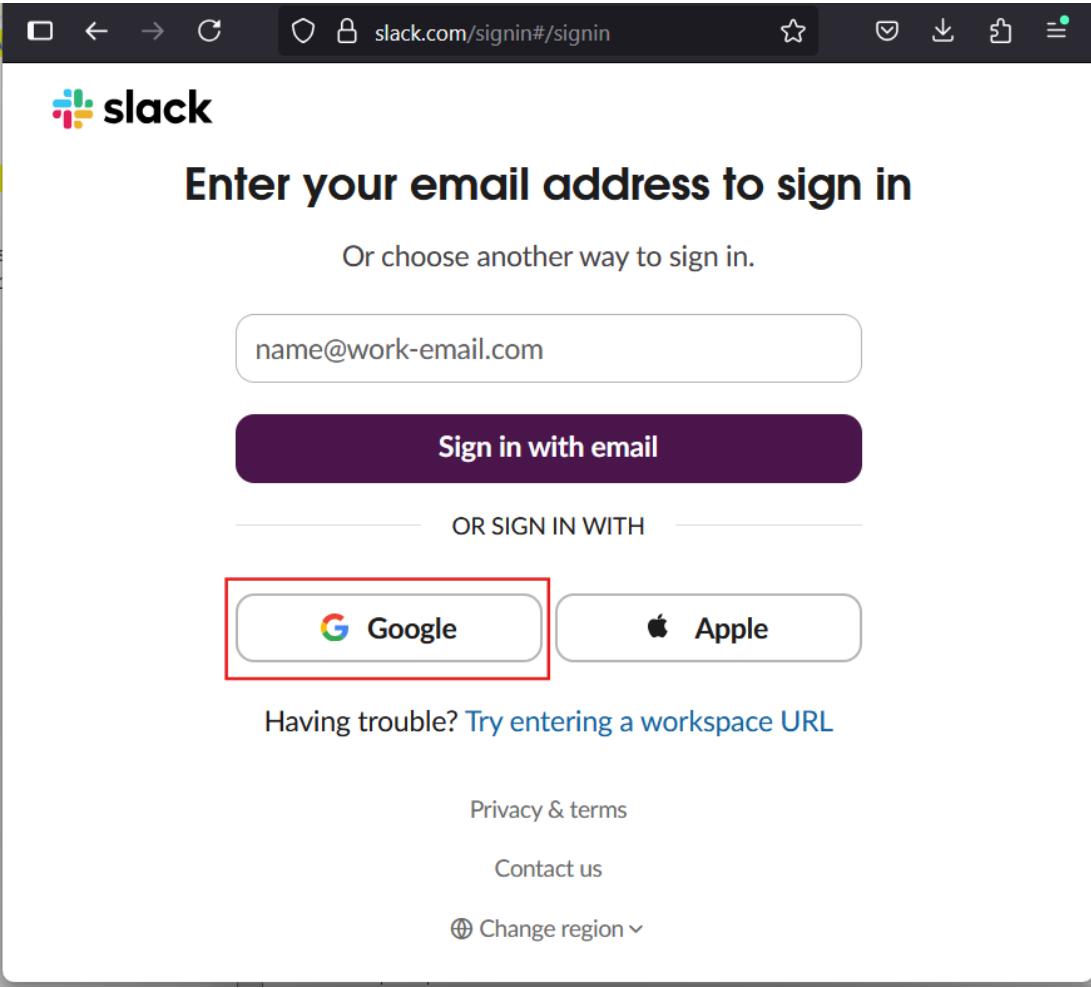
This lab assumes that you have at your disposal:

- A Windows VM Image which has ACE 13.0.3.0 installed.

2. Configure Slack

If you are already in possession of a Slack Access Token, perhaps having completed another ACE Early eXperience Program Lab, then you won't need to complete this section. If you do not have a Slack Access Token, then follow the instructions in this section of the Lab guide to obtain one for your Slack user. In this section you will register to a "free" version of slack and configure your environment so that ACE can communicate with it. **Note**, if you have access to the eXp workshop Slack channel, you will NOT be able to use this Slack environment for the purpose of this lab as IBM has restrictions on the apps that guest users are able to create.

2.1 Login to Slack and create a workspace

1.	Open a web browser and go to the URL: https://slack.com/signin#/signin
2.	For the purpose of this Lab guide, we used a Gmail account to register for Slack: 

3. Enter your Gmail account and click **Next**:

Sign in

to continue to [Slack](#)

Email or phone
benthompson[REDACTED]com

Forgot email?

Before using this app, you can review Slack's [privacy policy](#) and [terms of service](#).

Create account **Next**

English (United States) ▾ Help Privacy Terms

4. Enter your password and click **Next**:

Sign in with Google

Hi Ben

benthompson442@gmail.com

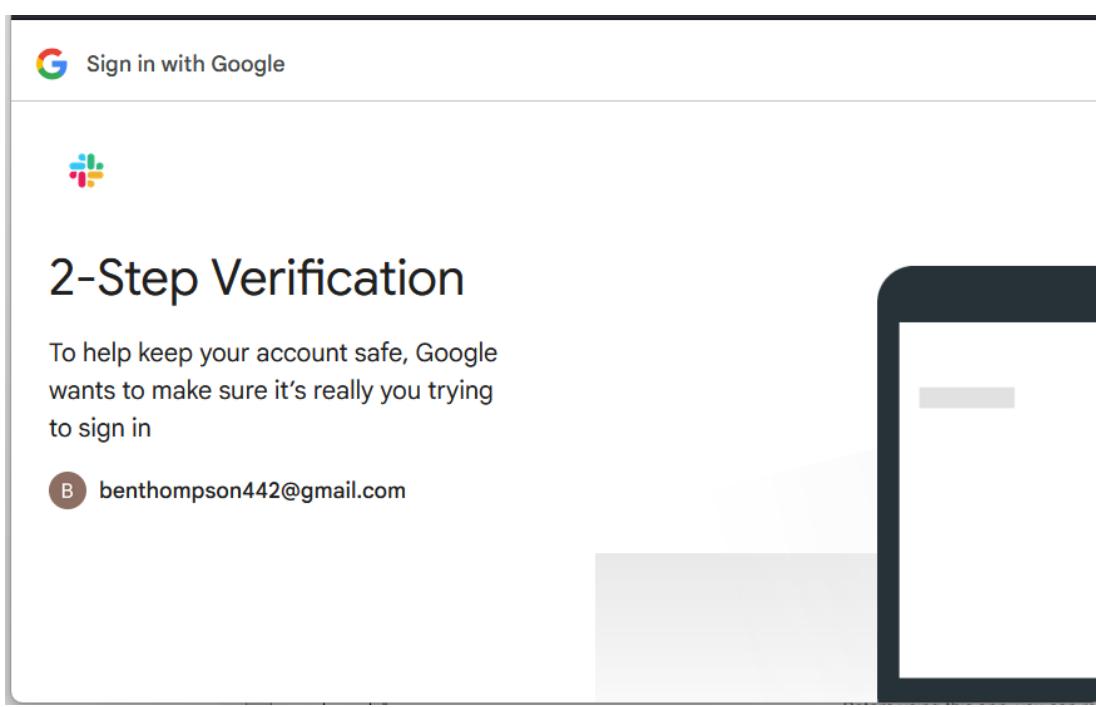
Enter your password
[REDACTED]

Show password

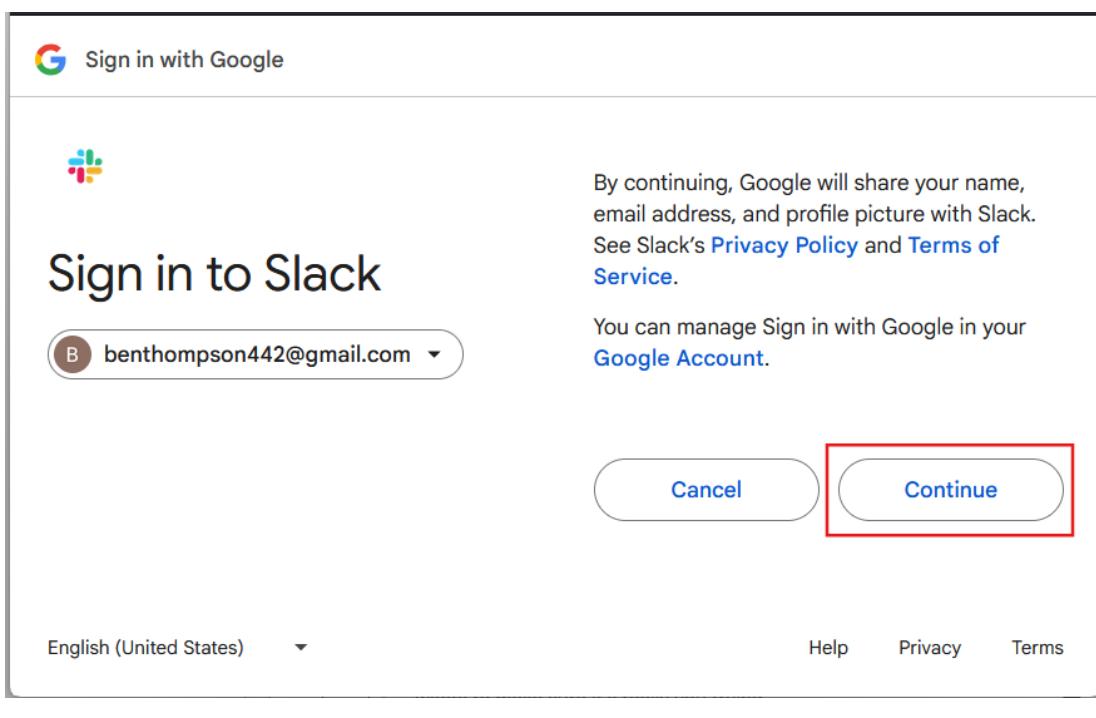
Before using this app, you can review Slack's [privacy policy](#) and [terms of service](#).

Try another way **Next**

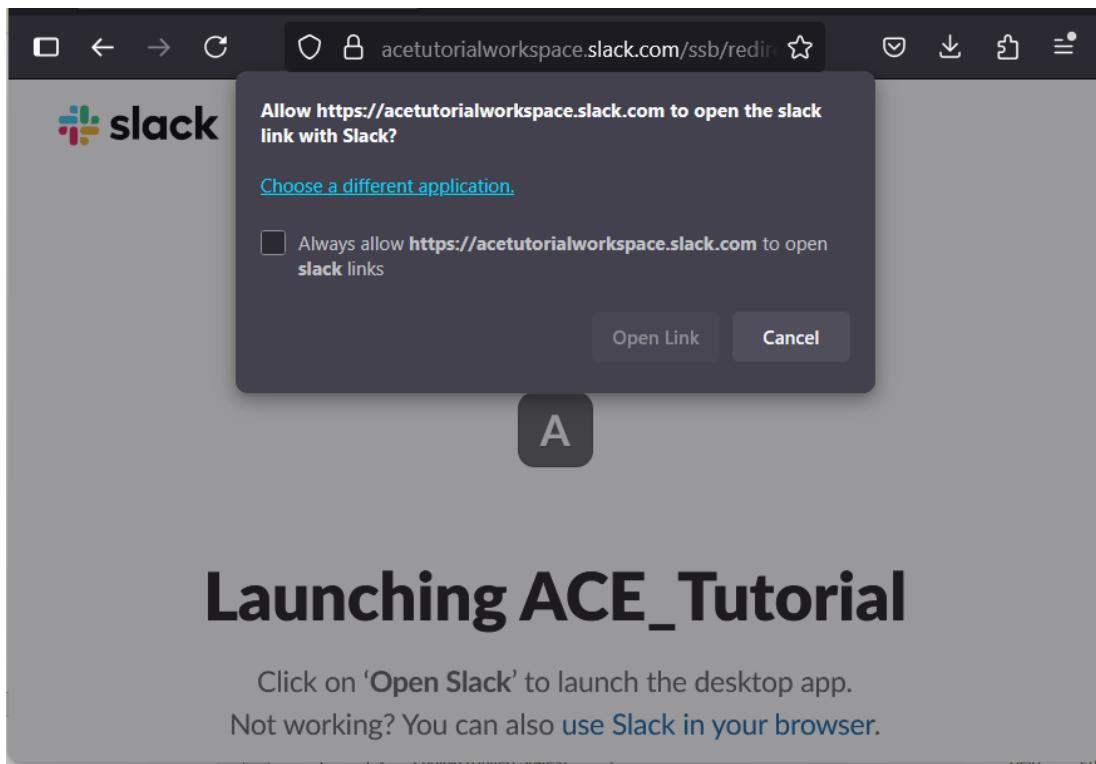
5. Depending on your account set up, you may be asked to complete a 2-Step Verification.



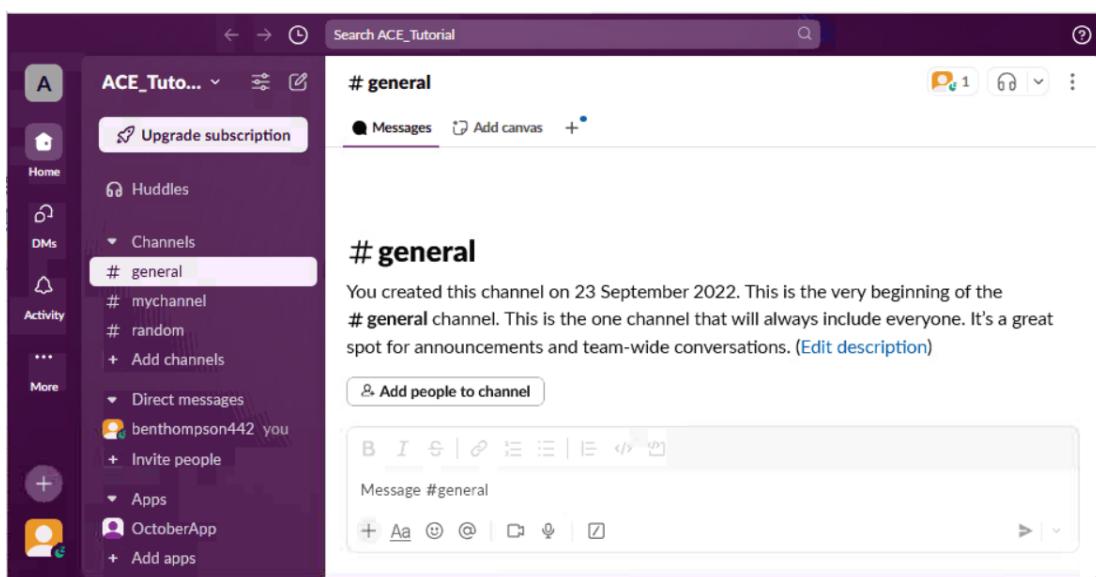
6. Once you have completed ID verification, the browser will update, asking you to continue if you are happy with the Terms of Service. Click **Continue**:



7. Depending what is already installed on the laptop or computer where you are following the instructions in this lab guide, you may also be prompted by the web browser to move to the Slack application:



You can also stick with the web browser by clicking 'use Slack in your browser':



In the example picture above, the Gmail ID had previously used Slack and already had a Slack workspace created. If you are using an email address to complete this Lab Guide which has never used Slack before, then you will be prompted for information which will enable the Slack workspace to be created. Some suggested values are provided below.

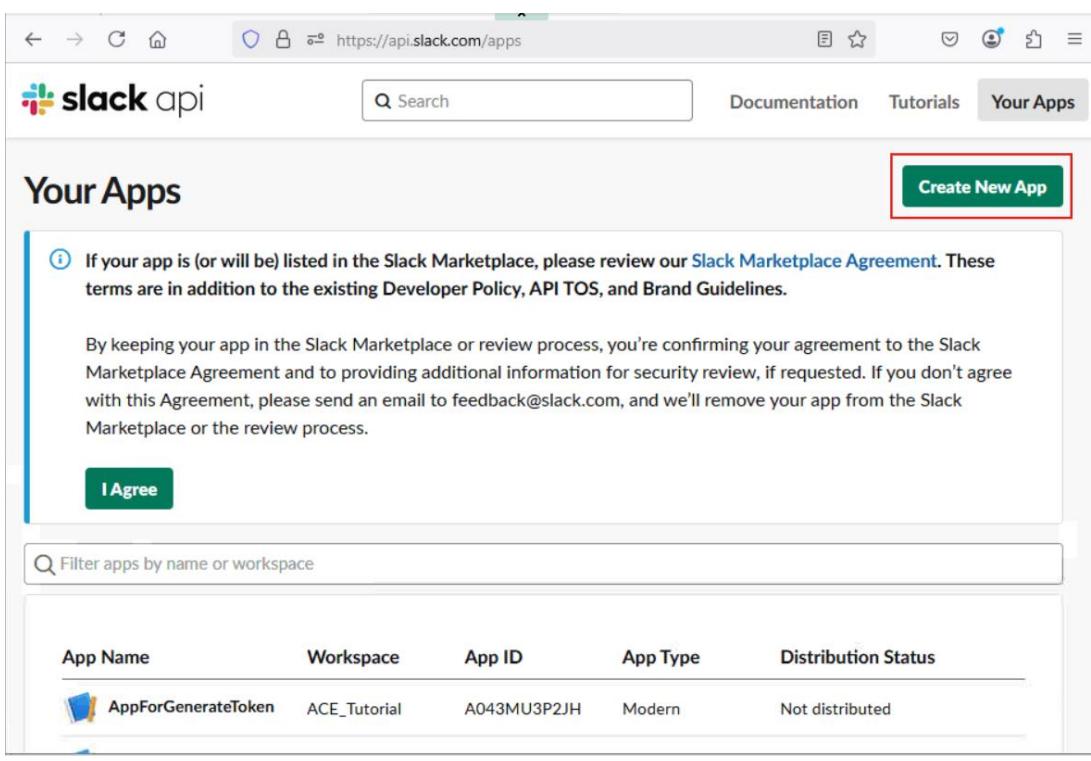
8. Suggested values for creating your new Slack workspace:
- Workspace Name: ACE_Tutorial
 - Set your name when prompted
 - Nobody else is on the team (skip this step)
 - What's your team working on: my ace channel
 - Choose the Free subscription

2.2 Get Slack Access Token

The ACE Slack Request node (either in a Toolkit flow or in a Designer flow) communicates with Slack using an Access Token. In this section you will obtain an access token to use with ACE.

9. In the same browser window where you are logged on to Slack, open a new browser tab and go to this URL: <https://api.slack.com/apps>

Click the **Create New App** button:



The screenshot shows a web browser displaying the Slack API Apps page at <https://api.slack.com/apps>. The page has a header with the Slack logo and navigation links for Documentation, Tutorials, and Your Apps. A prominent green button labeled "Create New App" is highlighted with a red box. Below the button, there is a note about the Slack Marketplace Agreement and a "I Agree" button. A search bar and a table listing existing apps are also visible.

App Name	Workspace	App ID	App Type	Distribution Status
AppForGenerateToken	ACE_Tutorial	A043MU3P2JH	Modern	Not distributed

10. Select the **From scratch** option:

The screenshot shows the Slack API 'Create an app' dialog box overlaid on the main 'Your Apps' page. The 'From scratch' option is highlighted with a red border. The dialog contains instructions: 'Use our configuration UI to manually add basic info, scopes, settings, & features to your app.' Below the dialog, the main page shows a table with one row: App Name (AppForGenerateToken), Workspace (ACE_Tutorial), App ID (A043MU3P2JH), App Type (Modern), and Distribution Status (Not distributed).

11. Set the App Name to be **MyACEApp** and choose the workspace that you just created and click **Create App**:

The screenshot shows the 'Name app & choose workspace' dialog box. The 'App Name' field is set to 'MyACEApp'. The 'Pick a workspace to develop your app in:' dropdown is set to 'ACE_Tutorial'. At the bottom right of the dialog, the 'Create App' button is highlighted with a red border.

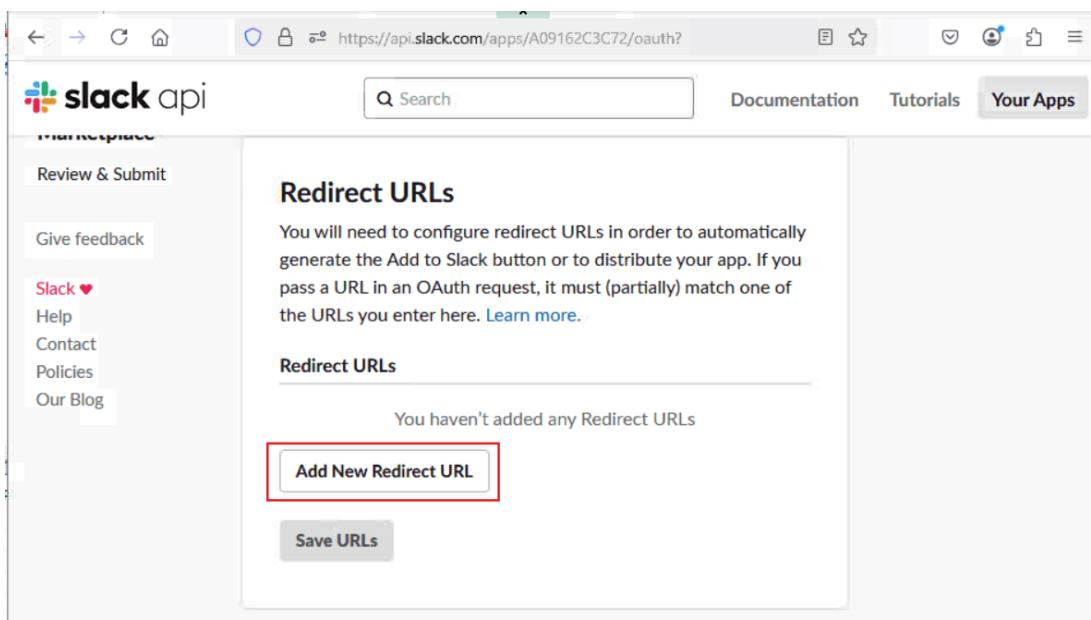
12. The **Basic Information** page for the App will open in the browser:

The screenshot shows the 'Basic Information' page for an app named 'MyACEA...'. The left sidebar has 'Basic Information' selected. The main content area is titled 'App Credentials' and contains fields for App ID (A09162C3C72), Date of App Creation (June 7, 2025), Client ID (4150207431696.9040080114240), and Client Secret. Buttons at the bottom right say 'Discard Changes' and 'Save Changes'.

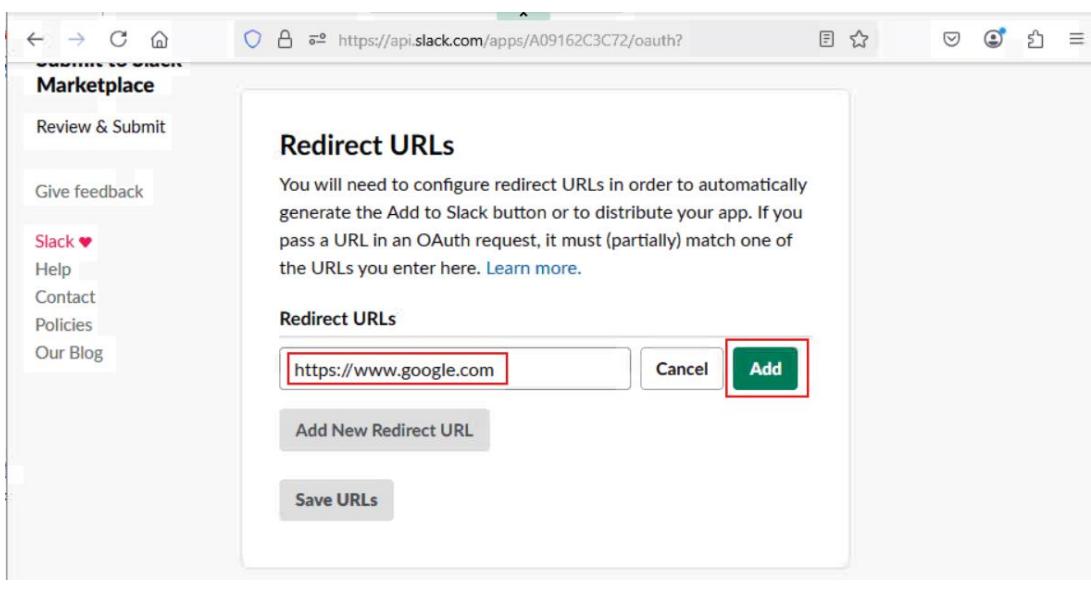
13. In the left navigation bar, select **OAuth & Permissions** (in the **Features** section of the navigation menu):

The screenshot shows the 'OAuth & Permissions' page. The left sidebar has 'OAuth & Permissions' selected. The main content area has a section titled 'Advanced token security via token rotation' with a note that at least one redirect URL needs to be set before token rotation can be opted in. A 'Opt In' button is present. Below it is a section titled 'OAuth Tokens' with a note that tokens will be generated after installing the app to a workspace.

14. In the OAuth & Permissions section, scroll down to the Redirect URLs and Click **Add New Redirect URL**:



15. Set the Redirect URL to <https://www.google.com> and then click the **Add** button:



16. Click the **Save URLs** button:

The screenshot shows a web browser window with the URL <https://api.slack.com/apps/A09162C3C72/oauth?>. The page is titled "Redirect URLs". It contains instructions about configuring redirect URLs for OAuth requests. A single URL, "https://www.google.com", is listed in the "Redirect URLs" section. Below the list are edit and delete icons. At the bottom of the form is a green "Save URLs" button.

17. The Redirect URLs box will now look like this:

The screenshot shows the same web browser window as the previous step, but with a red box highlighting the URL "https://www.google.com" in the "Redirect URLs" input field. The rest of the interface is identical to the previous screenshot.

2.3 Obtain Slack Client ID, Client Secret and Authorization Code

Please note: The following process requires a Slack authorization code. This authorization does expire after 10 minutes of being created, so you will need to complete the steps within this section within this 10 minute window (so if you need coffee or biscuits then get them now!)

18. In the left navigation bar, click **Basic Information**, and scroll down to the **App Credentials**. Copy the values of the **Client ID** and **Client Secret** into a text file (*note you will need to click “Show” in the Client Secret field in order to see the value*):

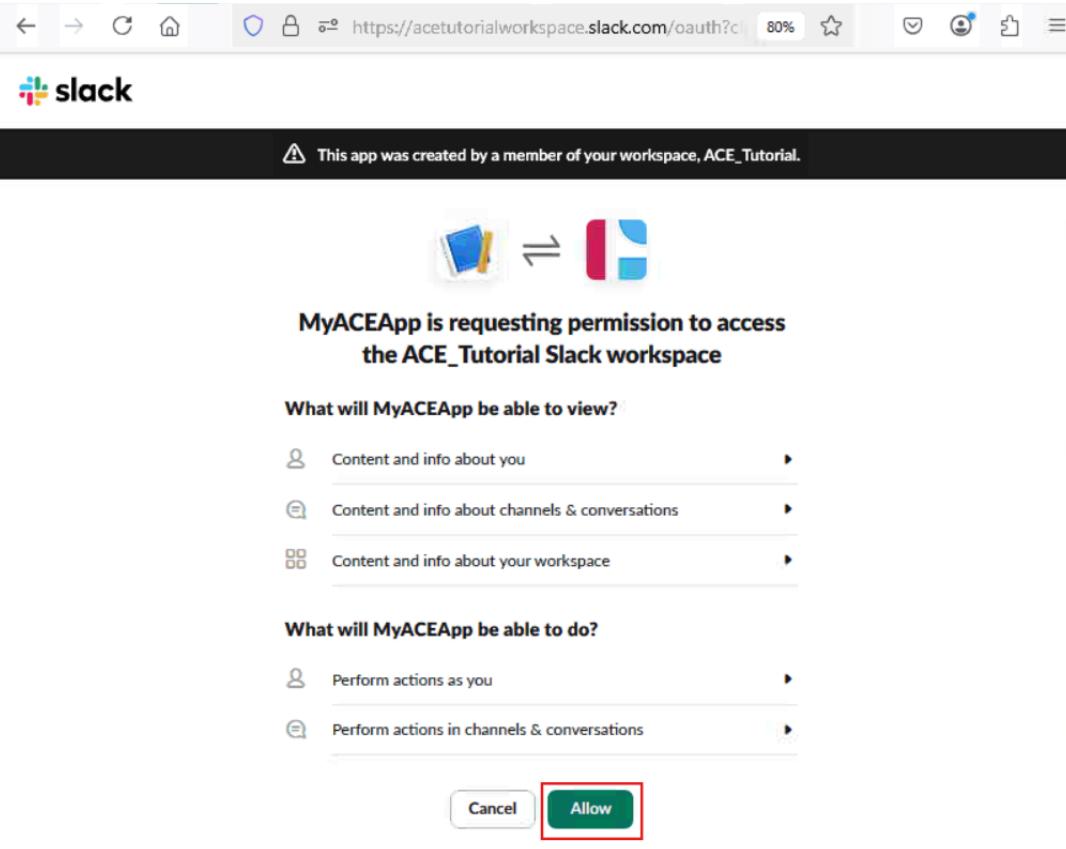
The screenshot shows the Slack API settings page. On the left, there's a sidebar with 'Basic Information' selected. The main area is titled 'App Credentials'. It contains fields for 'App ID' (A09162C3C72), 'Date of App Creation' (June 7, 2025), 'Client ID' (4150207431696.9040080114240), and 'Client Secret' (redacted). A note at the bottom says: 'You'll need to send this secret along with your client ID when making your oauth.v2.access request.'

19. Copy the following link, and then replace <client_id> with the **Client ID** from the previous step (note it may be easiest to construct this URL in the text file you just created):

https://slack.com/oauth/authorize?response_type=code&state=123456789&client_id=<client_id>&redirect_uri=https://www.google.com&access_type=offline&scope=identify,channels:read,users:read,users:read.email,chat:write:bot,im:read,mpim:read,files:write:user,files:read,groups:read,usergroups:read,chat:write:user,im:write,mpim:write,search:read

20. Copy the constructed link with the substituted value for Client ID and open the link in a web browser tab.

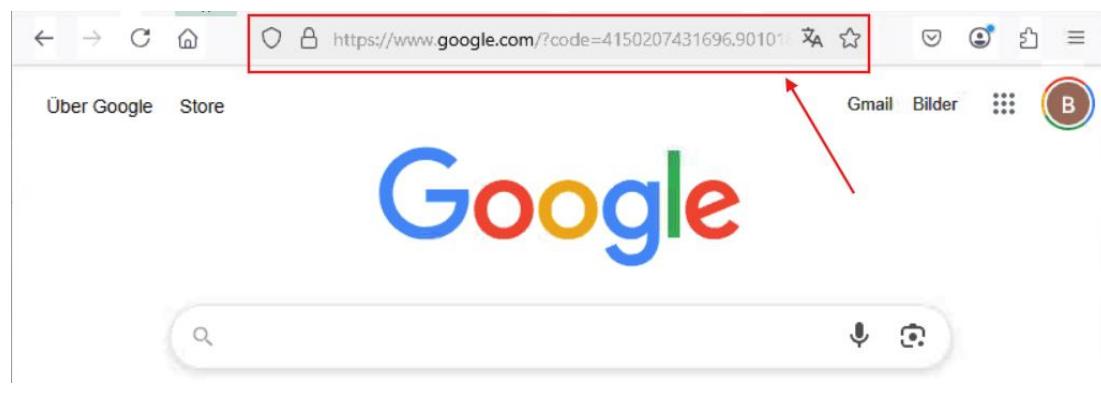
21. You will be prompted to authorize the app. Click **Allow**:

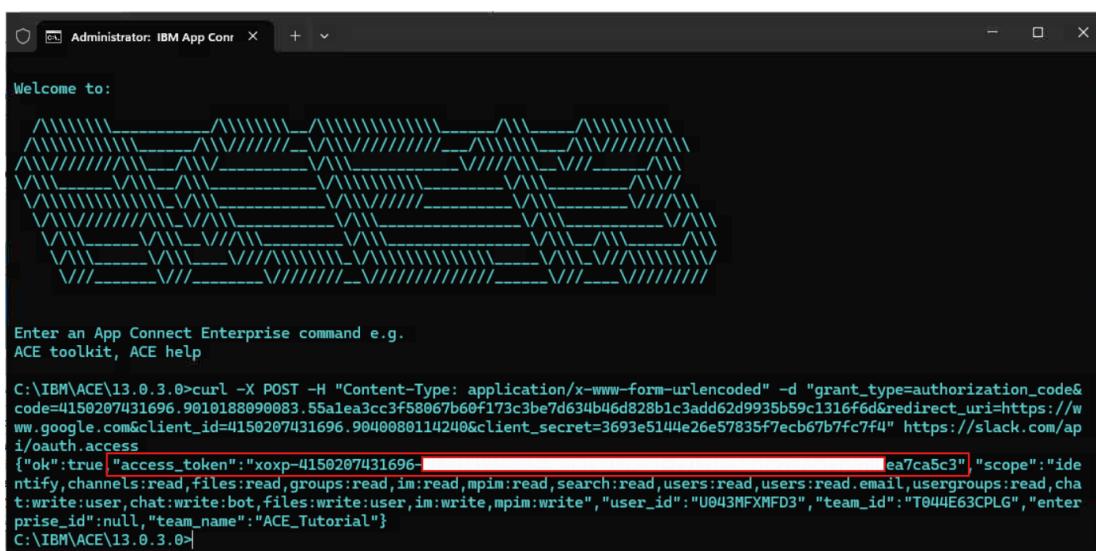


22. When you click **Allow**, this will change the address of the browser tab to the following:

<https://www.google.com/?code=<Code>&state=123456789>
(where <Code> is the authorization code)

Copy the authorization code (<Code>) to your text file as you will need it in the next step).



23.	<p>Copy the following curl command to your text file and replace the variables <client_id>, <client_secret> and <code> obtained above.</p> <pre>curl -X POST -H "Content-Type: application/x-www-form-urlencoded" -d "grant_type=authorization_code&code=<code>&redirect_uri=https://www.google.com&client_id=<client_id>&client_secret=<client_secret>" https://slack.com/api/oauth.access</pre>
24.	<p>Open IBM App Connect Enterprise Console for 13.0.3.0 and run the edited curl command (you will see the access_token you will need for ACE to communicate with Slack in the response from the curl command). Copy the value for the access token, as you will need this when configuring Slack in the Designer Event Driven Flow. Here is an example response from the curl command (<i>with the access token obscured</i>):</p>  <pre>C:\IBM\ACE\13.0.3.0>curl -X POST -H "Content-Type: application/x-www-form-urlencoded" -d "grant_type=authorization_code&code=4150207431696.9010188090083.55alea3cc3f58067b60f173c3be7d634b46d828b1c3add62d9935b59c1316f6d&redirect_uri=https://www.google.com&client_id=4150207431696.9040080114240&client_secret=3693e5144e26e57835f7ecb67b7fc7f4" https://slack.com/api/oauth.access {"ok":true,"access_token":"xoxp-4150207431696-[REDACTED]ea7ca5c3","scope":"ide ntify,channels:read,files:read,groups:read,im:read,mpim:read,search:read,users:read,users:read_email,usergroups:read,cha t:write:user,chat:write:bot,files:write:user,im:write,mpim:write","user_id":"U043MFXMFD3","team_id":"T044E63CPLG","enter prise_id":null,"team_name":"ACE_Tutorial"}</pre>
25.	<p>Copy the value of the access token into your text file and save the file somewhere so you can access it later in the lab. Note that the above instructions are also available in the ACE Knowledge Center at this URL:</p> <p>https://www.ibm.com/docs/en/app-connect/containers_cd?topic=slack-connecting-from-app-connect-enterprise-as-service</p>

3. Configure Jira

Note: If you have already completed a lab which uses Jira, then you can skip this section of the lab guide. These lab guide instructions assume you will use Jira Cloud. The items that you will require are:

- <yourdomain>.atlassian.net
- Jira Username
- Jira API Token

If you do not have these items, follow the instructions in this chapter of the guide to obtain one for your Slack user.

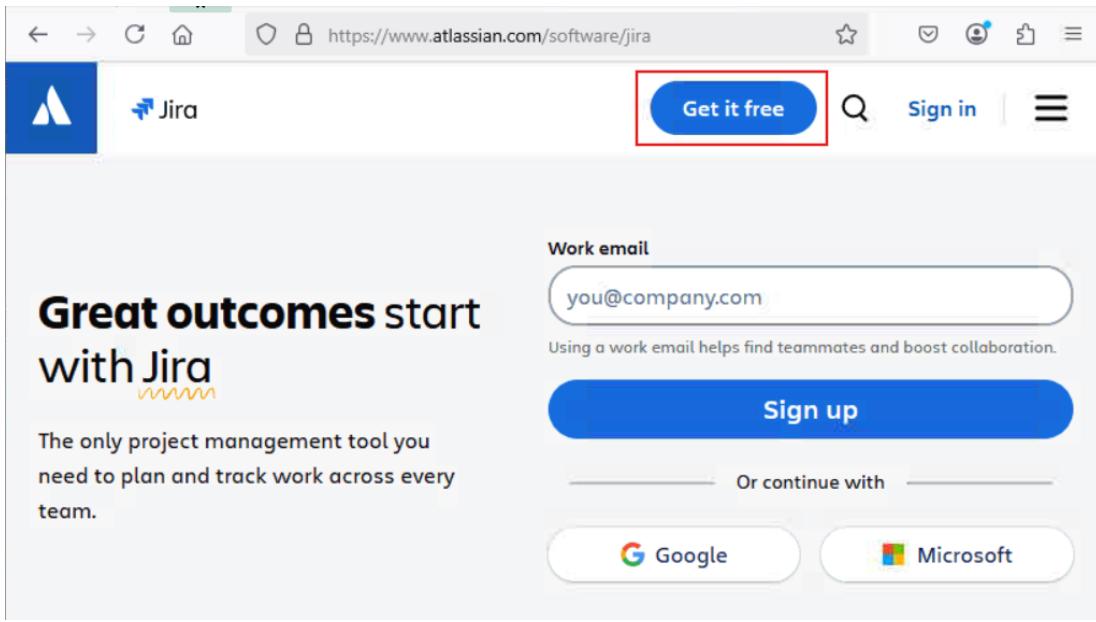
3.1 Create two issues

In this section you will configure some issues in Jira so that the Designer Flow Editor can obtain the details and send them to Slack.

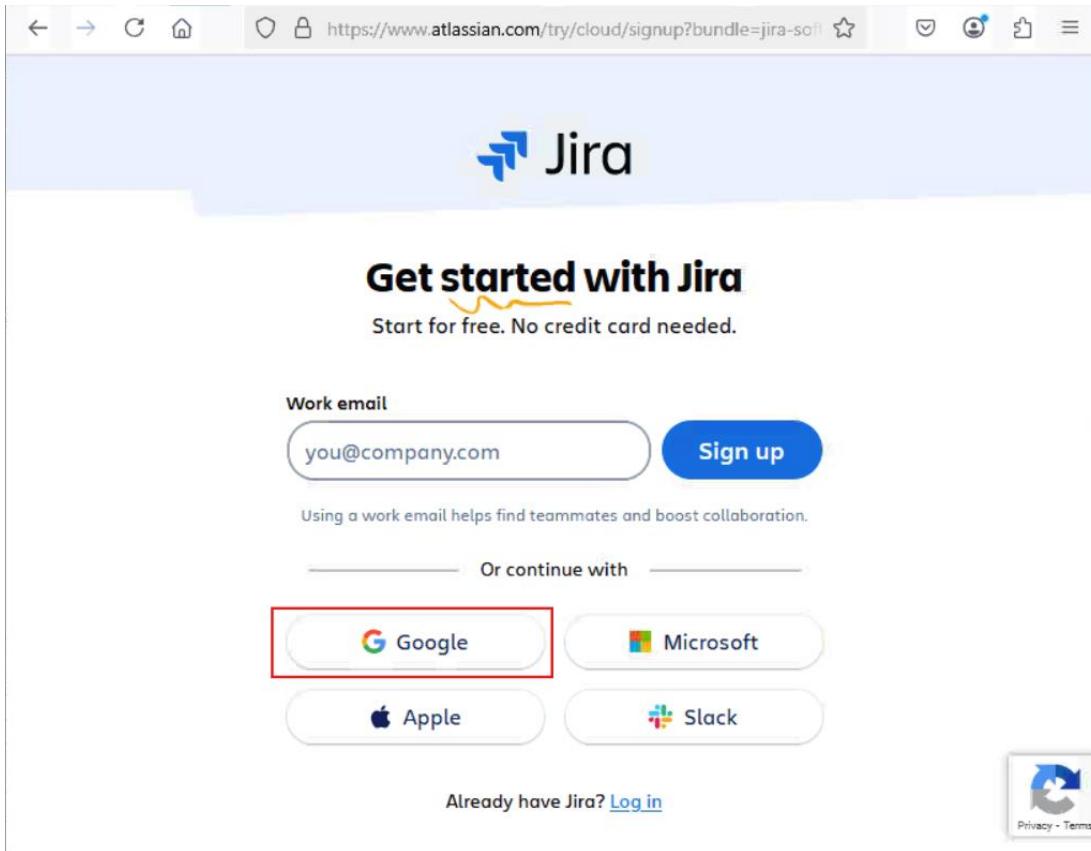
26. Open a browser window and go to the following URL:

<https://www.atlassian.com/software/jira>

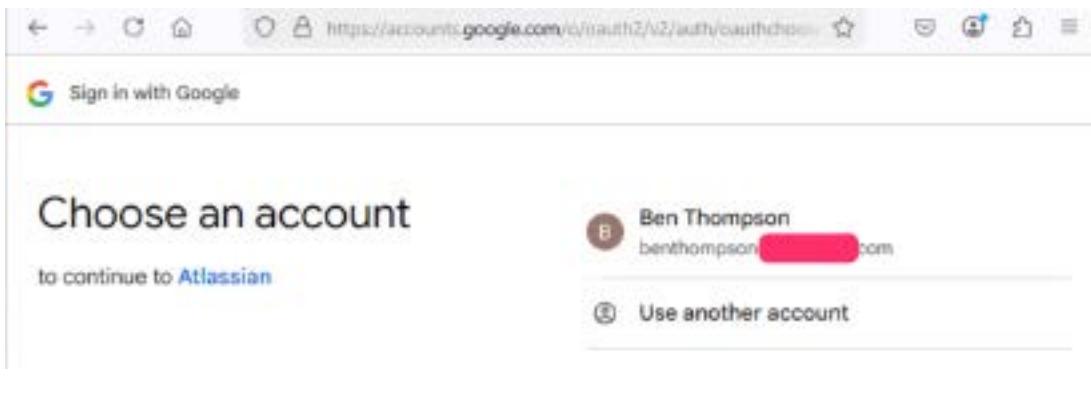
27. Click the **Get it free** button:



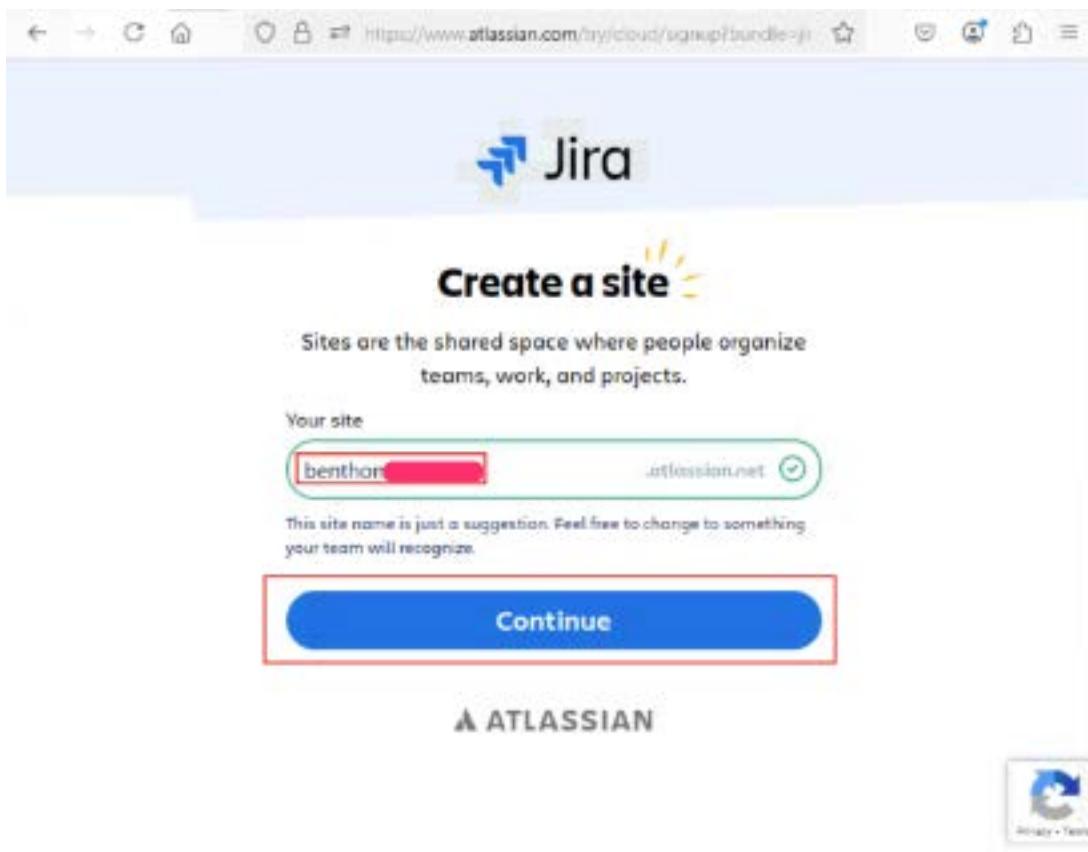
28. Click **Continue with Google**



29. Choose your Gmail account and sign in with it (you may need to go through 2-factor authentication to do this):

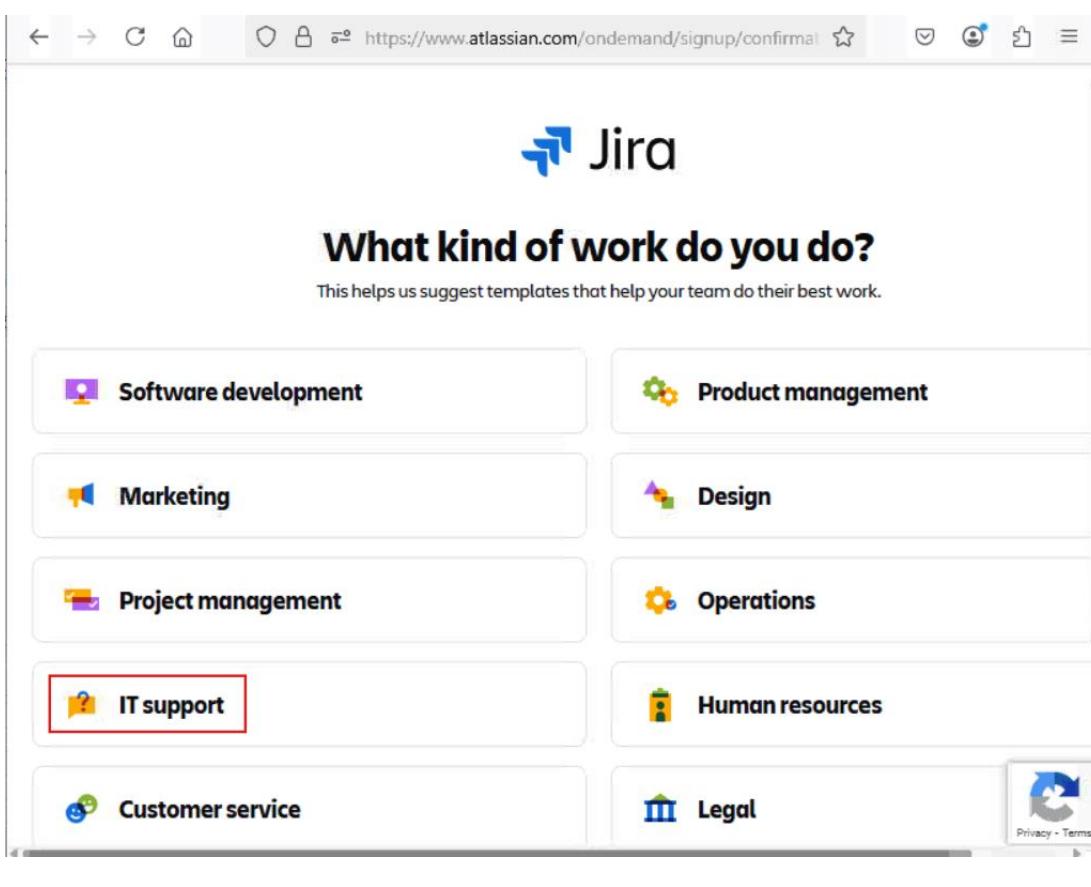


30. Specify a name for your site (or just agree with the suggestion as shown below), and click **Continue**:

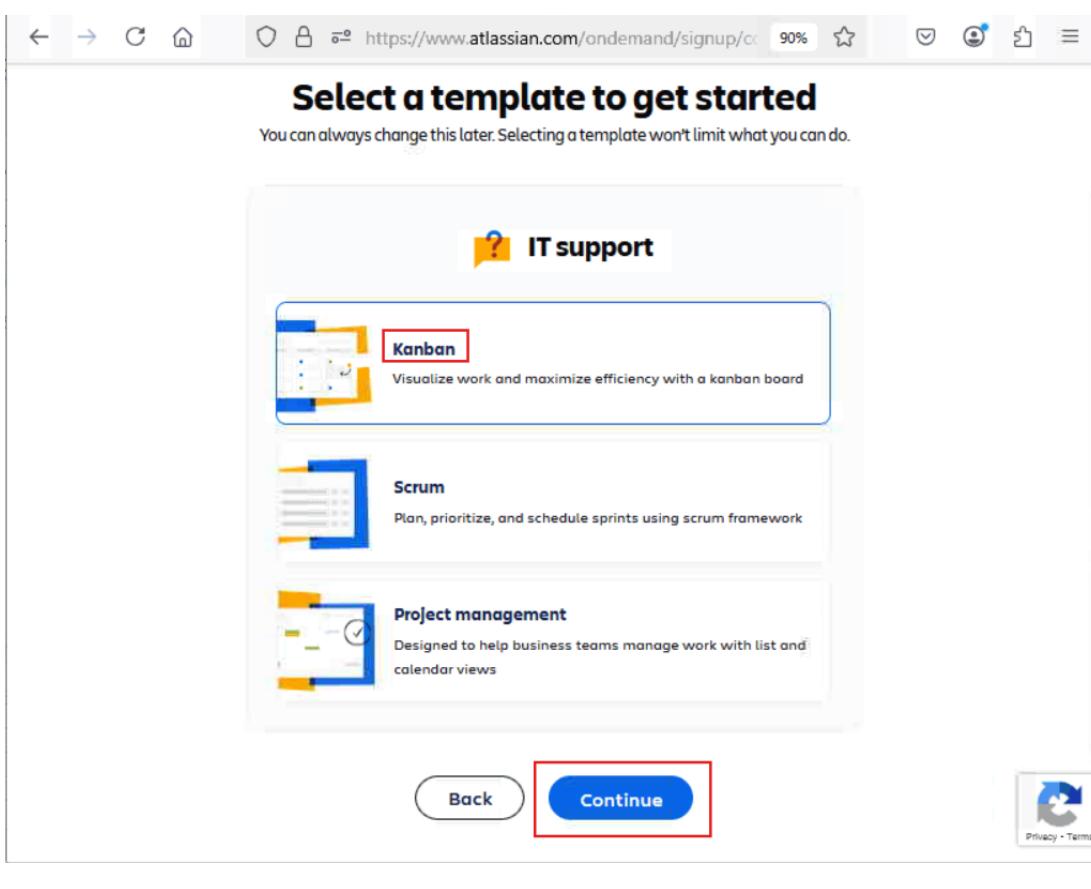


Copy the value for **Your site** to a text file, as you will use this fully qualified site address as your Jira host and port when configuring your Jira Request Node in your ACE flow later in this lab guide. In the example shown above, this is **benthompson442.atlassian.net**

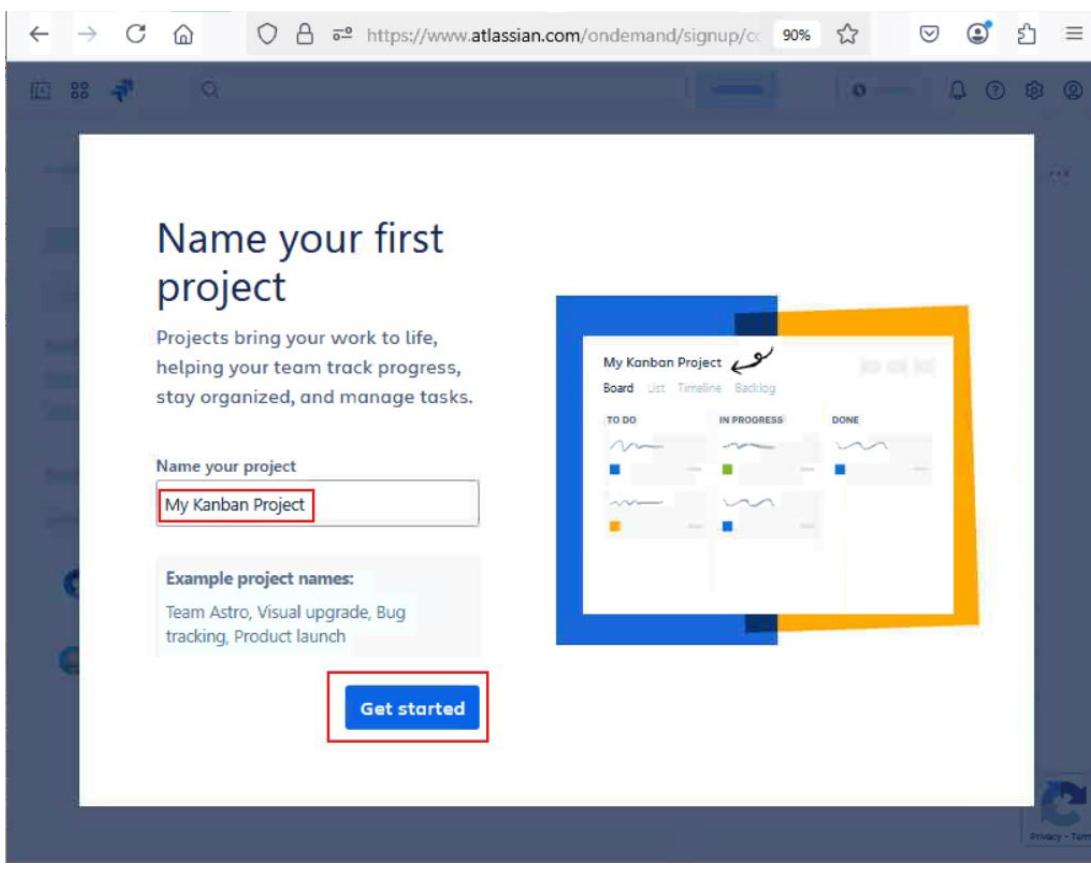
31. Select one of the options. At the time of writing this lab guide, the options were as shown below and we went with **IT support**:



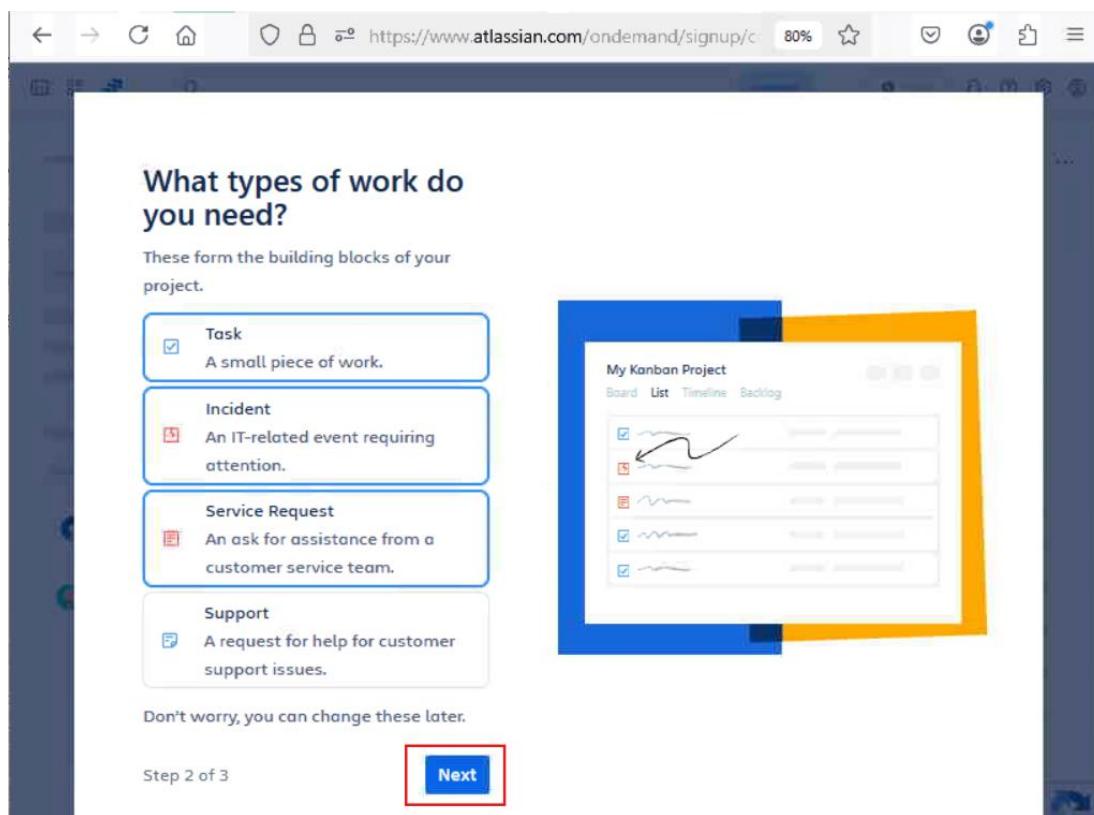
32. On the next screen you will be asked to **Select a template to get started**. Stick with the default of **Kanban** and click the **Continue** button:



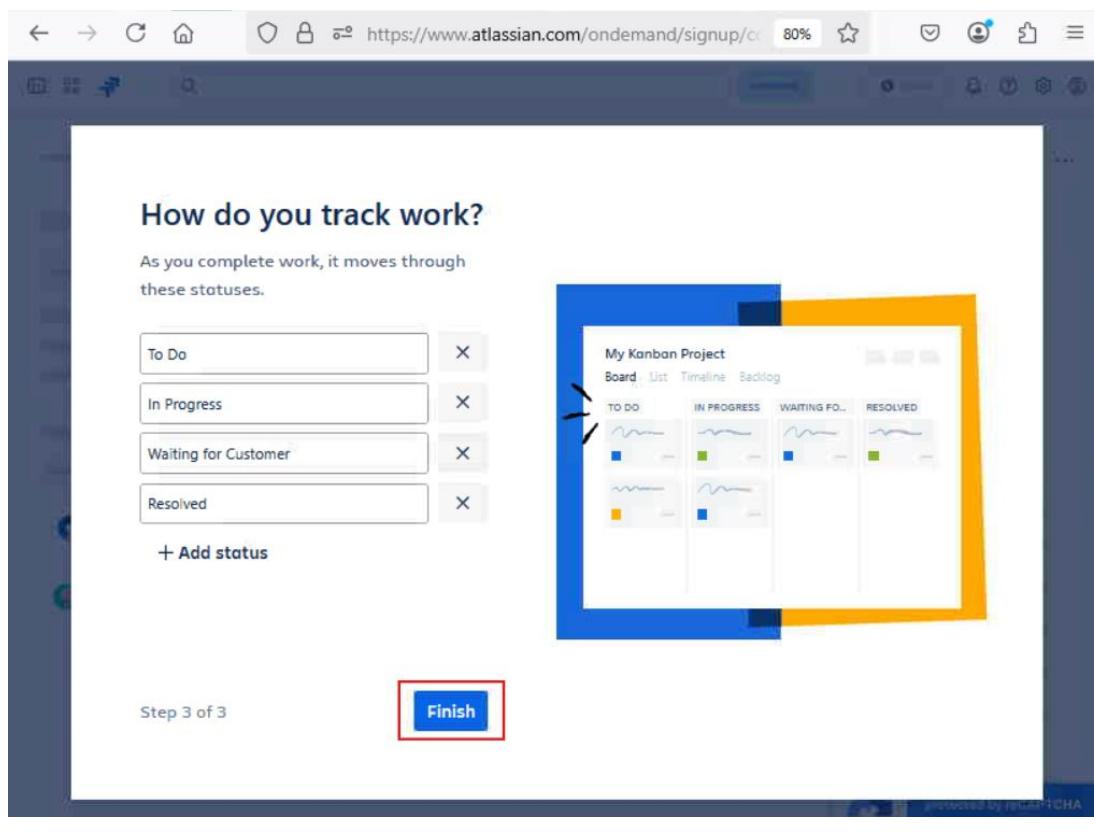
33. On the next page, stick with the default project name of **My Kanban Project** and click the **Get started** button:



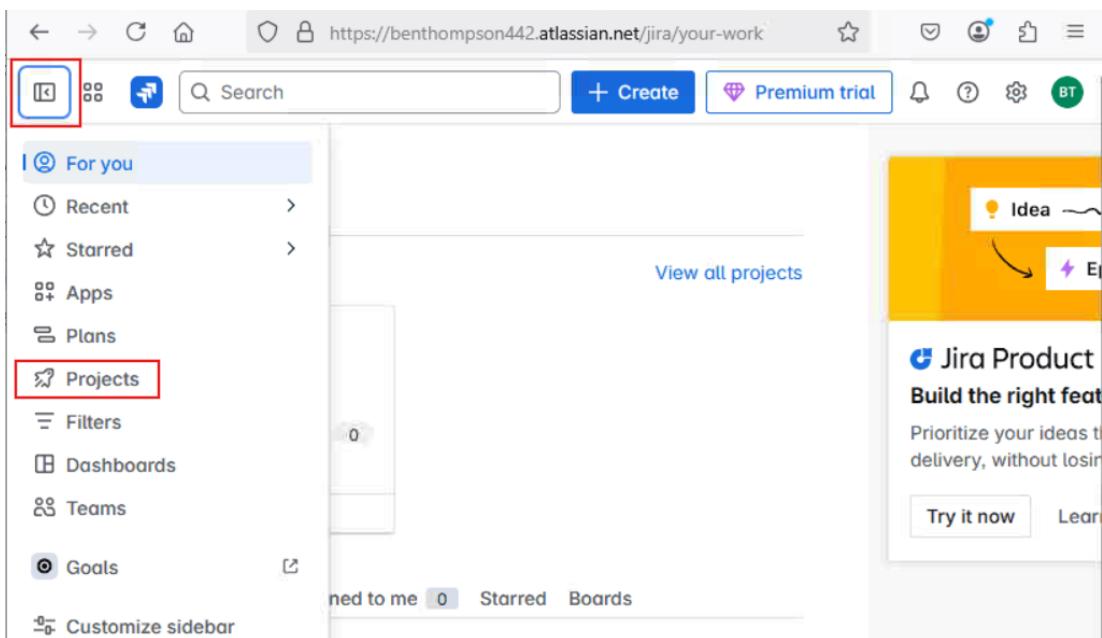
34. On the next screen, stick with the defaults offered as shown and click the **Next** button:



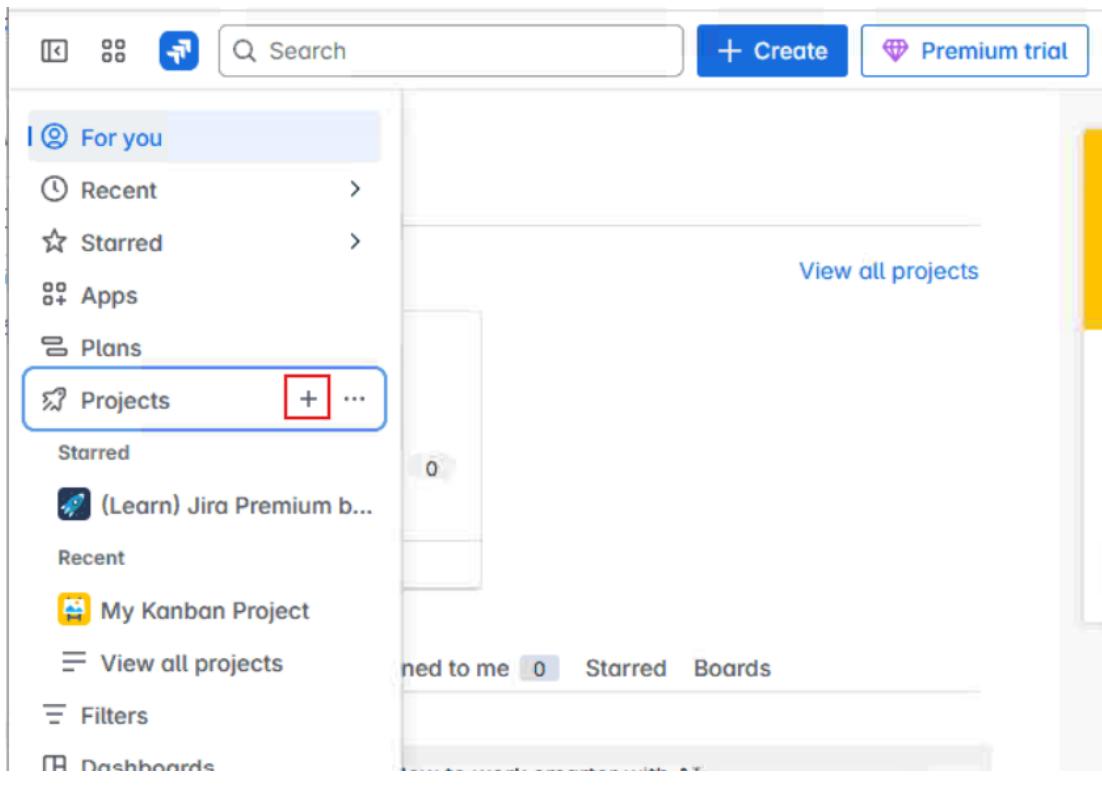
35. On the last step, stick with the defaults again and click the **Finish** button:



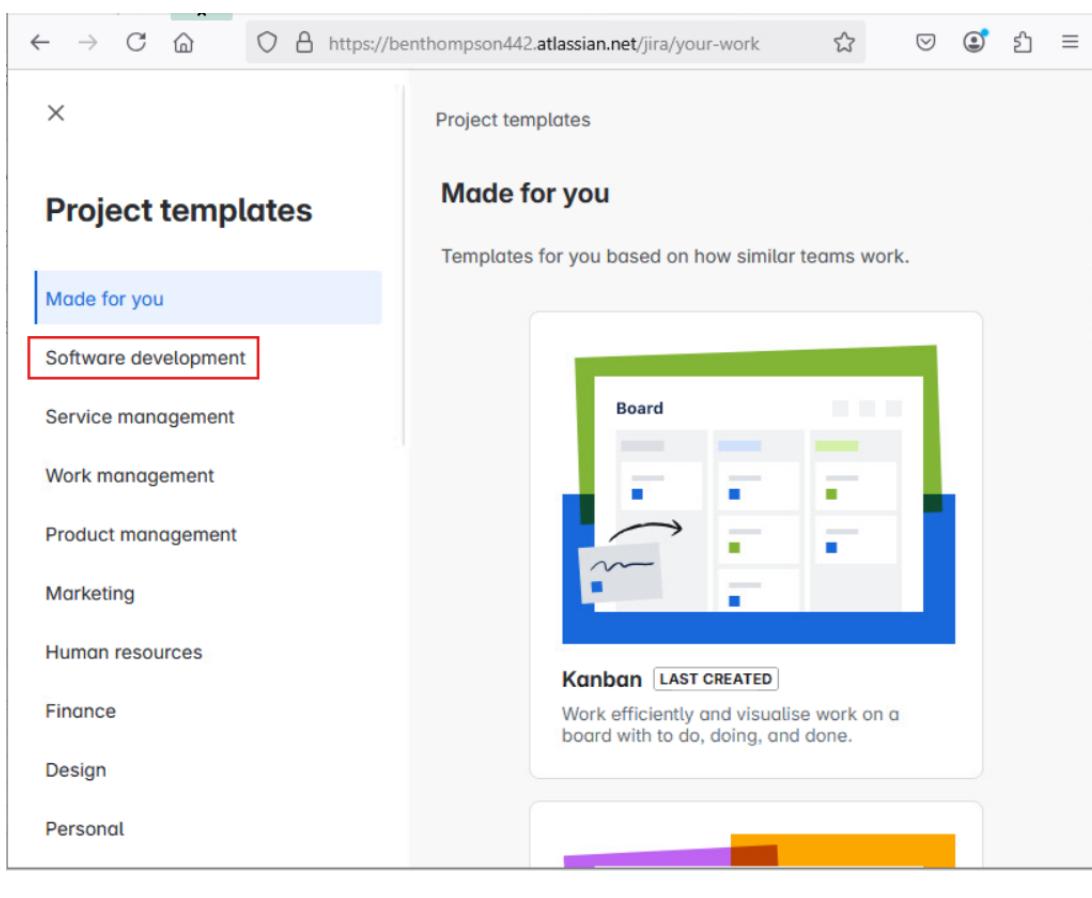
36. We will now create a new project. Start by expanding the menu on the left side using the icon in the corner as shown below. From the menu, click on **Projects**:



The menu will expand and you can click on the plus icon to create a new project:



37. The **Project templates** menu will be presented. Select the **Software development** option:



38. Select the **Software development** options and select **Bug tracking**:

The screenshot shows the Jira Project templates interface. On the left, a sidebar lists categories: Service management, Work management, Product management, Marketing, Human resources, Finance, Design, and Personal. The 'Software development' category is highlighted with a blue selection bar. To the right, there are several project template cards. One card for 'Product roadmap' is partially visible at the top. Below it is a card for 'Prioritization'. The 'Bug tracking' card is at the bottom, featuring a blue bug icon with a red 'X' and the text 'Manage a list of development tasks and bugs.' A red arrow points from the 'Software development' sidebar to the 'Bug tracking' card.

39. Click the option to **Use template**:

The screenshot shows the 'Bug tracking' template details page. The left sidebar shows the 'Software development' category selected. The main content area has a blue header 'Bug tracking'. Below the header, there is a description: 'Capture, track and resolve bugs and issues throughout your entire development process. Provide a single source of truth of all your issues and help your team prioritize against their big picture goals, while continually delivering value to your customers.' There is also a section titled 'Identify and capture bugs' with a sub-description: 'See all your bugs in one place. Once you've identified a bug, capture its details by creating a work item from anywhere in your project. Each unique work type can have its own custom workflow.' On the right side, there is a 'Product' section showing 'Jira', a 'Recommended for' section mentioning 'Teams that are capturing, tracking, and resolving bugs', and a 'Work types' section listing 'Epic', 'Bug', 'Improvement', and 'New Feature'. A prominent blue 'Use template' button is located at the bottom right of the main content area.

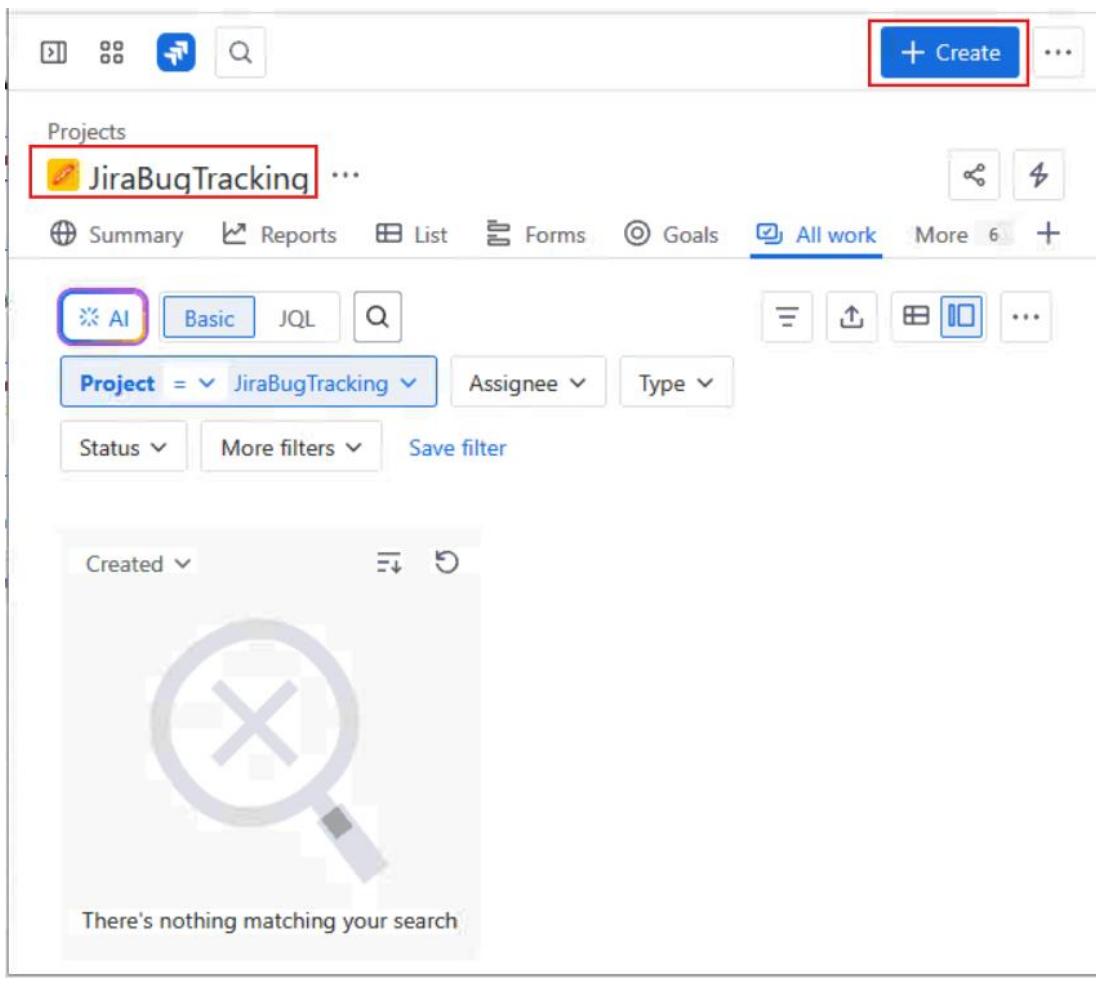
40. Add a name for your project – make a note of the name as you will need it when configuring ACE later in the lab guide. The **Key** field will update itself (stick with the **Jira** suggestion) automatically as you type in the **Name**. Click the **Create project** button:

The screenshot shows the 'Add project details' page in Jira. The URL in the browser is <https://benthompson442.atlassian.net/jira/your-work>. The page title is 'Add project details'. It says 'Explore what's possible when you collaborate with your team. Edit project details anytime in project settings.' Required fields are marked with an asterisk (*). The 'Name' field has 'JiraBugTracking' entered. The 'Key' field has 'JIR' entered. There is a checkbox for 'Share settings with an existing project' which is unchecked. On the right, there is a 'Template' section with a 'Bug tracking' option selected, which is described as 'Manage a list of development tasks and bugs.' Below the template section are 'Cancel' and 'Create project' buttons, with 'Create project' being highlighted by a red box.

41. You can **Skip** the next page which offers you the chance to invite teammates (we don't need this for the lab):

The screenshot shows the 'Bring the team with you' page in Jira. The URL in the browser is <https://benthompson442.atlassian.net/jira/your-work>. The page title is 'Bring the team with you'. It says 'Invite these teammates to your project, and create work together.' Below this is a 'Invite teammates' section with a 'Enter names or emails' input field containing 'John Smith' and a 'Role' dropdown set to 'Administrators'. At the bottom are 'Skip' and 'Invite and continue' buttons, with 'Skip' being highlighted by a red box.

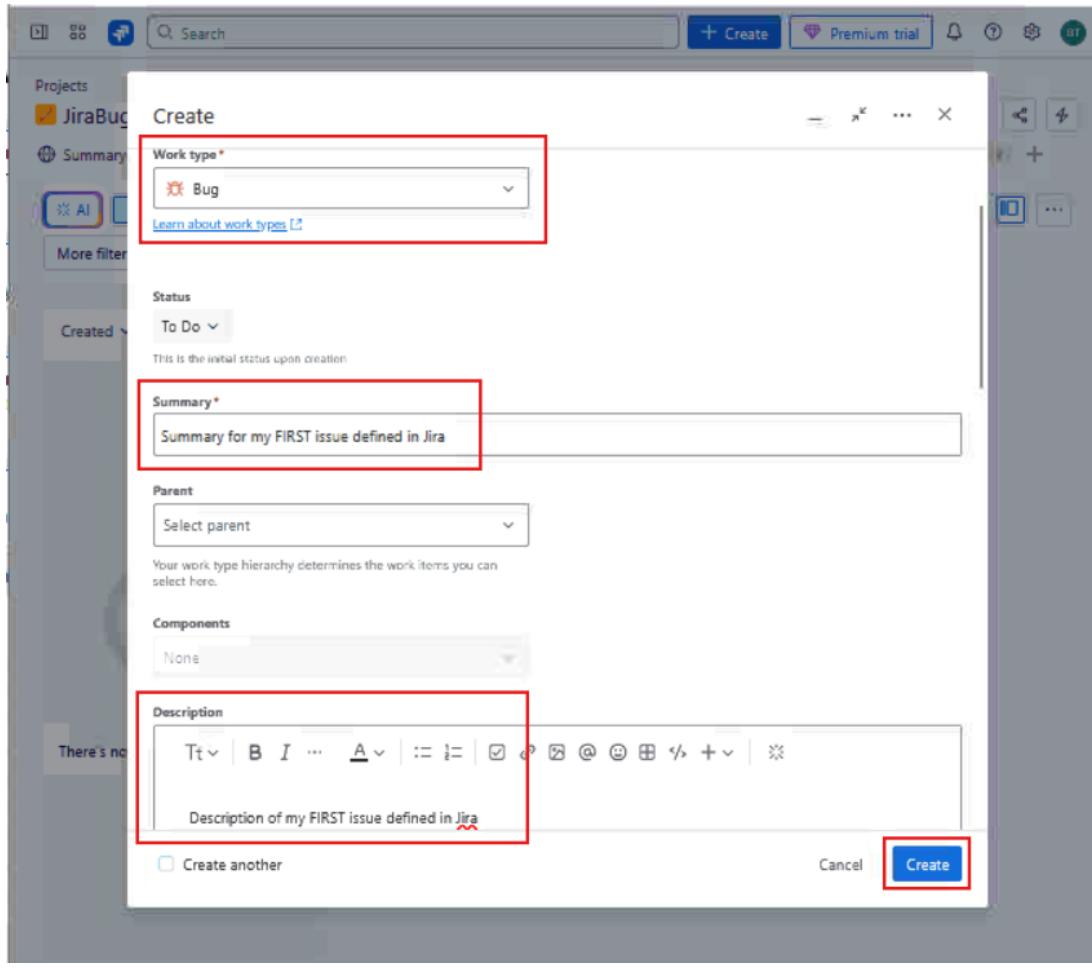
42. The default view for the Jira Bug Tracking template looks like the picture below at the time of creating this lab. Click the **Create** button to create an issue:



43. In the **Create issue** popup window, select the following:

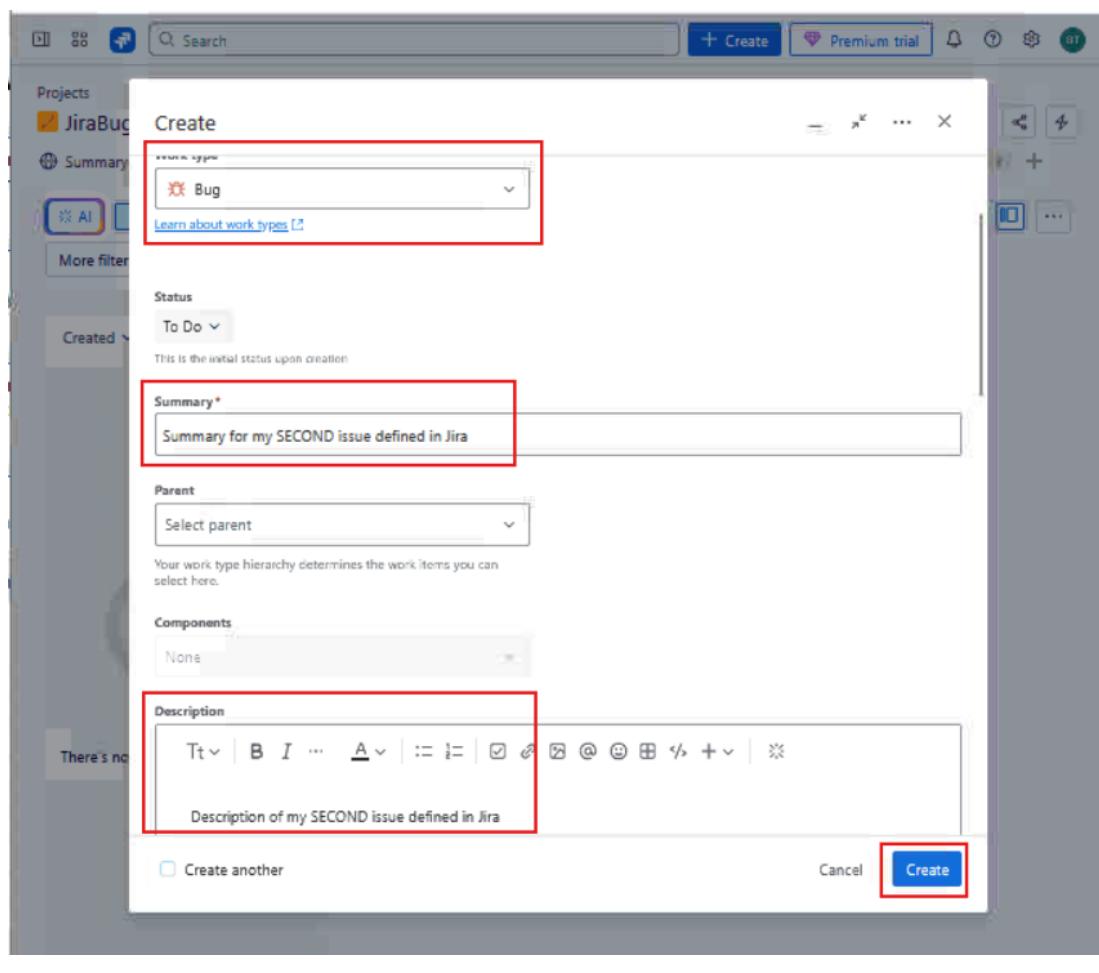
- **Issue type** = Bug
- **Summary** = Summary for my FIRST Issue defined in Jira
- Scroll down and set the **Description** to: Description of my FIRST Issue in Jira.

Leave the defaults for the other fields and click the **Create** button:



44. Repeat the same process to create a second issue. In the **Create issue** popup window, select the following:
- **Issue type** = Bug
 - **Summary** = Summary for my SECOND Issue defined in Jira
 - Scroll down and set the **Description** to: Description of my SECOND Issue in Jira.

Leave the defaults for the other fields and click the **Create** button:



45. You can use the **List** link to list the issues which you just created:

Type	Key	Summary	Status	Comments
XX	JIR-1	Summary for my FIRST issue defined in Jira	TO DO	Add comment
XX	JIR-2	Summary for my SECOND issue defined in Jira	TO DO	Add comment

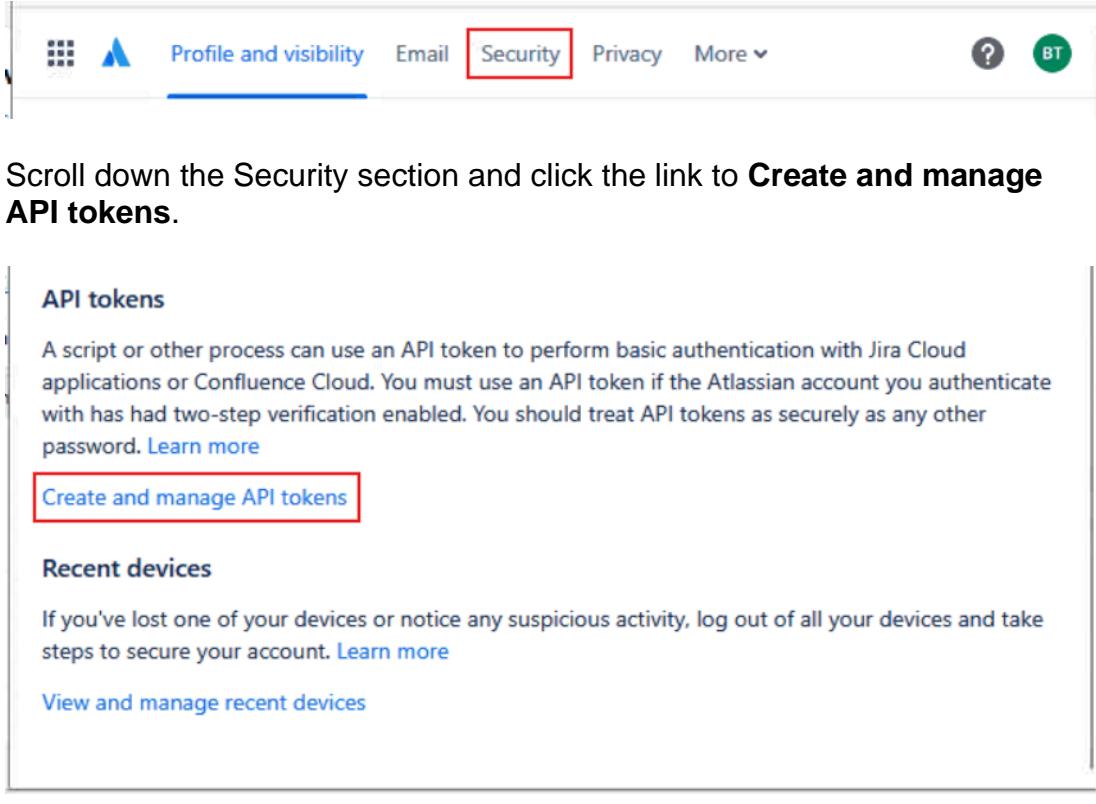
3.2 Create Jira API Token

When communicating with Jira, the ACE Designer Flow Editor requires an **API Token**, which we will create now in this section.

46. In the top right corner of the Jira web page, click the icon that represents your userid and from the menu select **Account settings**:

- BT Ben Thompson
benthompson442@gmail.com
- [Profile](#)
- Account settings**
- [Theme](#)
- [Open Quickstart](#)
- [Switch account](#)
- [Log out](#)

47. In the header of your profile, select the **Security** tab:



The screenshot shows the Atlassian profile header with several tabs: Profile and visibility, Email, Security (which is highlighted with a red box), Privacy, and More. Below the header, the 'Security' section is displayed. It includes a heading 'API tokens' with a descriptive paragraph and a link 'Create and manage API tokens' (also highlighted with a red box). Another heading 'Recent devices' is shown with a link 'View and manage recent devices'.

Scroll down the Security section and click the link to **Create and manage API tokens**.

API tokens

A script or other process can use an API token to perform basic authentication with Jira Cloud applications or Confluence Cloud. You must use an API token if the Atlassian account you authenticate with has had two-step verification enabled. You should treat API tokens as securely as any other password. [Learn more](#)

[Create and manage API tokens](#)

Recent devices

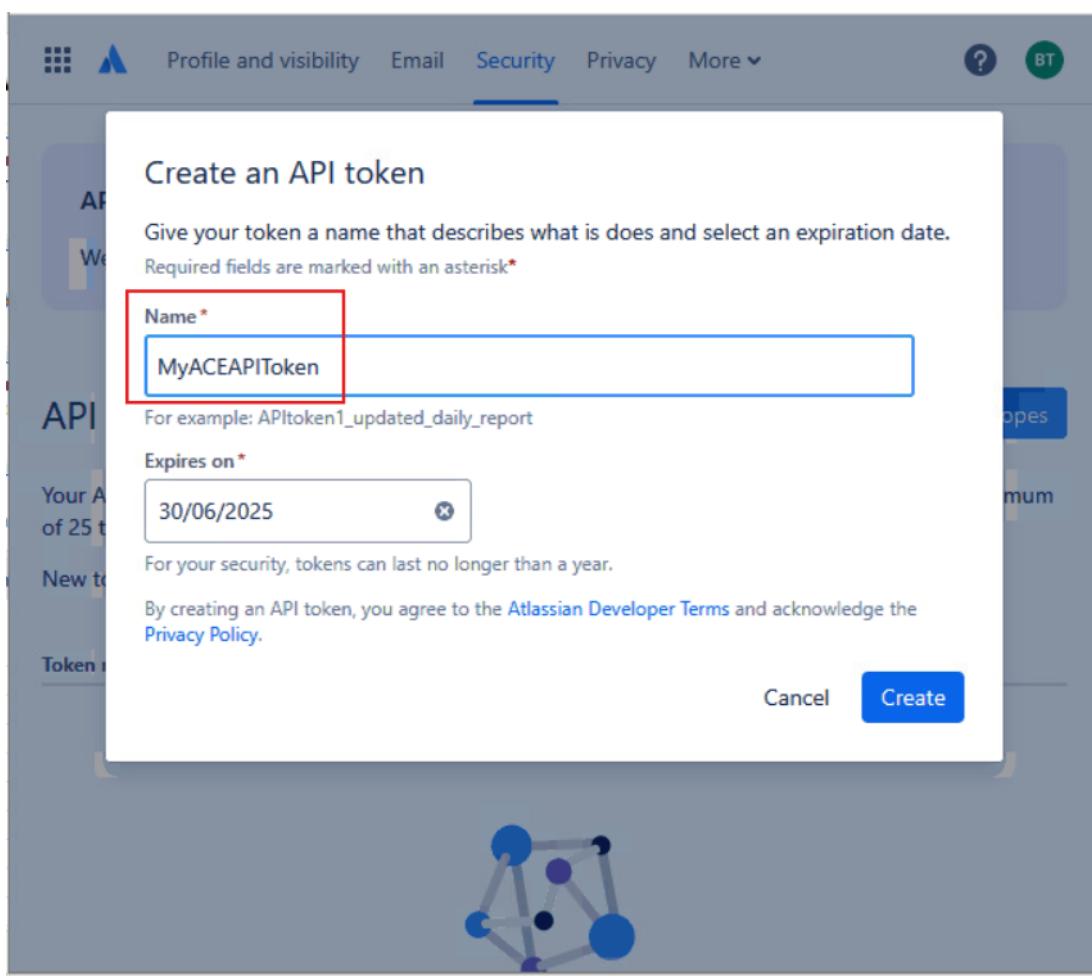
If you've lost one of your devices or notice any suspicious activity, log out of all your devices and take steps to secure your account. [Learn more](#)

[View and manage recent devices](#)

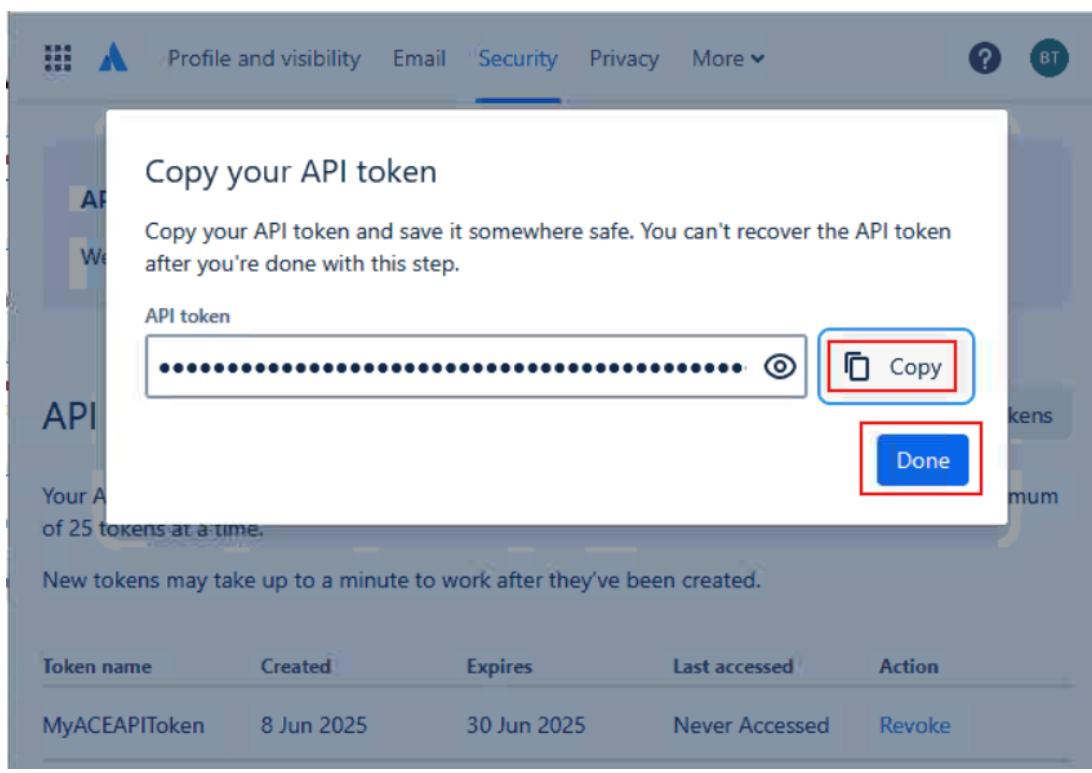
48. Click the button to **Create API token**:

The screenshot shows the 'API Tokens' section of the IBM App Connect interface. At the top, there is a message: 'API tokens without scopes are being deprecated' followed by 'We encourage creating the more secure API tokens with scopes whenever possible.' Below this, there are two buttons: 'Create API token' (highlighted with a red box) and 'Create API token with scopes'. A note below the buttons states: 'Your API tokens need to be treated as securely as any other password. You can only create a maximum of 25 tokens at a time.' Another note says: 'New tokens may take up to a minute to work after they've been created.' At the bottom, there is a decorative graphic of interconnected nodes.

49. Provide a **Name** for the API token such as `MyACEAPIToken` and click **Create**:



50. IMPORTANT: Make a copy of your API token (*you will need this later when configuring ACE*) and save it somewhere temporary such as in a text file:

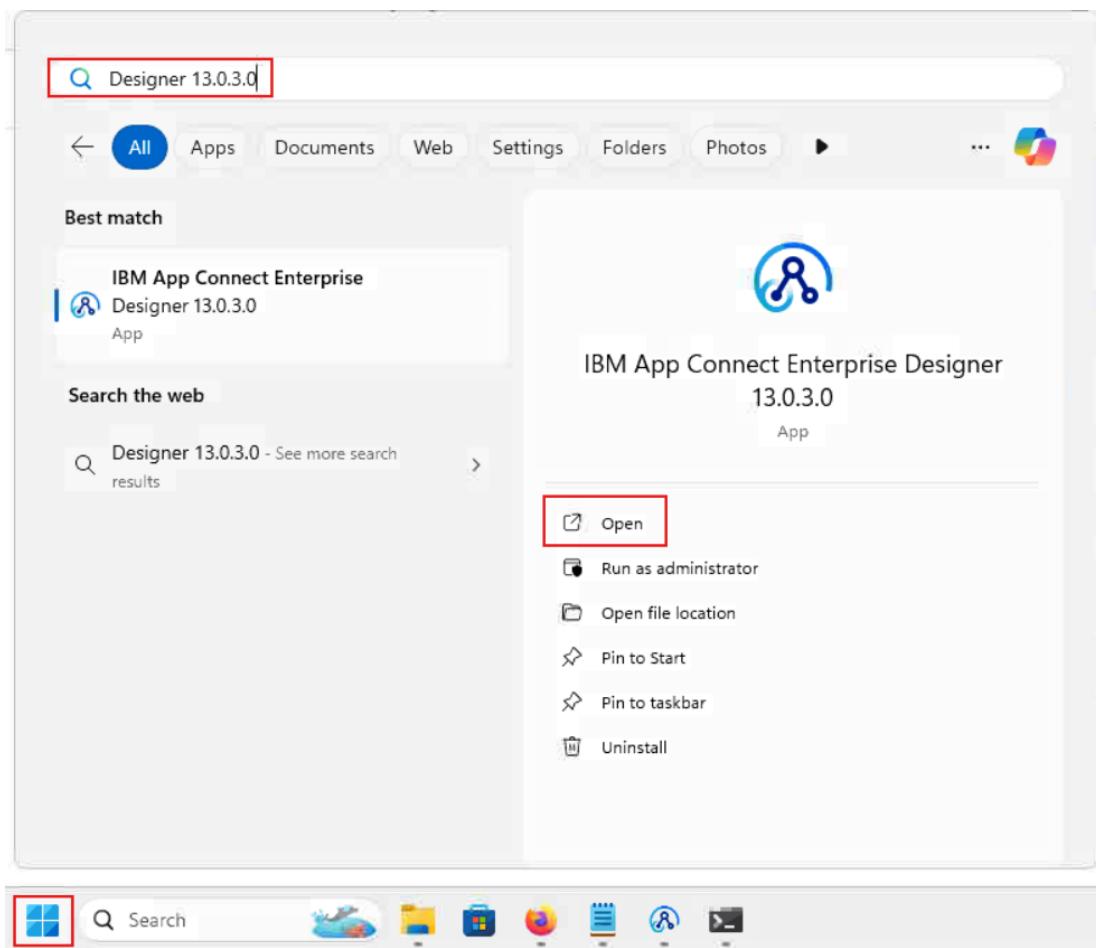


4. Write Data from Jira to Console Log

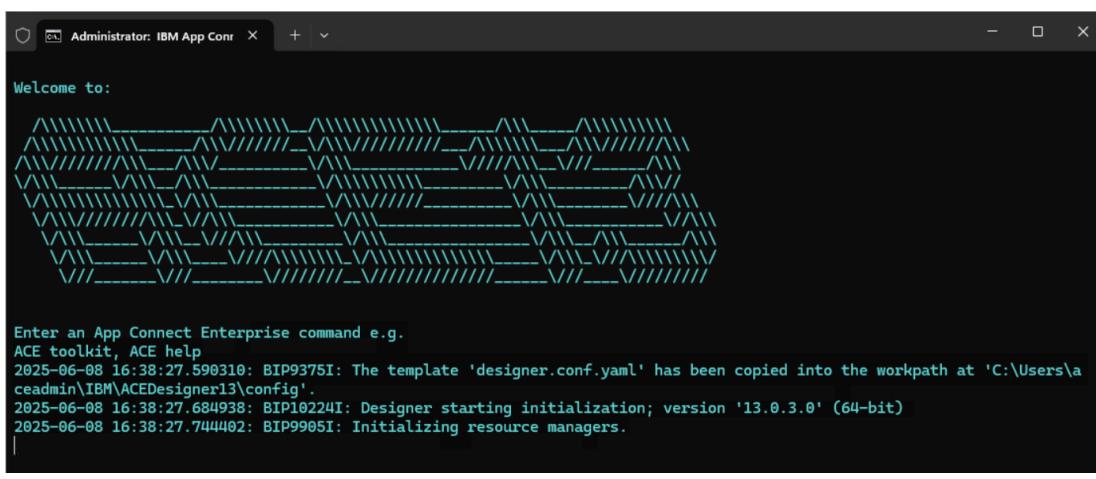
In this section, you will use the ACE 13 Designer to create a very simple Event-driven message flow that will capture information about Add or Changed Issues in Jira and write the Jira Key field to the Designer Console Log.

4.1 Start Designer Flow Editor

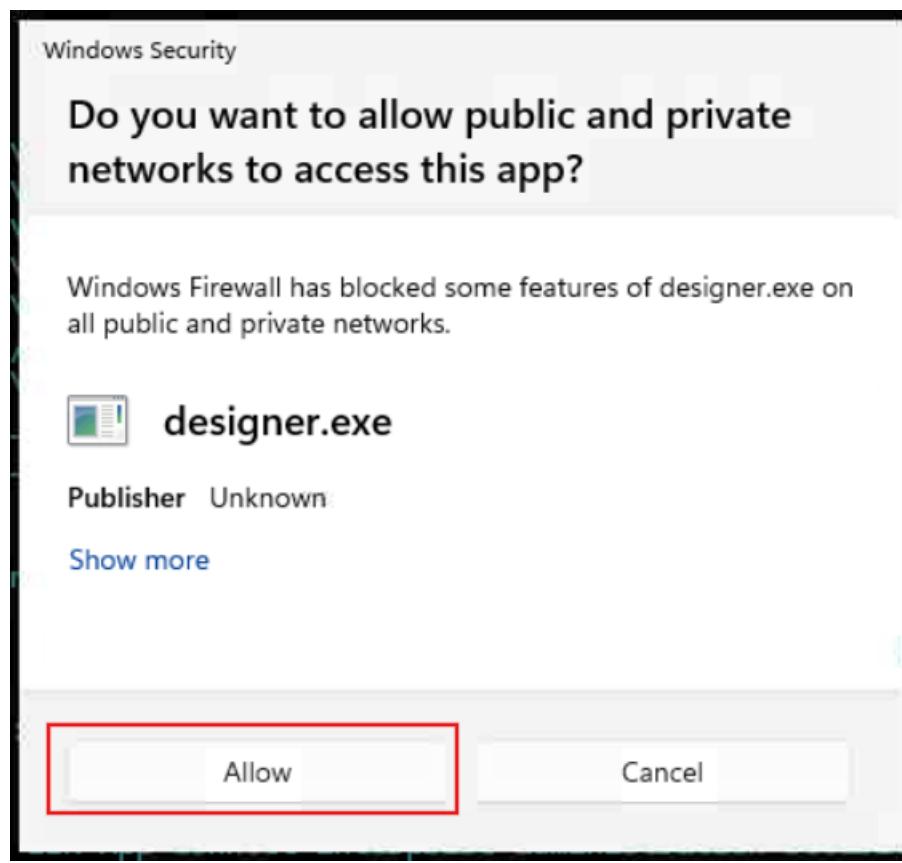
51. From the Windows start menu, type the search term of “Designer 13.0.3.0” and when the result appears, select to **Open the IBM App Connect Enterprise Designer 13.0.3.0** application:



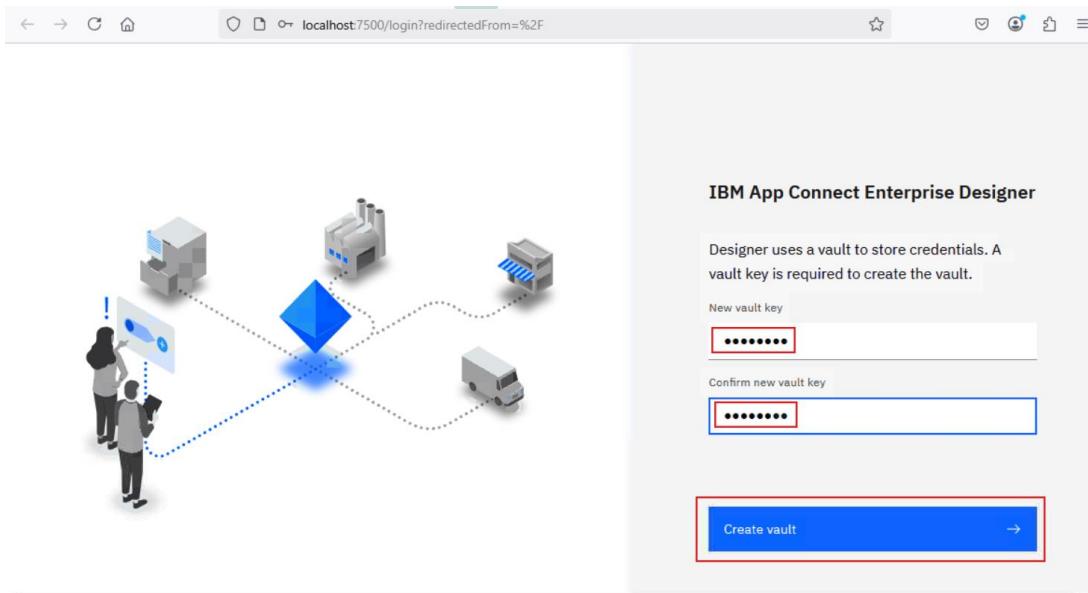
52. At first you will see a window launched which looks a bit like an ACE Command Console. This window initialises the Designer process, which actually also involves creating a runtime integration server which will be used whenever you want to test out a message flow.



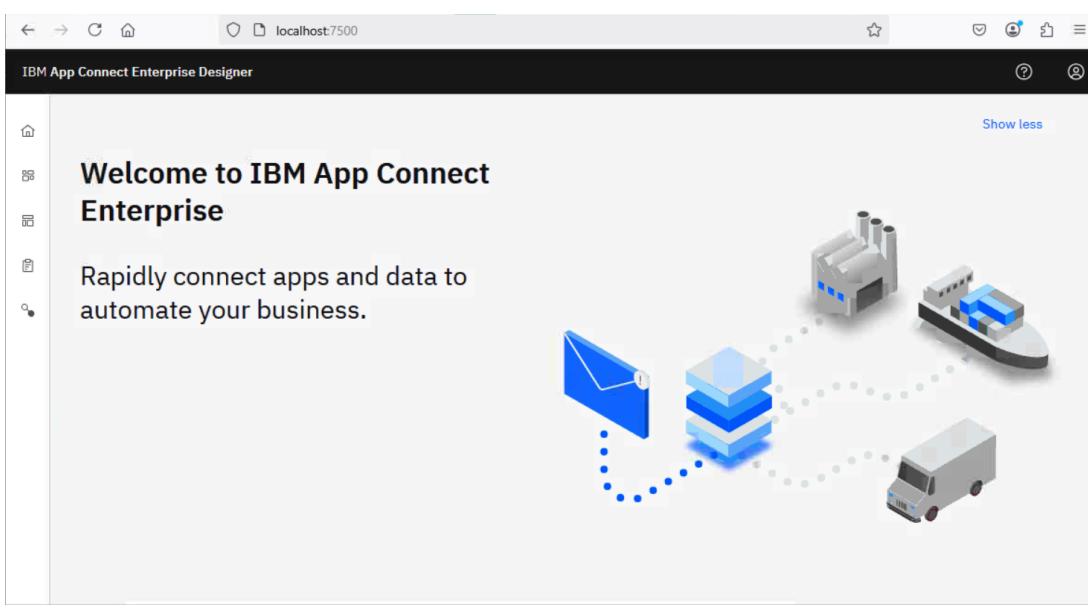
53. If this is the first time you've launched the Designer after ACE was first installed then you may see a Windows Security pop-up such as the one shown below. Click the **Allow** button:



54. After a few seconds, the Designer process will be ready and a web browser tab will be opened ready for you to use. As you use Designer to create message flows, we will be connecting to various applications and third party systems, so Designer requires a means of safely storing the credentials and secrets which are used to access them securely. ACE uses its vault technology for this purpose. When first used, a vault key must be entered (and then repeated for confirmation). Use the value `passw0rd` and then click the **Create vault** button:



The Designer welcome screen will be displayed:



55. From the left side menu select the **Designer** option:

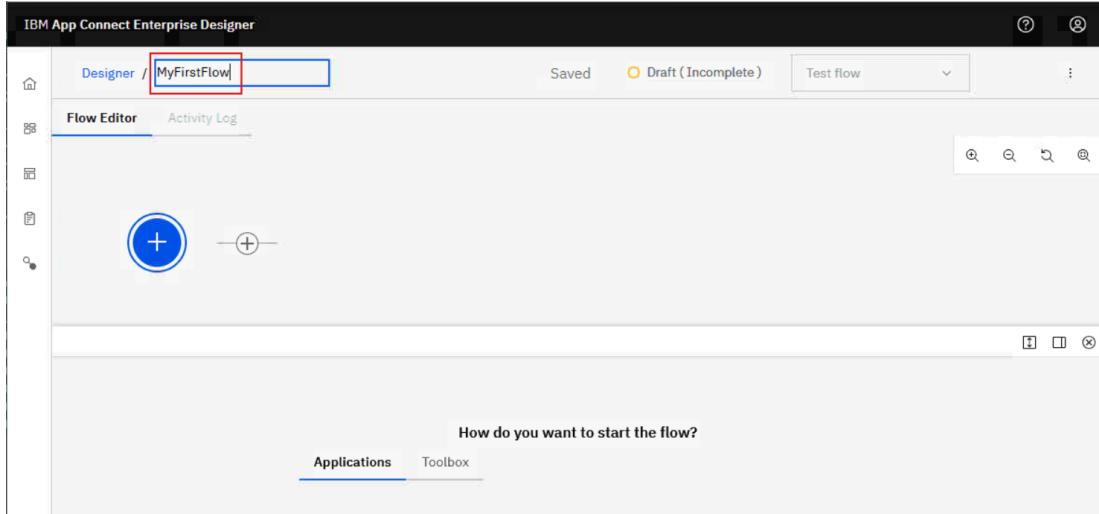
The screenshot shows the IBM App Connect Enterprise Designer interface. On the left, there is a vertical sidebar with the following options: Home, Designer (which is highlighted with a red border), Templates, Logs, and Connect. To the right of the sidebar, the main area has a large title 'IBM App Connect' and a subtitle 'Connect apps and data to your business.' Below the title, there is a search bar with the placeholder 'Search' and a 'Create' button. A tooltip or dropdown menu is visible over the 'Create' button, showing options: 'Event-driven flow' (which is highlighted with a blue border), 'Flows for an API', and 'Import flow...'. The top of the screen shows a browser header with the URL 'localhost:7500'.

4.2 Create Event-driven flow

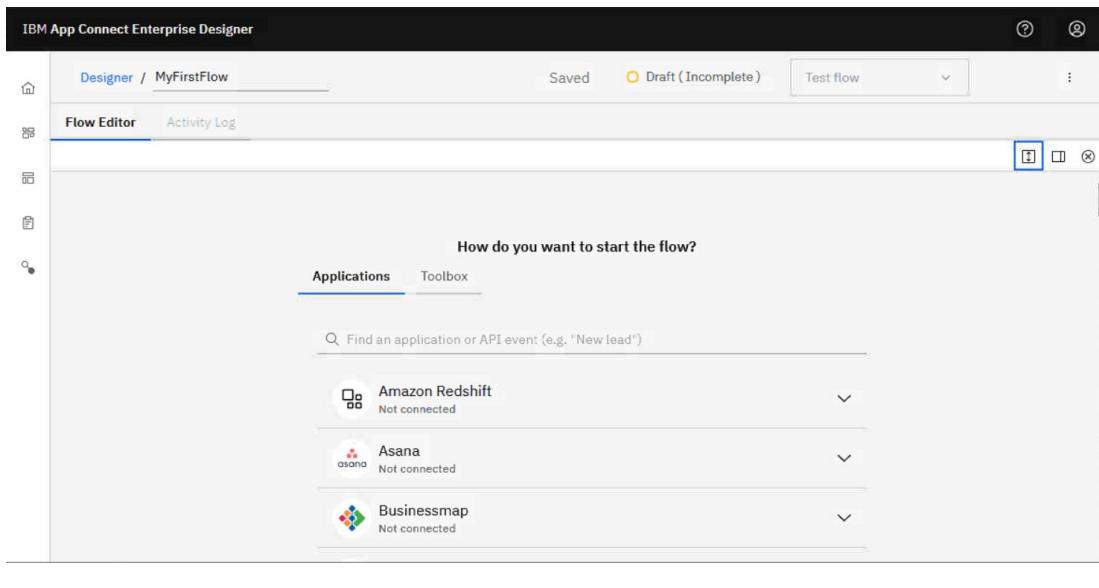
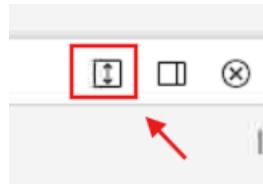
56. Click the **Create** button and choose **Event-driven flow**:

This screenshot is similar to the previous one, showing the Designer interface. The 'Create' button in the main area has a tooltip/dropdown menu open. The 'Event-driven flow' option is highlighted with a blue border, indicating it is the selected choice. The other options in the dropdown are 'Flows for an API' and 'Import flow...'. The rest of the interface, including the sidebar and the main content area, remains the same.

57. Specify a name of **MyFirstFlow** using the box in the top left corner:



58. On the right side of the page, click the maximize panel icon so that you can more easily see the list of applications that can use to start your event driven flow:



4.3 Configure Jira Trigger Event

59. In the search bar location marked below, type in **Jira** as a filter:

How do you want to start the flow?

Applications Toolbox

Q Find an application or API event (e.g. "New lead")

Amazon Redshift
Not connected

Asana
Not connected

Businessmap
Not connected

Callable flow

ClickSend
Not connected

CMIS
Not connected

Coupa
Not connected

Eventbrite
Not connected

The display will update and you should click **Configure more events...**:

How do you want to start the flow?

Applications Toolbox

Q Jira

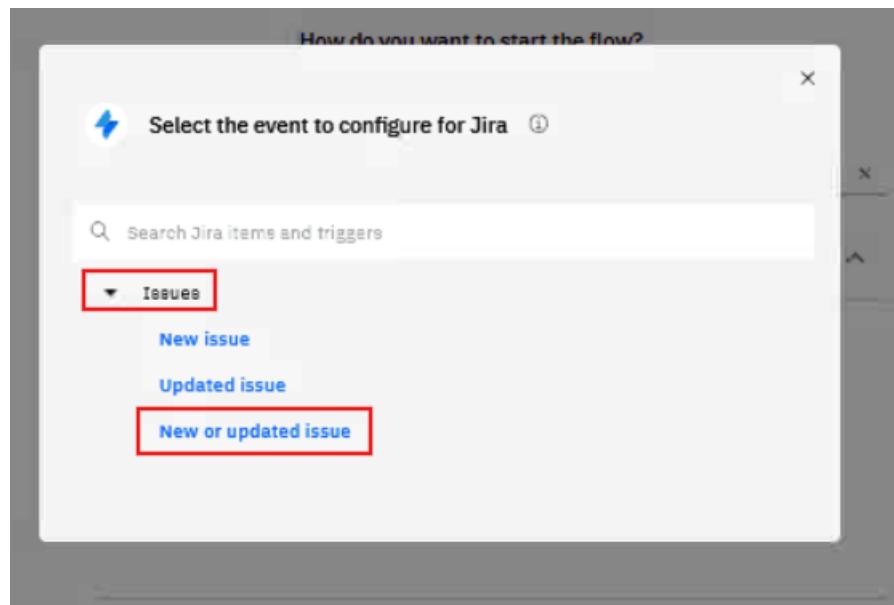
Jira
Not connected

Jira is a proprietary issue tracking product developed by Atlassian which allows bug tracking and agile project management.

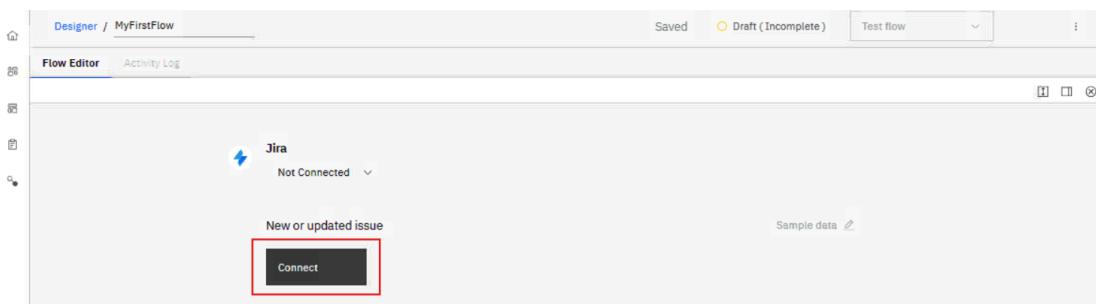
More info

Configure more events...

60. Expand **Issues** and select **New or updated issue**:



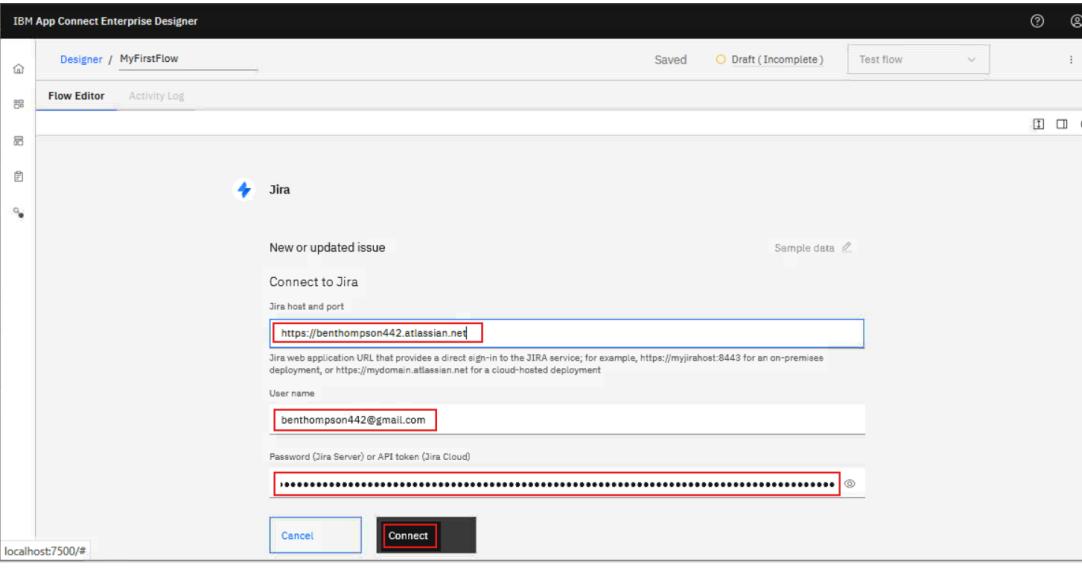
61. Click the **Connect** button:



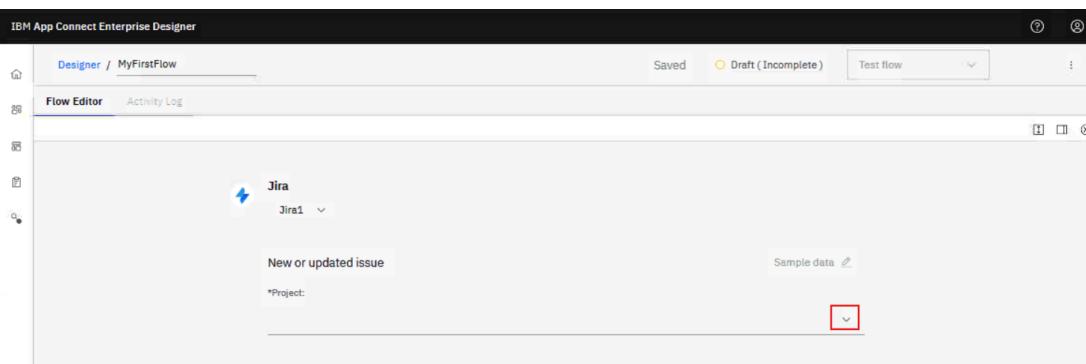
62. Copy the following properties from your text file where you saved the Jira credentials earlier on in the lab:

- Jira host and port: `https://<mydomain>.atlassian.net` (the example earlier used `https://benthompson442.atlassian.net`)
- Username: `<your username>`
- API Token: `<your Jira API Token>`

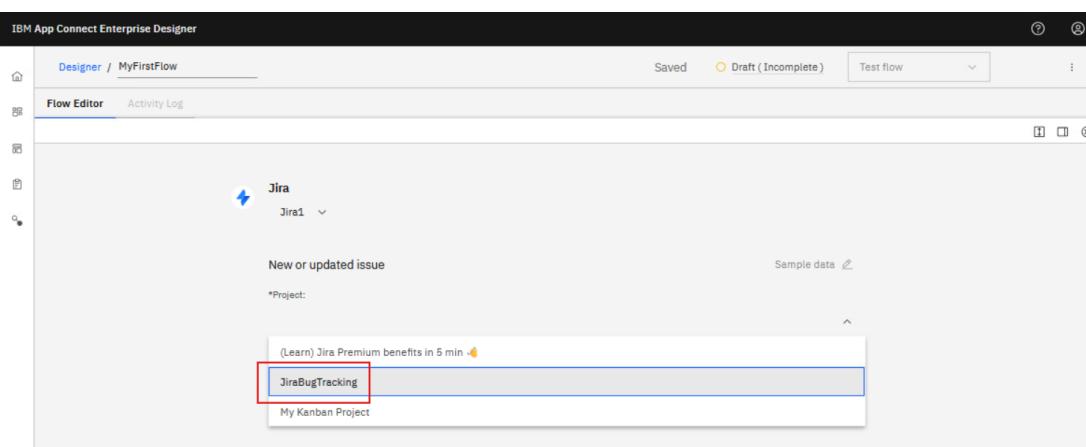
Click **Connect** when completed.



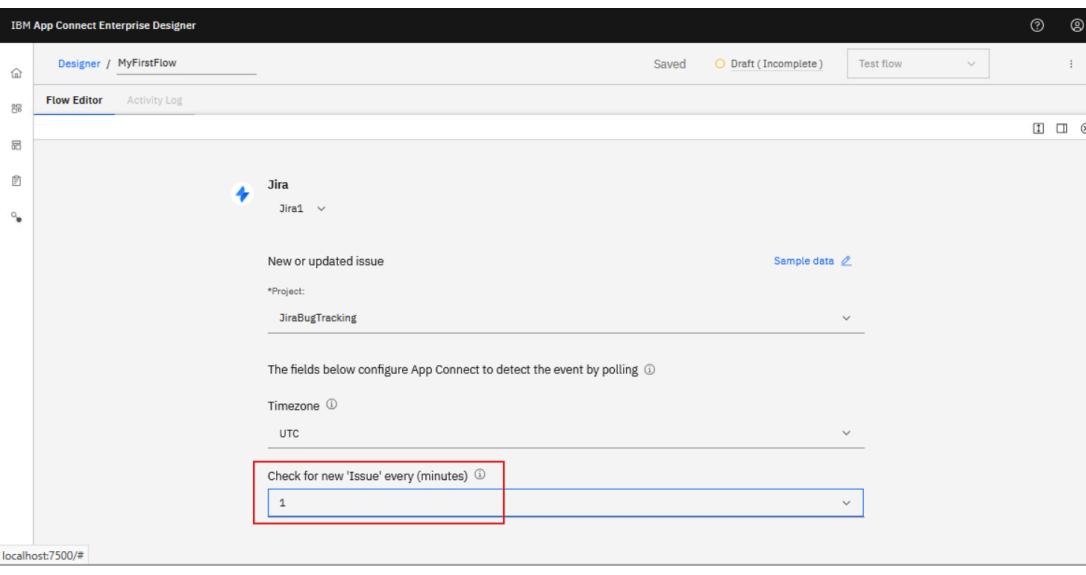
63. After a few seconds, the Designer Flow Editor will connect to Jira and provide a drop down for projects available on your Jira account.



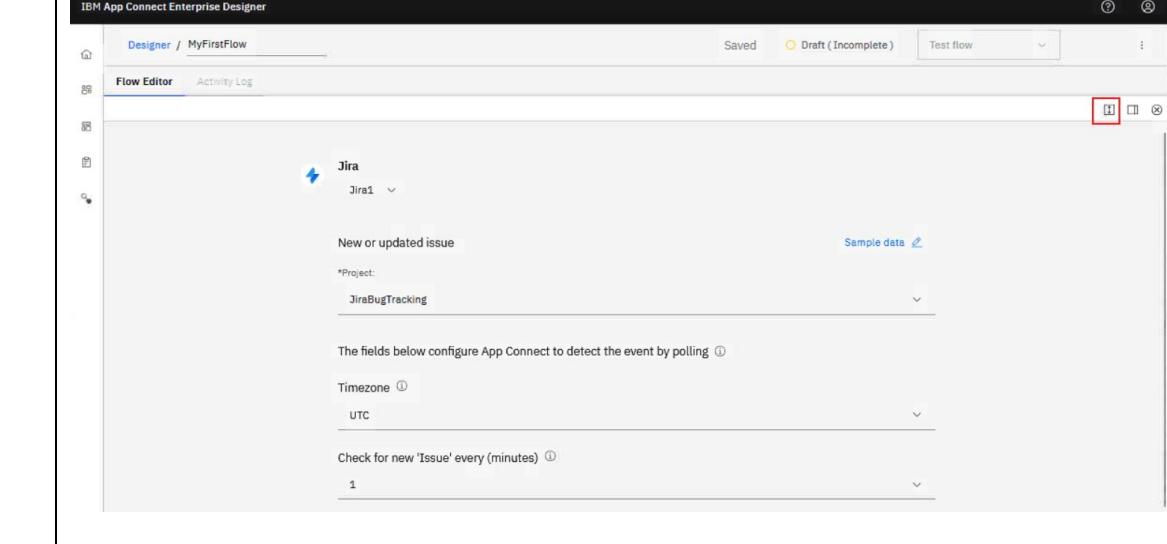
Select the **JiraBugTracking** project where you created issues earlier:



64. Change the frequency of checking for new or updated issues to be every minute:

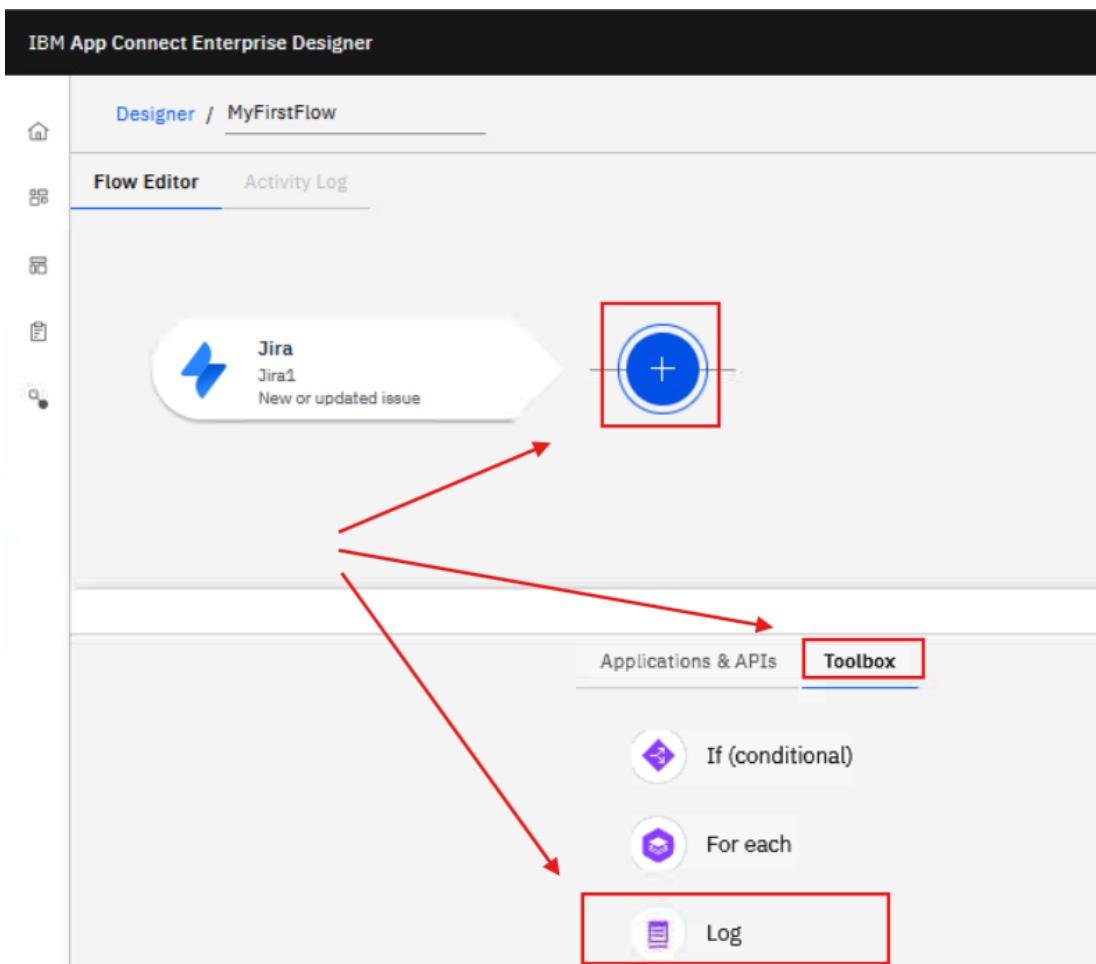


65. Click the **minimize panel** button to see the whole flow view again:

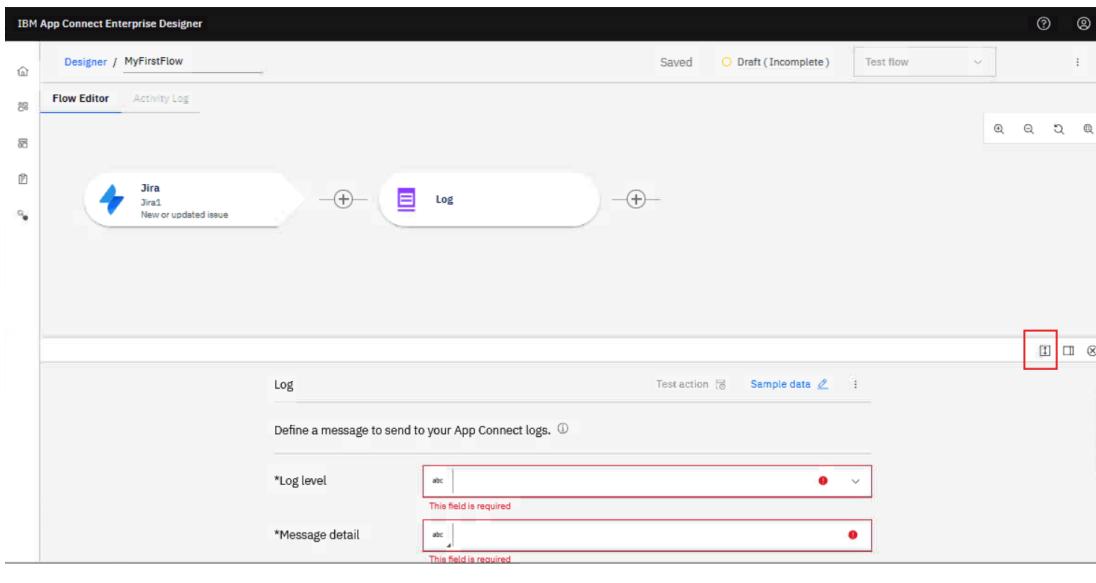


4.4 Configure Log Action

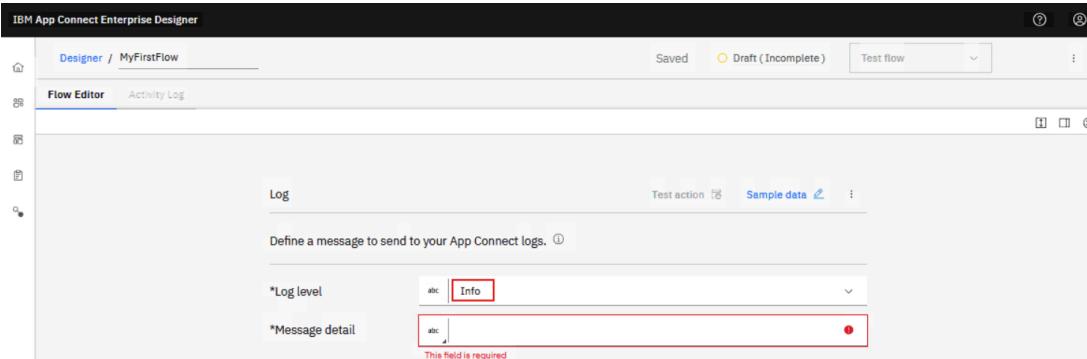
66. In the Designer Flow Editor, click the **plus sign** icon and select the **Toolbox** tab. Scroll down and select **Log**:



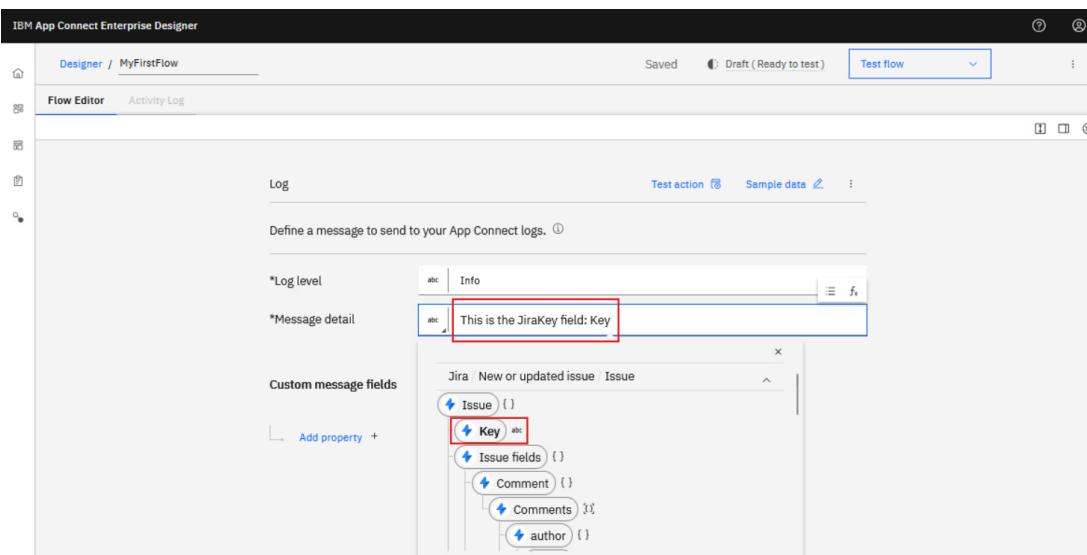
67. The **Log** node will be added to the flow. Click the maximize panel button again so that you can see the configuration window for the **Log** node more easily:



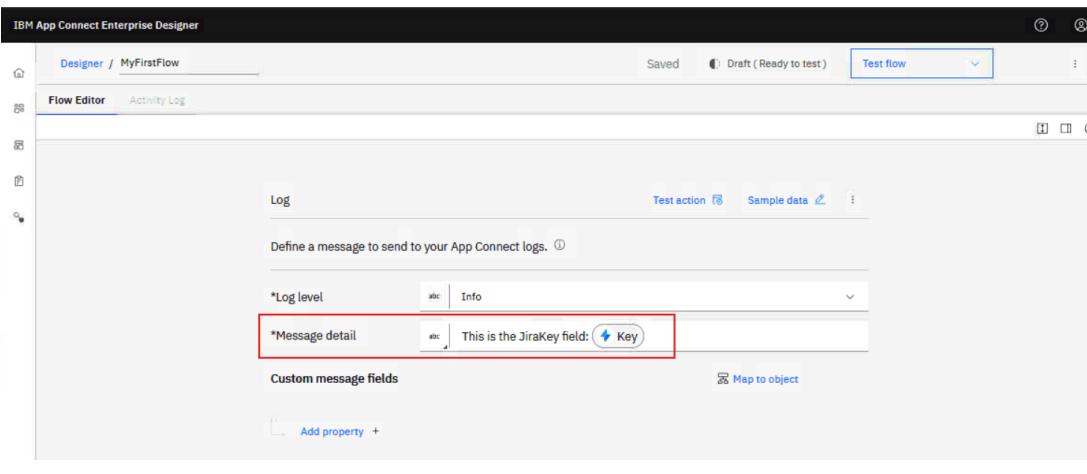
68. In the Log action, set the **Log level** to **Info**:



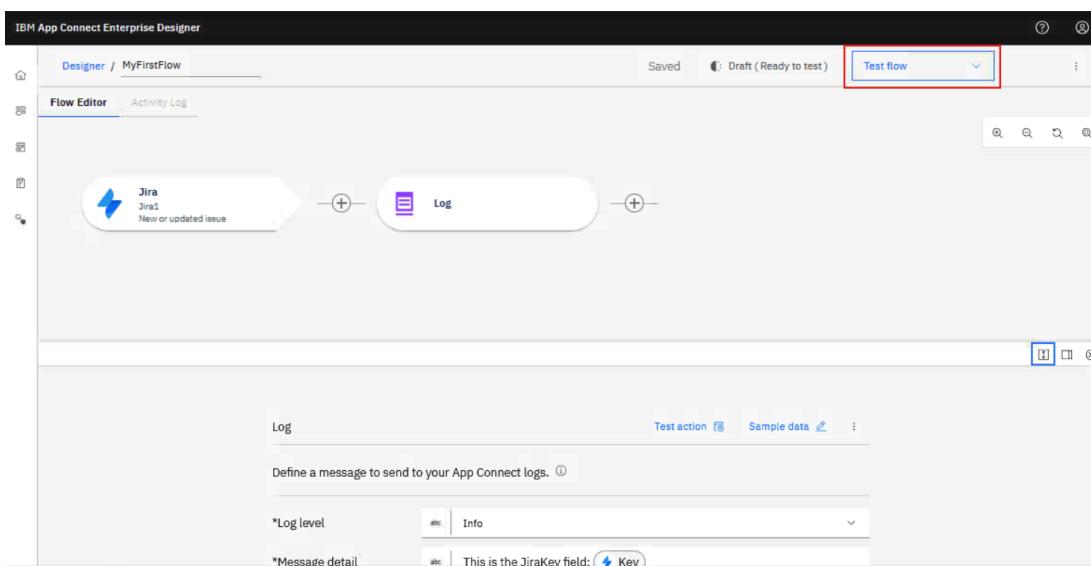
69. In the Message detail field, type the following text: This is the JiraKey field: followed by the word Key which will cause the auto completion assist feature to kick in. Select Key from the hierarchy (underneath the parent field called Issue):



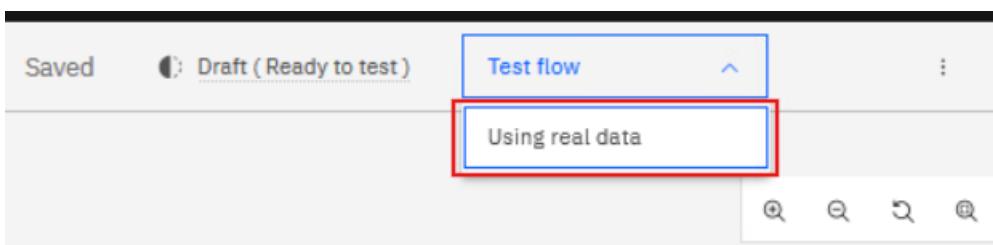
Once you've selected the **Key** field the Message detail field as a whole should look like this:



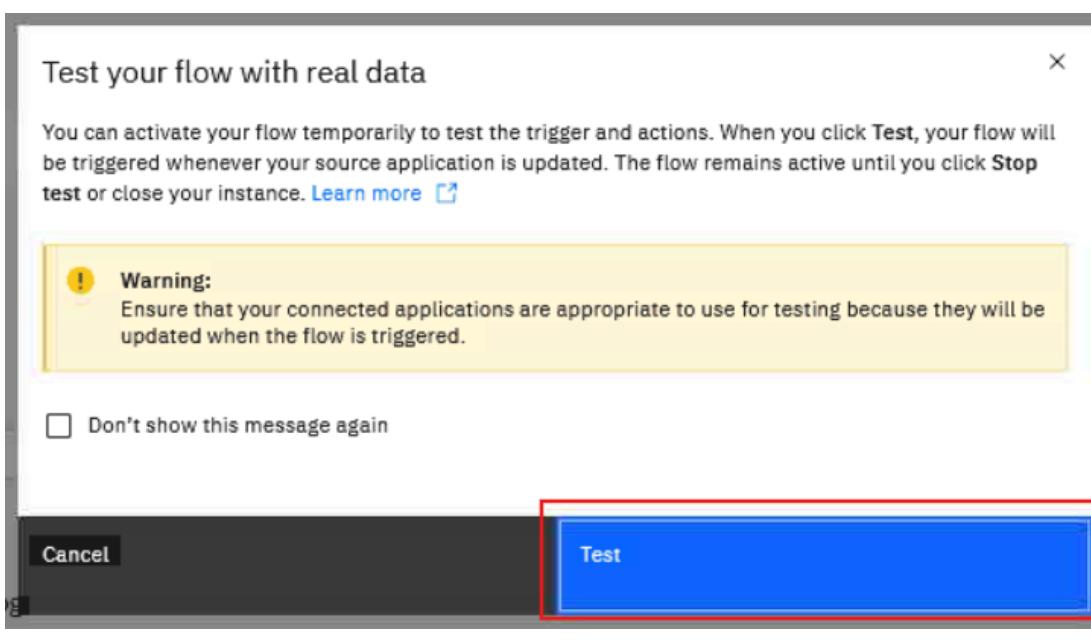
70. Click the minimise panel icon again and you will see the two nodes in the flow have been configured. Click the **Test flow** button:



When the button expands, click **Using real data**:



71. A popup window will be presented with a warning. Click **Test**:



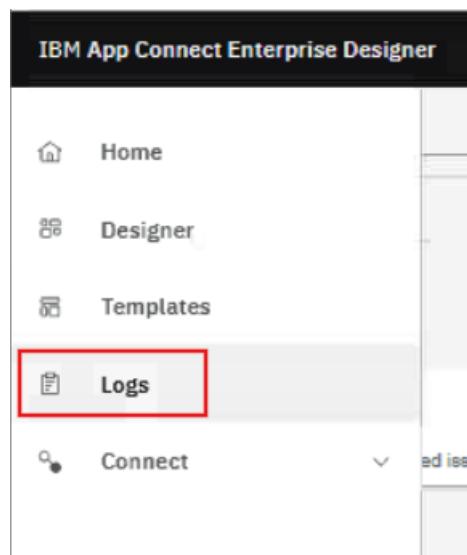
72. The flow will now show as **Testing**:



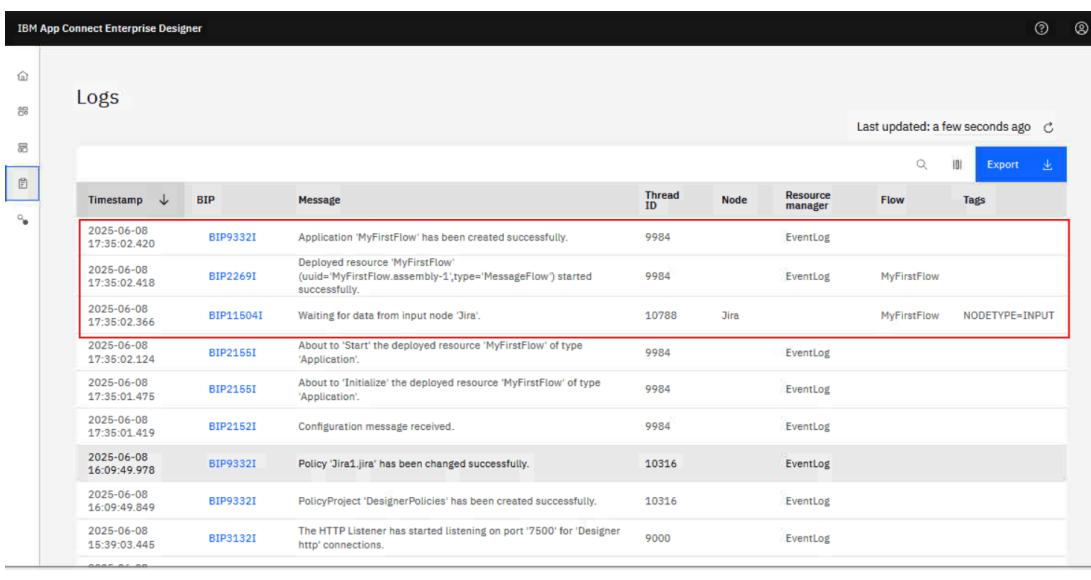
73. If you switch back to the window that came into view (that looks like command console) when launching the Designer, you will see some basic feedback about the items you have been creating. Entries were made when the PolicyProject and Jira policy were created when you clicked on the Connect button earlier. There will also be an entry corresponding to the flow now being started up (in Testing state):

```
Enter an App Connect Enterprise command e.g.
ACE toolkit, ACE help
2025-06-08 16:38:27.590310: BIP9375I: The template 'designer.conf.yaml' has been copied into the workpath at 'C:\Users\aceadmin\IBM\ACEDesigner13\config'.
2025-06-08 16:38:27.684938: BIP10224I: Designer starting initialization; version '13.0.3.0' (64-bit)
2025-06-08 16:38:27.744402: BIP9905I: Initializing resource managers.
2025-06-08 16:38:40.994816: BIP9906I: Reading deployed resources.
2025-06-08 16:39:02.309230: BIP2866I: IBM App Connect Enterprise administration security is inactive.
2025-06-08 16:39:02.309332: BIP10225I: Designer has finished initialization.
2025-06-08 16:39:03.442928: BIP3132I: The HTTP Listener has started listening on port '7500' for 'Designer http' connect ions.
2025-06-08 17:09:49.847100: BIP9332I: PolicyProject 'DesignerPolicies' has been created successfully.
2025-06-08 17:09:49.977080: BIP9332I: Policy 'Jiral.jira' has been changed successfully.
2025-06-08 18:35:01.407580: BIP2152I: Configuration message received.
2025-06-08 18:35:01.472619: BIP2155I: About to 'Initialize' the deployed resource 'MyFirstFlow' of type 'Application'.
2025-06-08 18:35:02.121029: BIP2155I: About to 'Start' the deployed resource 'MyFirstFlow' of type 'Application'.
2025-06-08 18:35:02.164156: BIP2269I: Deployed resource 'MyFirstFlow' (uuid='MyFirstFlow.assembly-1',type='MessageFlow') started successfully.
2025-06-08 18:35:02.419065: BIP9332I: Application 'MyFirstFlow' has been created successfully.
```

74. Back in the web browser tab for Designer, use the left side menu to switch to the **Logs** view:



75. The **Logs** view will show that **MyFirstFlow** is started and is waiting for input from Jira which will trigger it to take an action:

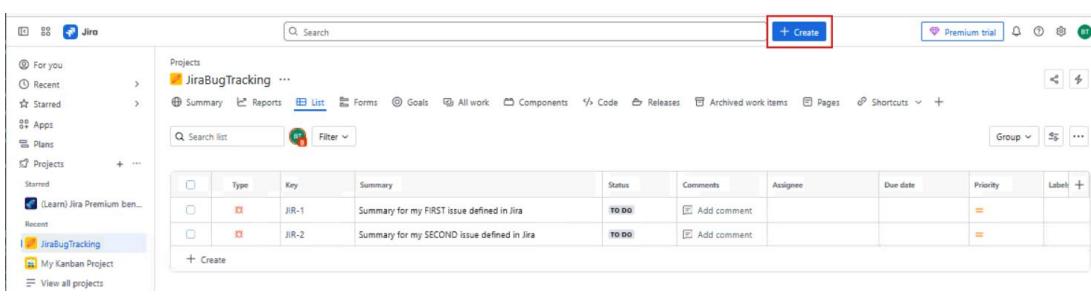


The screenshot shows the 'Logs' view in the IBM App Connect Enterprise Designer interface. The table has columns: Timestamp, BIP, Message, Thread ID, Node, Resource manager, Flow, and Tags. A red box highlights the third log entry, which is: '2025-06-08 17:35:02.366 BIP11504I Waiting for data from input node 'Jira''. This indicates that the flow is waiting for data from the Jira node.

Timestamp	BIP	Message	Thread ID	Node	Resource manager	Flow	Tags
2025-06-08 17:35:02.420	BIP9332I	Application 'MyFirstFlow' has been created successfully.	9984		EventLog		
2025-06-08 17:35:02.418	BIP2269I	Deployed resource 'MyFirstFlow' (uuid='MyFirstFlow_assembly-1',type='MessageFlow') started successfully.	9984		EventLog	MyFirstFlow	
2025-06-08 17:35:02.366	BIP11504I	Waiting for data from input node 'Jira'.	10788	Jira		MyFirstFlow	NODETYPE=INPUT
2025-06-08 17:35:02.124	BIP2155I	About to 'Start' the deployed resource 'MyFirstFlow' of type 'Application'.	9984		EventLog		
2025-06-08 17:35:01.475	BIP2155I	About to 'Initialize' the deployed resource 'MyFirstFlow' of type 'Application'.	9984		EventLog		
2025-06-08 17:35:01.419	BIP2152I	Configuration message received.	9984		EventLog		
2025-06-08 16:09:49.978	BIP9332I	Policy 'Jira1.jira' has been changed successfully.	10316		EventLog		
2025-06-08 16:09:49.849	BIP9332I	PolicyProject 'DesignerPolicies' has been created successfully.	10316		EventLog		
2025-06-08 15:39:03.445	BIP93132I	The HTTP Listener has started listening on port '7500' for 'Designer http' connections.	9000		EventLog		

4.5 Testing

76. In your web browser, return to the Jira cloud environment where we will create a new issue. Click the **Create** button



The screenshot shows the Jira cloud interface. On the left, there's a sidebar with 'For you', 'Recent', 'Starred', 'Apps', 'Plans', and 'Projects'. Under 'Projects', 'JiraBugTracking' is selected. The main area shows a table of issues with columns: Type, Key, Summary, Status, Comments, Assignee, Due date, Priority, and Labels. Two issues are listed: JIR-1 and JIR-2. A red box highlights the '+ Create' button at the bottom left of the issue table.

77. Repeat the same process we used earlier to create a third issue. In the **Create issue** popup window, select the following:

- **Issue type** = Bug
- **Summary** = Summary for my THIRD Issue defined in Jira
- Scroll down and set the **Description** to: Description of my THIRD Issue in Jira.

Leave the defaults for the other fields and click the **Create** button:

The screenshot shows the 'Create' dialog box for Jira. The 'Work type' dropdown is set to 'Bug'. The 'Summary' field contains 'Summary for my THIRD issue defined in Jira'. The 'Description' rich text area contains 'Description of my THIRD Issue in Jira'. The 'Create' button is highlighted with a red box.

78. Note that once the issue is created, the summary table will display the key for the new issue (in this example **JIR-3**):

The screenshot shows the Jira software interface with the URL <https://benthompson442.atlassian.net/jira/software/c/projects/JIR/list>. The left sidebar shows projects like 'JiraBugTracking' and 'My Kanban Project'. The main area displays a table of issues with columns: Type, Key, Summary, Status, Comments, Assignee, Due date, Priority, and Label. The third row, labeled 'JIR-3', has its 'Key' column value 'JIR-3' highlighted with a red box.

79. Back in the Designer tab of the web browser, on the **Logs** view, click the refresh icon in the top right corner:

The screenshot shows the ACE Designer Logs view. At the top, it says 'Last updated: 15 minutes ago' with a refresh icon to its right, which is highlighted with a red box. Below this, there are tabs for 'Resource manager', 'Flow', and 'Tags', with 'EventLog' selected. The main area shows a table of logs with columns: Timestamp, BIP, Message, Thread ID, Node, Resource manager, Flow, and Tags. The first log entry's message 'This is the JiraKey field: JIR-3' is highlighted with a red box.

80. The **Logs** view will be refreshed and you will find that the creation of the Jira issue has resulted in the ACE Designer logging the key for the issue as shown below:

The screenshot shows the ACE Designer Logs view after refreshing. It displays a log entry with the timestamp '2025-06-08 18:11:03.783' and the message 'This is the JiraKey field: JIR-3', which is highlighted with a red box. The other log entries are as follows:

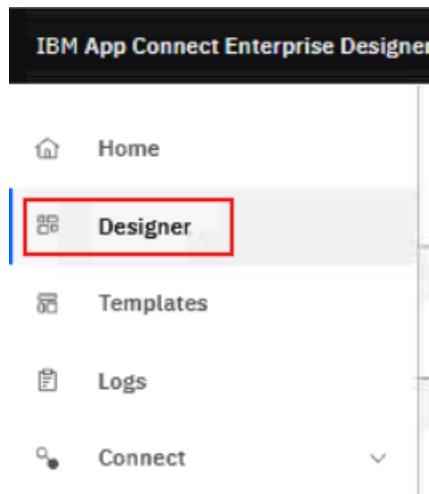
Timestamp	BIP	Message	Thread ID	Node	Resource manager	Flow	Tags
2025-06-08 18:11:03.783	BIP10206I	This is the JiraKey field: JIR-3	10788	Log		MyFirstFlow	NODETYPE=LOG MSG_ID=(00002A24-6845D237-000001)
2025-06-08 18:11:03.783	BIP11504I	Waiting for data from input node 'Jira'.	10788	Jira		MyFirstFlow	NODETYPE=INPUT
2025-06-08 18:11:03.765	BIP13098I	Connector 'Jira' received event 'New or Updated Issue'	10788	Jira		MyFirstFlow	EVENT=CREATEDORUPDATED_POLLER OBJECT=Issue CONNECTOR_TYPE=Jira
2025-06-08 17:35:02.420	BIP9332I	Application 'MyFirstFlow' has been created successfully.	9984		EventLog		

Congratulations on creating your first working Designer flow! In the next section we will further enhance the flow to send an output message to Slack.

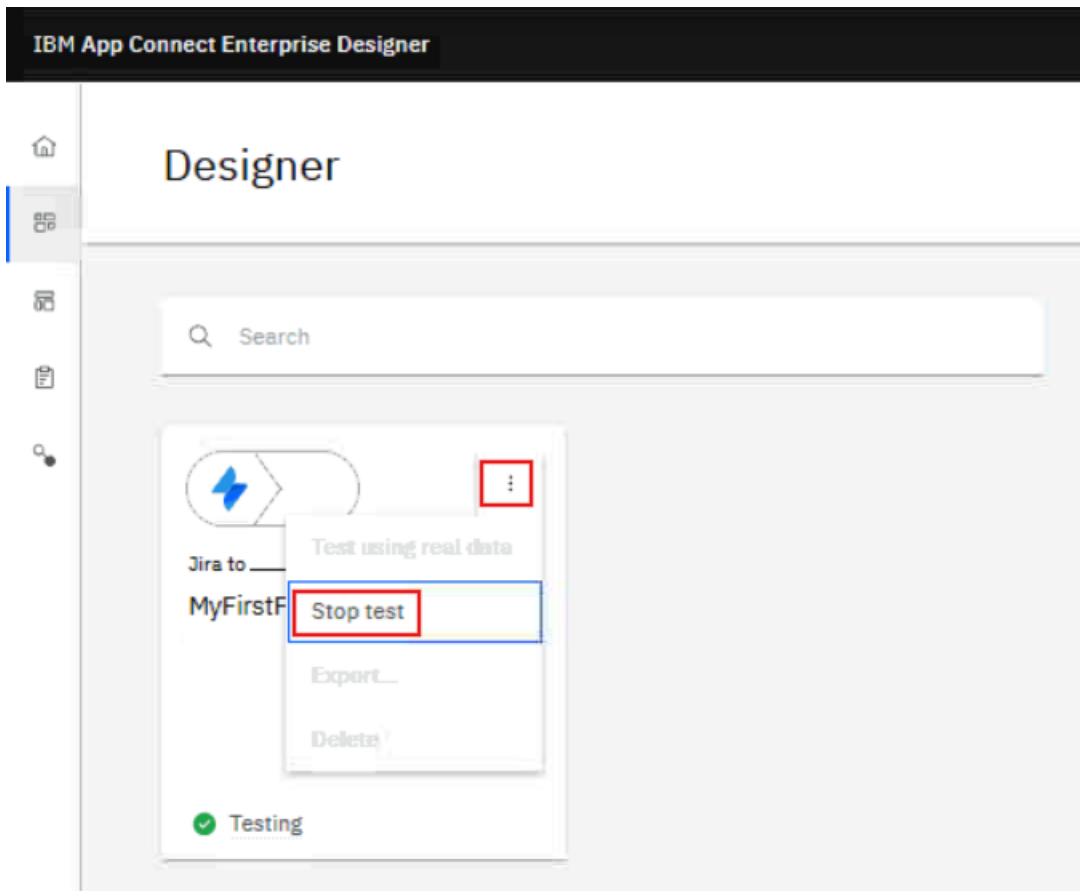
5. Send Jira Issue to Slack

5.1 Test Sending Jira Data to Slack

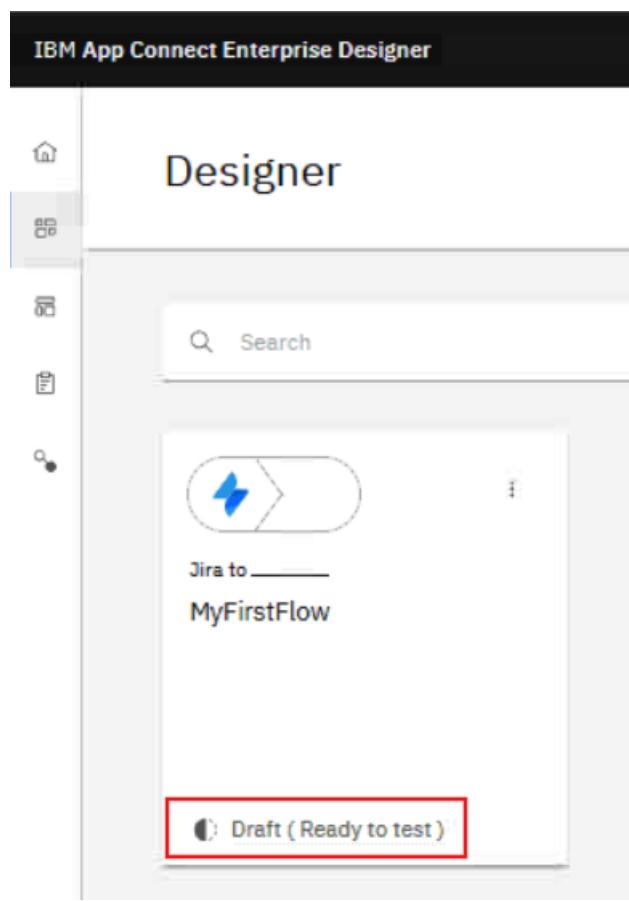
81. In the Designer Flow Editor, switch back from the **Logs** view to the **Designer** view using the left side menu:



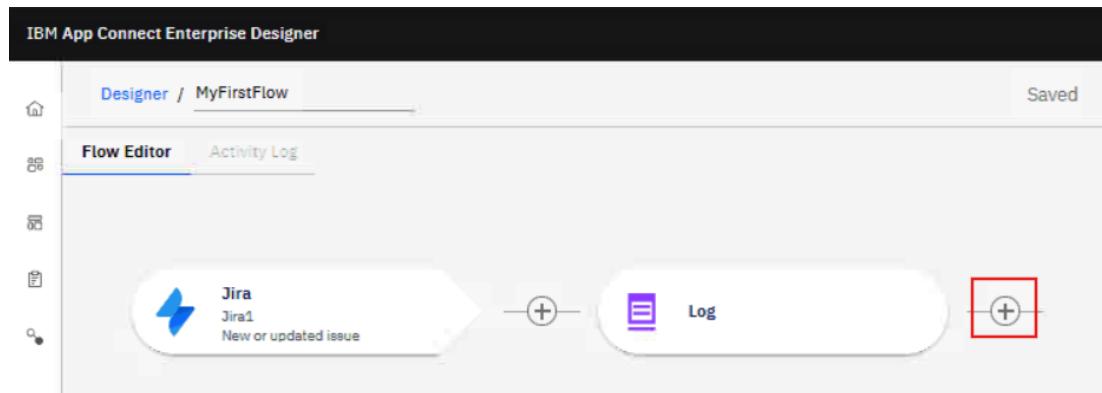
82. In the Designer view, you will find a tile representing **MyFirstFlow** which is currently in **Testing** state. Click the icon in the top right corner of the tile to expand its menu, and choose the only available option to **Stop test**:



83. The status of the flow will be returned to **Draft**. Click the tile so that we can begin to edit it again in the Flow Editor view:



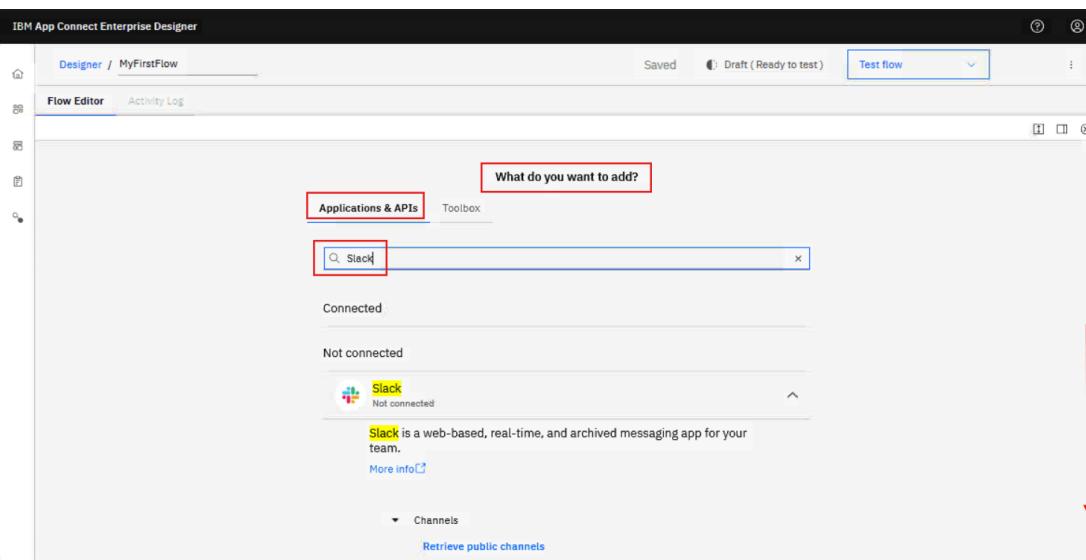
84. In the Designer Flow Editor, click the plus sign to the right of the **Log** action:



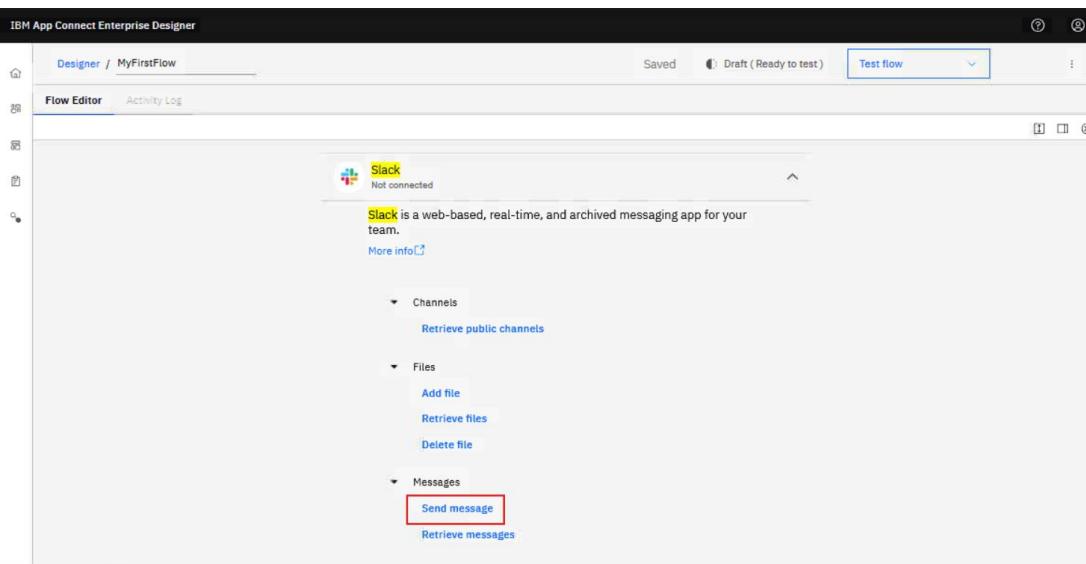
85. Just like we did previously, use the maximise panel button to increase the size of the catalog of options:



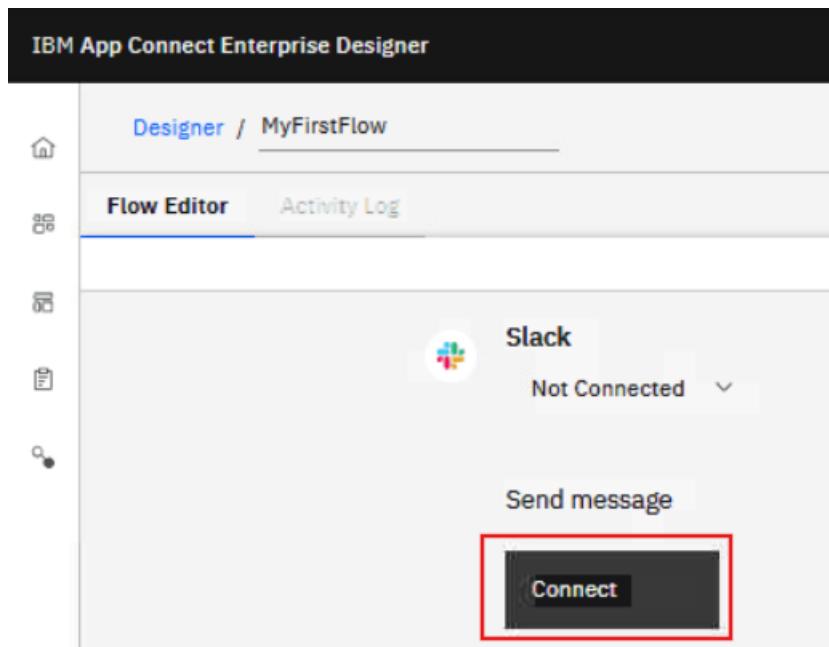
86. In the **What do you want to add** section of the panel, click the **Applications & APIs** tab and search for **Slack**:



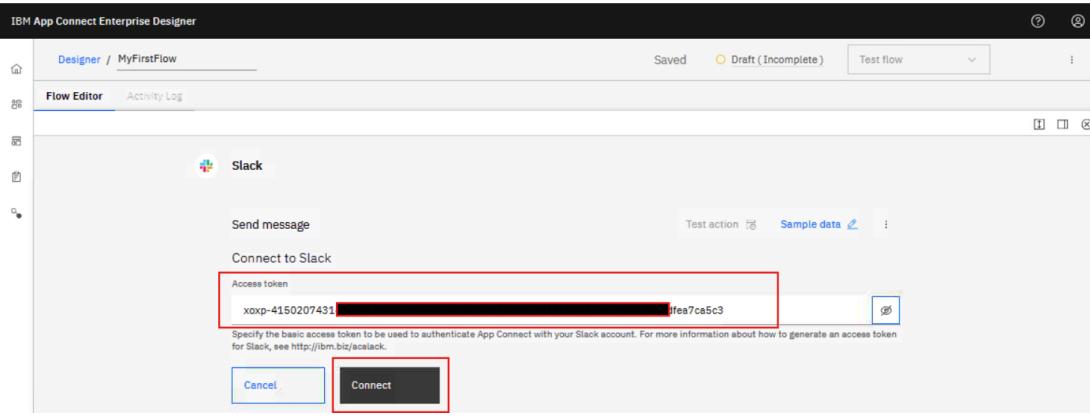
87. Scroll down the list of objects supported and select **Send message**:



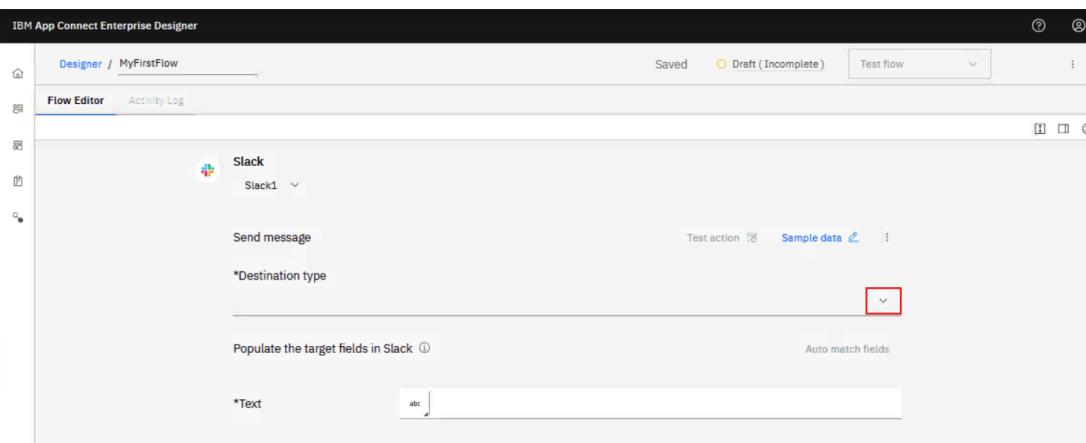
88. Note that the Slack action is **Not Connected**. Click the **Connect** button:



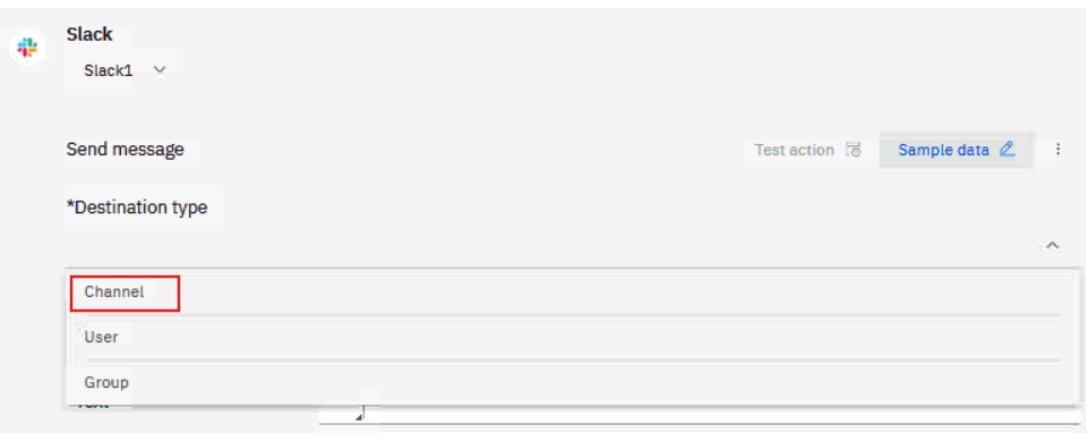
89. Copy the value of your saved Slack Access Token into the **Access token** field, and click **Connect** button:



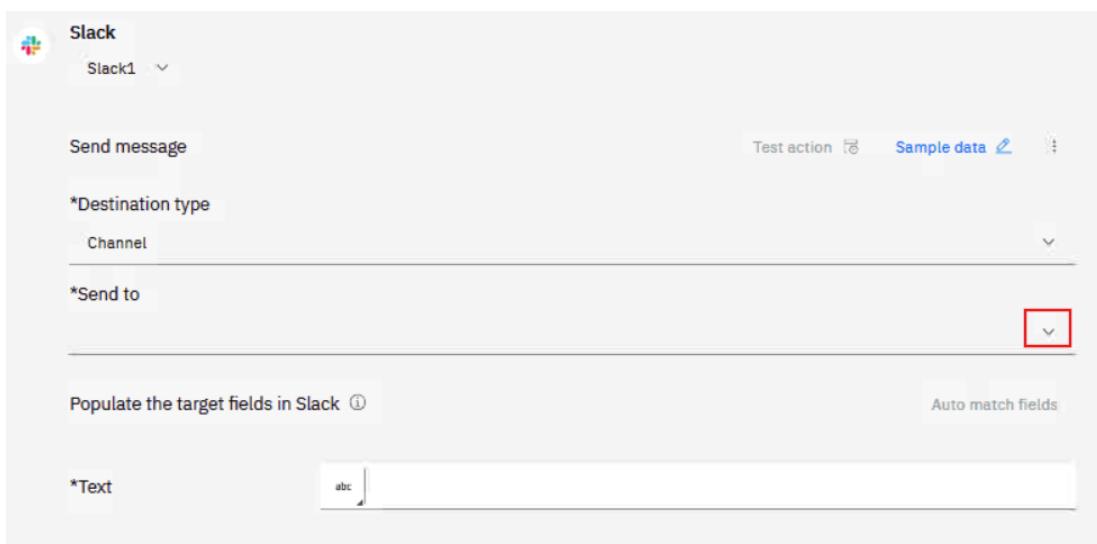
90. After a couple of seconds, the display will refresh. Click the button highlighted below to expand the dropdown menu for **Destination Type**:



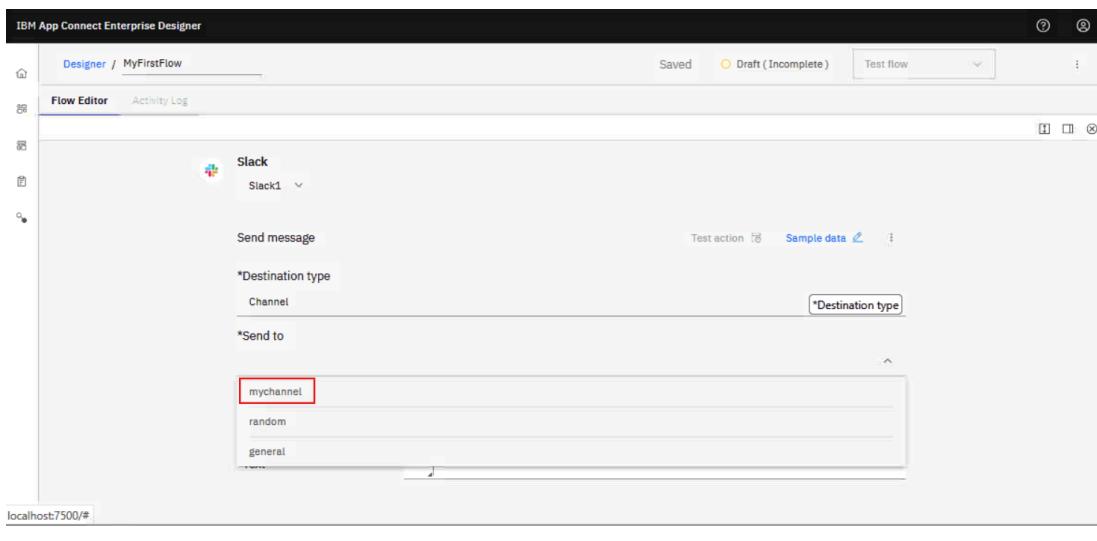
From the available options choose the **Destination type** of Channel:

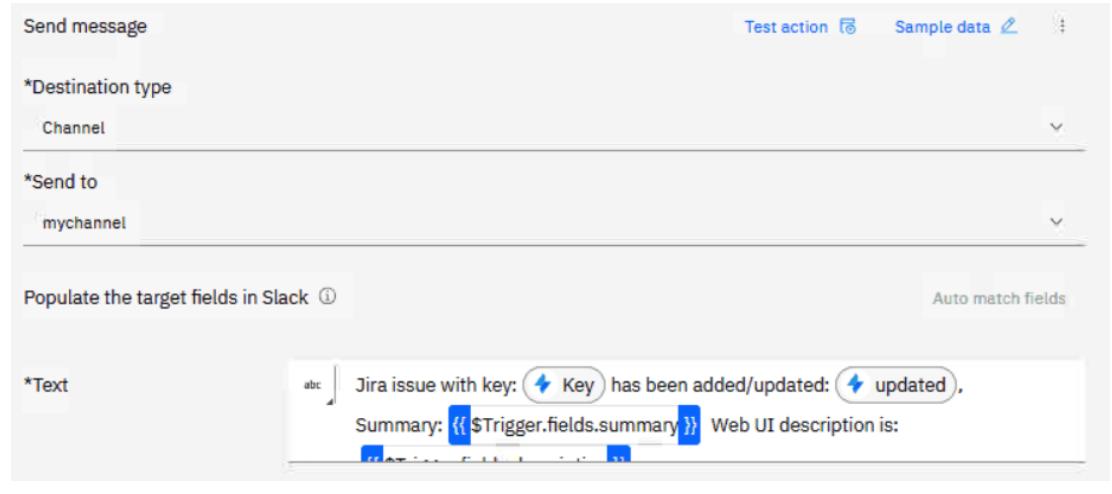
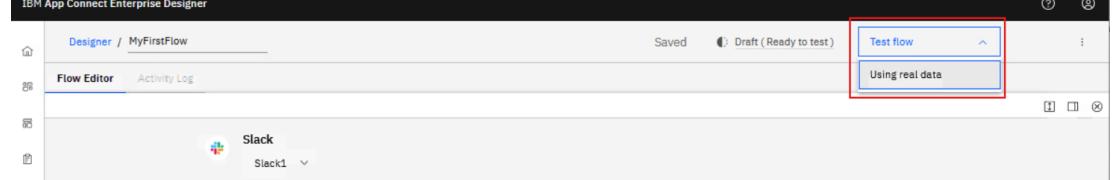
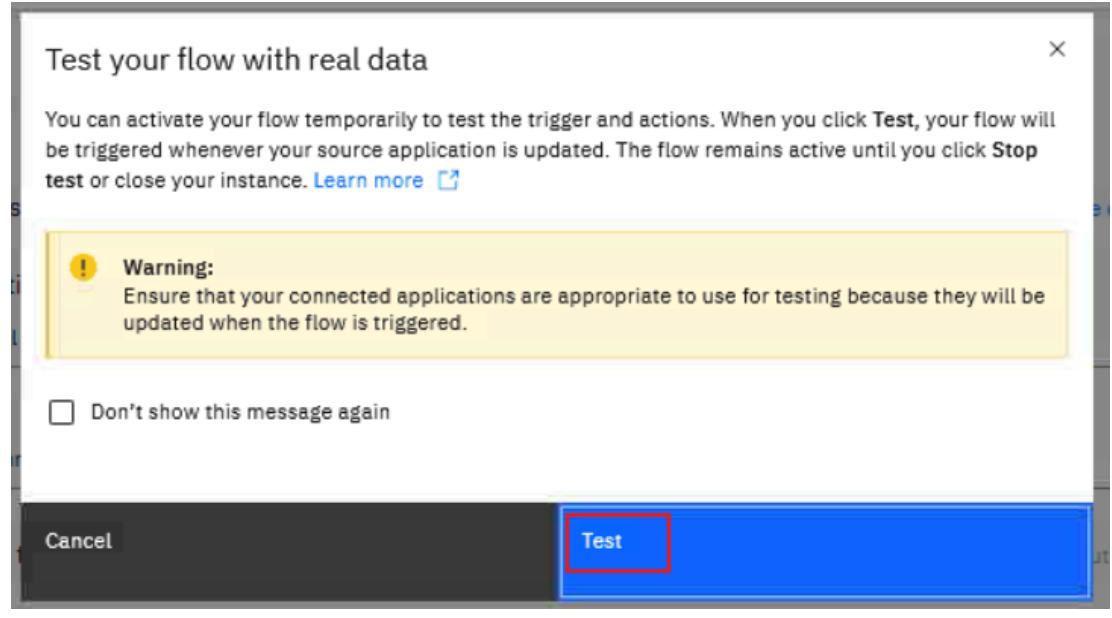


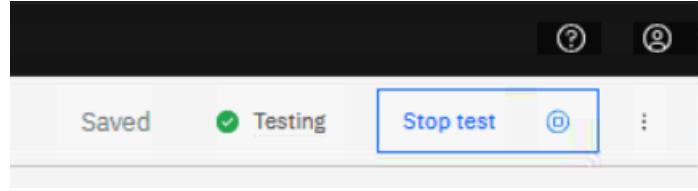
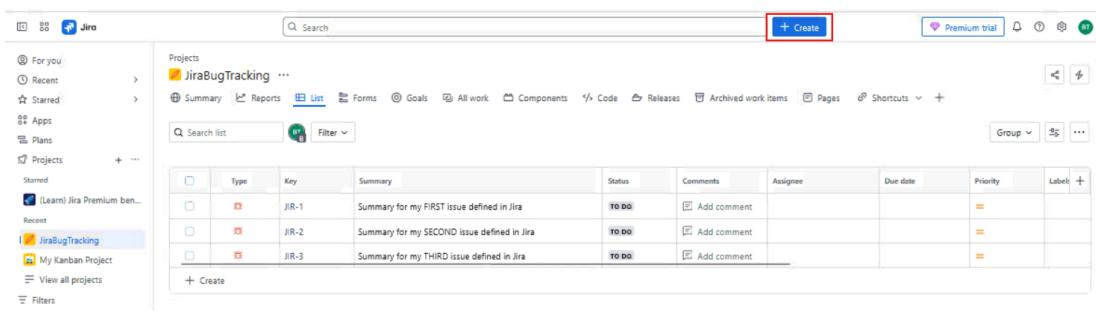
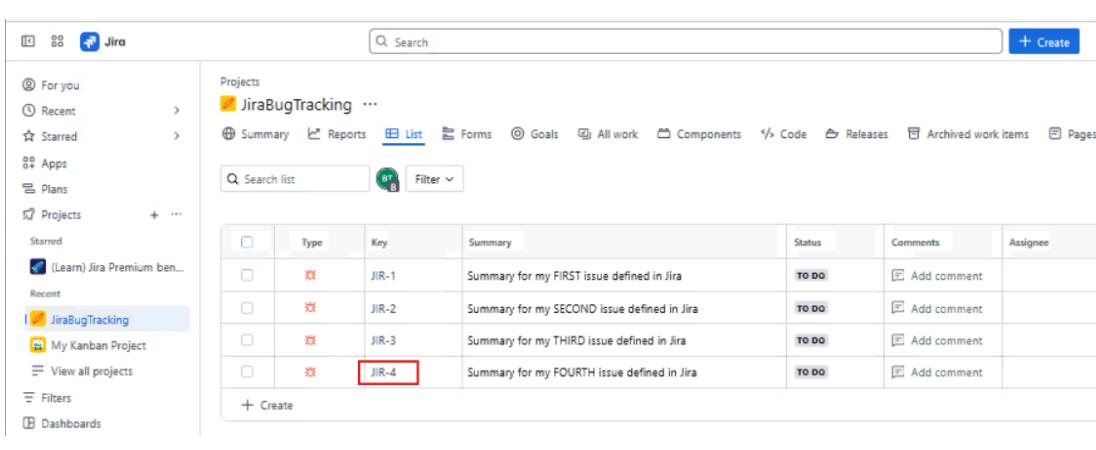
91. The display will update with a **Send to** field. Click the button highlighted to show the available channel options:

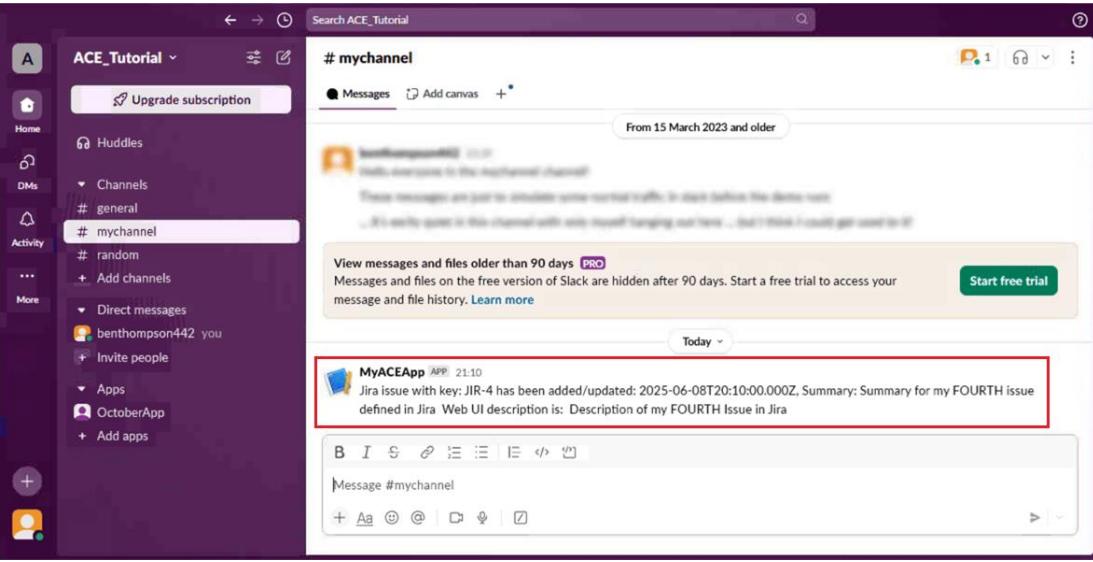
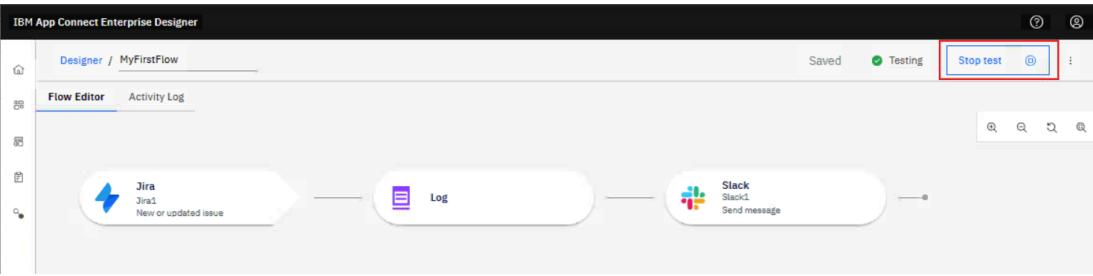
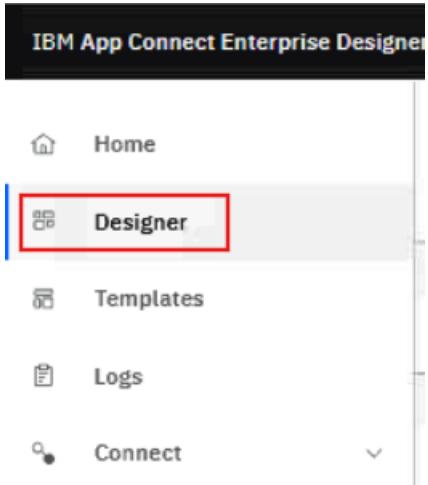


Choose a suitable channel (which one does not really matter). In the case of the Slack workspace we used to create this lab, we went for **mychannel**:

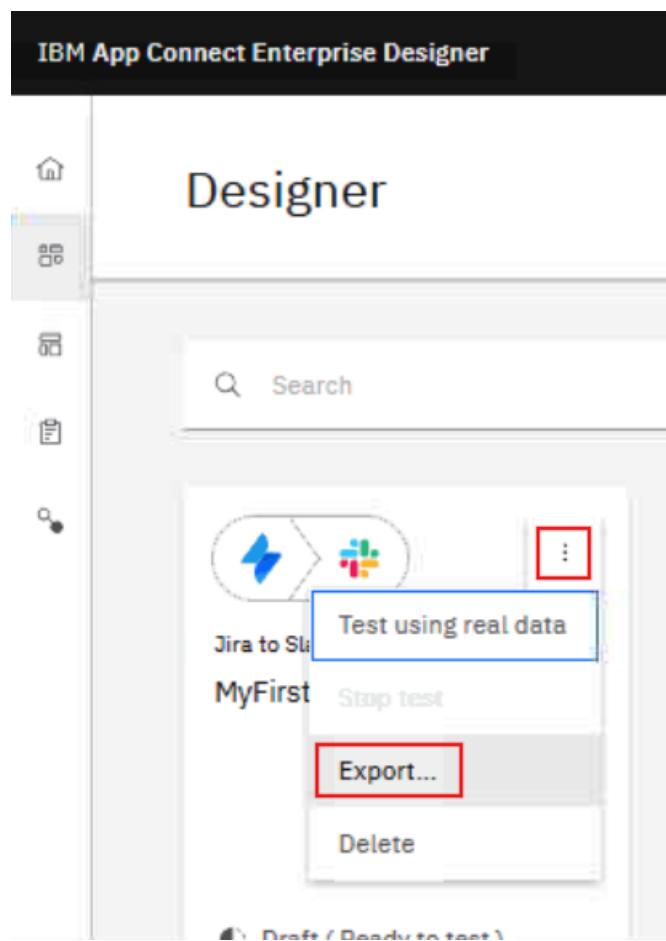


92.	<p>In the Text field, copy the following text into the field. This suggestion includes examples of JSONata expressions which will be interpreted to address fields which will have been populated by the Jira message flow node and are then passed into the Slack node in the message flow. If you prefer, instead of copying this text in verbatim, you can experiment with the available buttons and drop-downs for constructing the Text field contents:</p> <pre>Jira Issue with key: {{\$Trigger.key}} has been added/updated: {{\$Trigger.fields.updated}}, Summary: {{\$Trigger.fields.summary}} Web UI description is: {{\$Trigger.fields.description}}</pre> 
93.	<p>Click the Test flow button and select Using real data:</p> 
94.	<p>Agree to the warning by clicking the Test button:</p> 

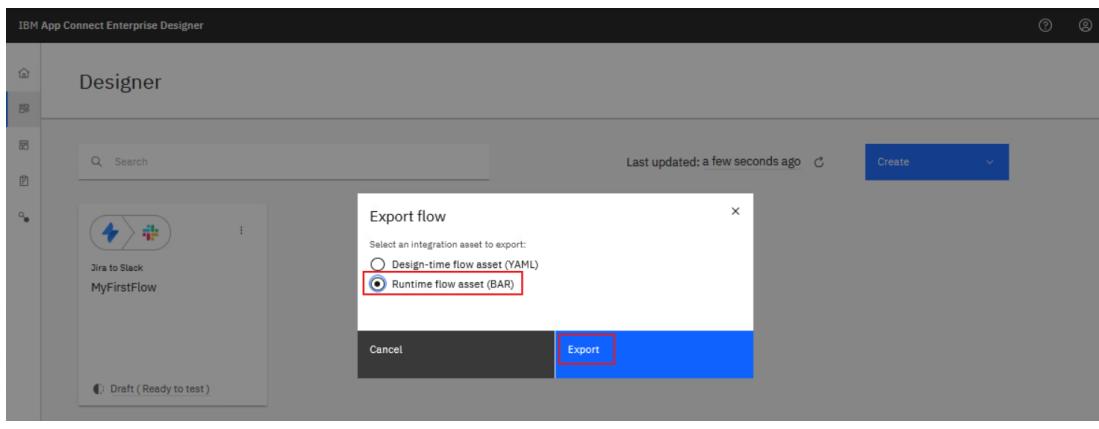
95.	<p>The flow will update to Testing status:</p> 
96.	<p>Return to the Jira web browser tab and click the Create button in order to create a new issue:</p> 
97.	<p>Repeat the same process which we used earlier to create a fourth issue. In the Create issue popup window, use the following details:</p> <ul style="list-style-type: none"> • Issue type = Bug • Summary = Summary for my FOURTH Issue defined in Jira • Scroll down and set the Description to: Description of my FOURTH Issue in Jira. <p>Leave the defaults for the other fields and click the Create button.</p>
98.	<p>When the issue is created, you will see on the List view that it will have been given a new Key (in this example, JIR-4):</p> 

99.	<p>The flow will poll for new Jira issues once per minute, so you may have to wait a few seconds for the update to come through to Slack. In the Slack tab of your web browser, when you navigate to the relevant channel (in this example, mychannel), you should find that a message has been sent from the message flow with the relevant detail that we set up just a moment ago, as shown in the picture below:</p> 
100.	<p>Back in the Designer tab of the web browser, click the Stop test button:</p> 
101.	<p>Navigate back to the Designer view by using the left hand navbar:</p> 

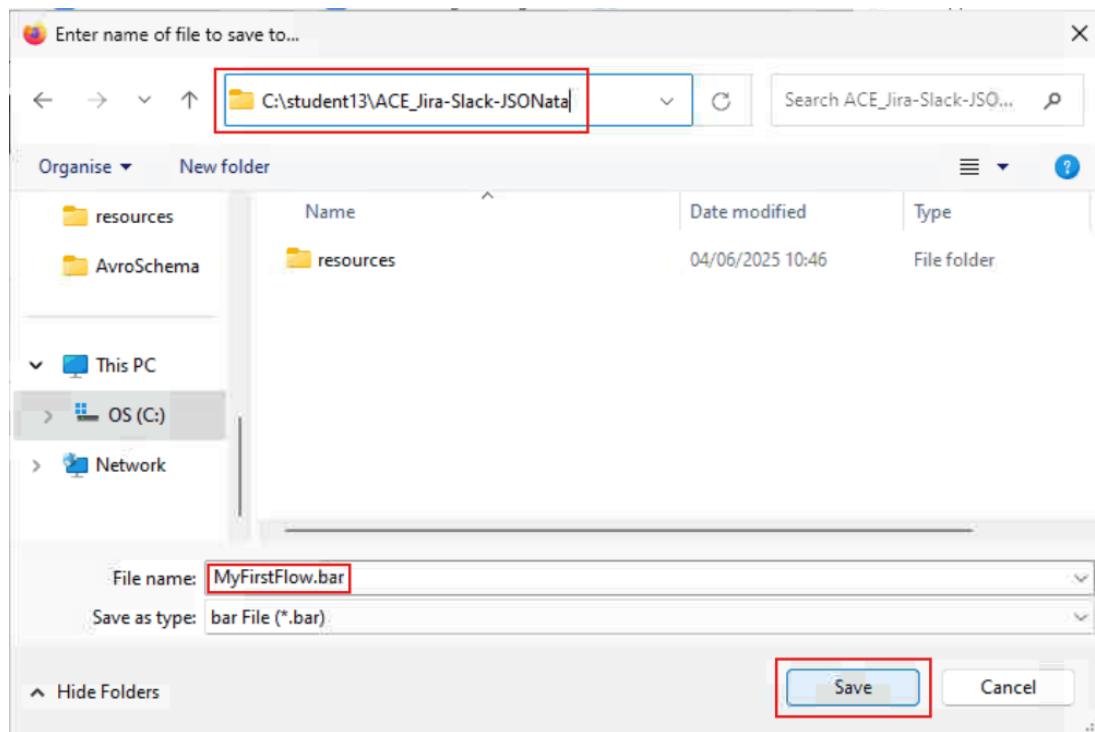
102. With the flow now stopped, you can use the menu in the top right corner of the flow tile to **Export**:



103. From the resulting Export flow dialog, select **Runtime flow asset (BAR)** and click **Export**:

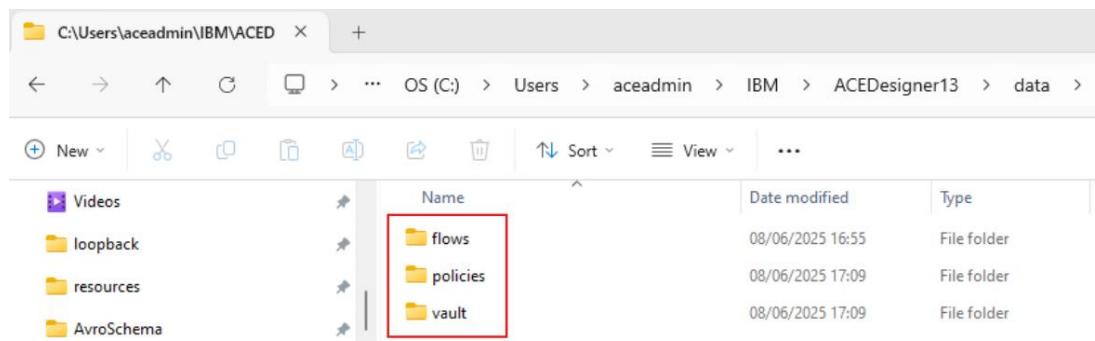


104. Navigate to the C:\student13\ACE_Jira-Slack-JSONata directory:



105. For readers who are familiar with ACE and the Toolkit, you might like to have a peek inside the resulting BAR file. This export mechanism provides a BAR which includes a flow definition (*inside an .appzip file*). The policy and credentials used by the flow are not placed in the BAR file, but they are available on the local disk system which underpins the Designer installation.

For the inquisitive, you can use Windows Explorer and navigate to the directory C:\Users\aceadmin\IBM\ACEDesigner13\data and you will find a flows subdirectory, a policies subdirectory and a vault:



This file system could be integrated with source version control and pipelines, and potentially in future we also expect further product mechanisms to help move this configuration into other runtime contexts.

END OF LAB GUIDE