IBM Cloud Pak for Business Automation Demos and Labs 2025

Consume & Publish Automation Services in IBM Business Automation Workflow

Jorge D. Rodríguez
jorgedr@us.ibm.com

Swapnil Agrawal
aswapnil@ca.ibm.com

V 1.0 (for CP4BA 25.0.0)

NOTICES

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

TRADEMARKS

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a Registered Trade Mark of AXELOS Limited.

ITIL is a Registered Trade Mark of AXELOS Limited.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Table of Contents

1 Introduction

1.1 IBM Business Automation Workflow

IBM Business Automation Workflow is software that combines business process management and case management capabilities in a single integrated workflow solution. It unites information, process, and users to provide a 360-degree view of work to help drive more successful business outcomes.

Additional information about IBM Business Automation Workflow can be found here.

1.2 Lab Overview

In this lab, you will learn how to work with automation services and external services.

Automation services provide a unified way to leverage services in the IBM Cloud Pak for Business Automation platform. Capabilities such as Decisions & Workflow can expose automation services to be consumed throughout the platform.

External services are used to call an application or a service that is external to IBM Business Automation Workflow. For example, you can create an external service to call a Java application that sends out emails.

As a part of this lab, you will consume an automation service published by the Decision capability to scoreboard (perform risk assessment and classification) a client. You will then create an external service that invokes a Java application that sends out emails. Finally, you will see how to publish the external service as an automation service so that the email capability can be leveraged by others in the platform.

Approximate Duration: 2 hours

1.3 Lab Setup Instructions

1. If you are performing this lab as a part of an IBM event, access the document that lists the available systems and URLs along with login instructions. For this lab, you will need to access **IBM Business Automation Studio**.
2. Download the **mailIntegration.jar** from the **Lab Data** folder onto your computer. This file contains the Java implementation to send an email.

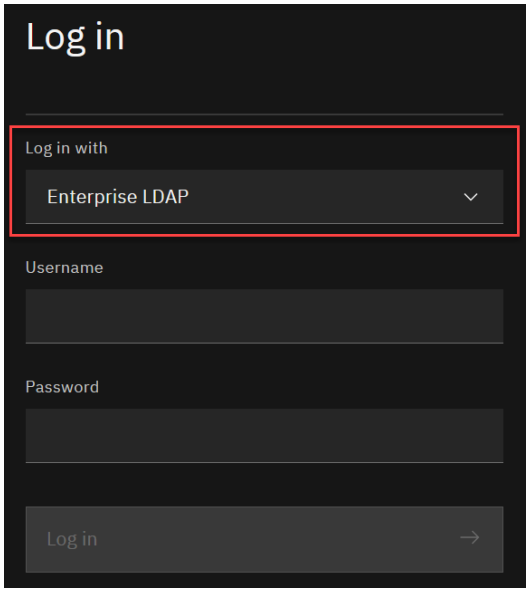
2 Exercise: Consume an Automation Service

2.1 Introduction

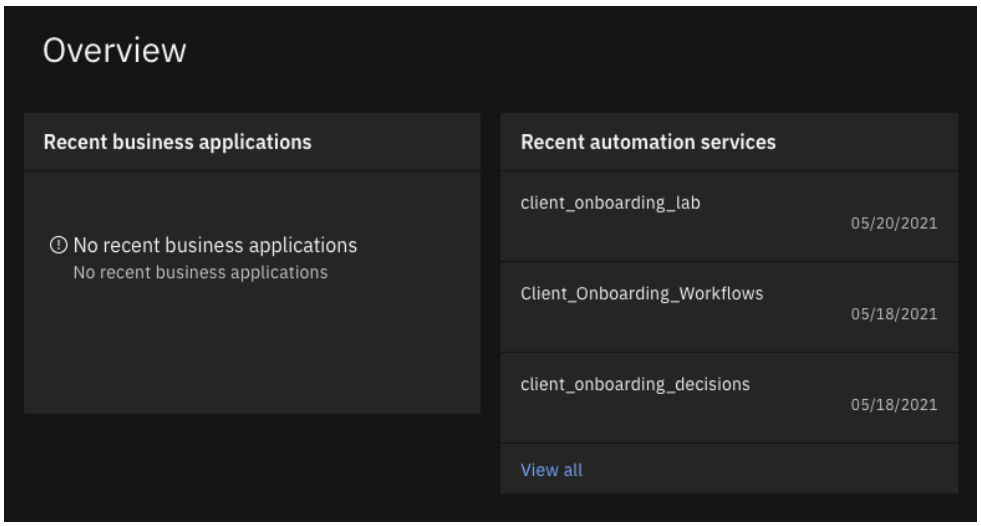
In this exercise, we will consume an automation service that is published using the IBM Automation Decision Service capability. This automation service invokes a decision that scoreboards a client i.e., gives an artificial intelligence backed risk assessment and classifies the client as Segment 1 or 2.

2.2 Exercise Instructions

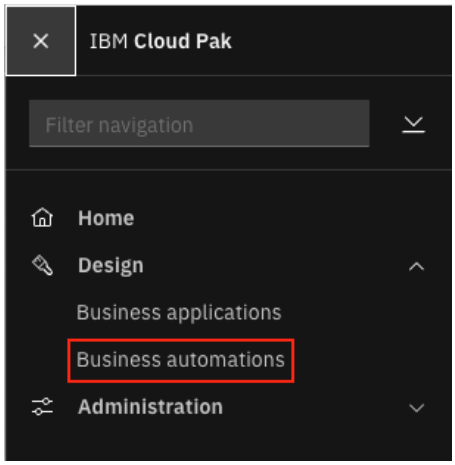
In your browser, login to IBM Business Automation Studio using the Enterprise LDAP option.



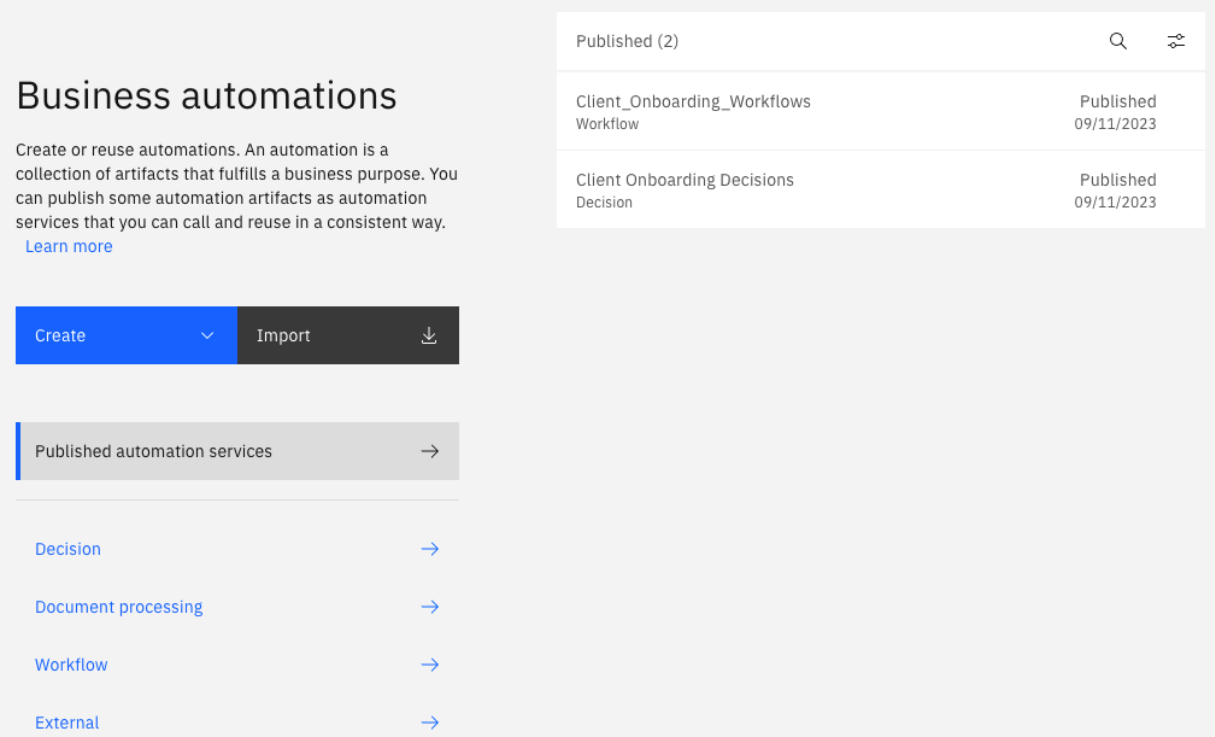
The homepage contains cards that showcase recent artifacts across all installed Cloud Paks in the system. For IBM Cloud Pak for Business Automation, the recent business applications and automation services are shown.



1. In the top-left corner, click on the menu icon and select **Design → Business automations** to access the automation repository.

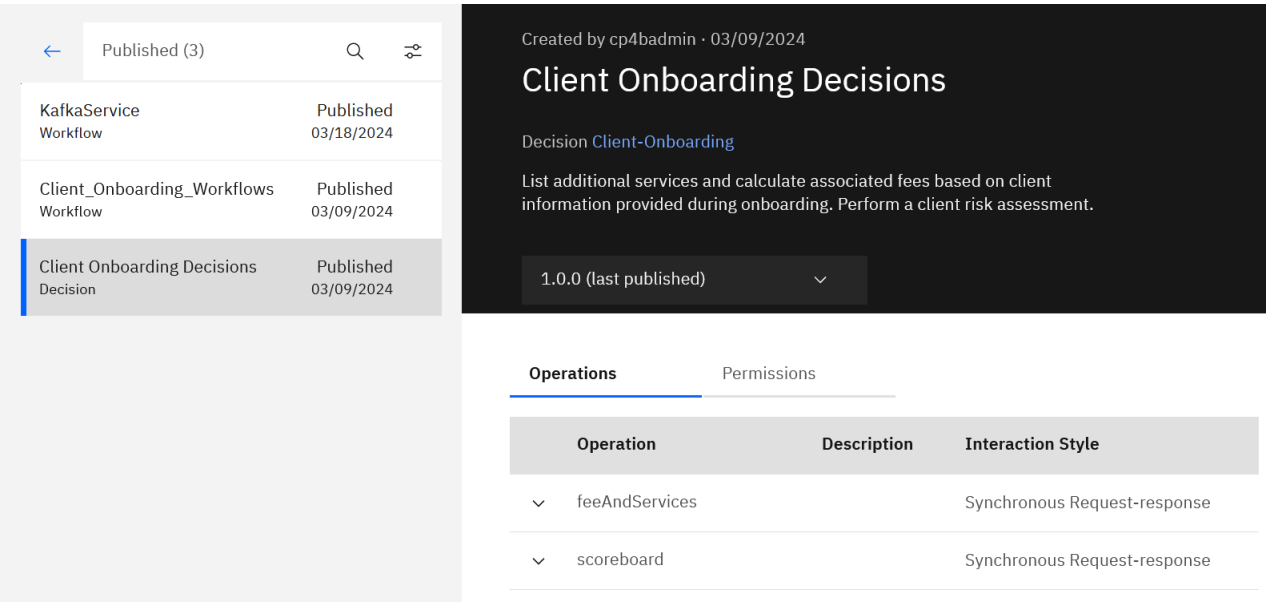


This brings up the Business automations page where you can create or reuse automations from different capabilities of IBM Cloud Pak for Business Automation. If a capability is not installed on the system, it will be greyed out.



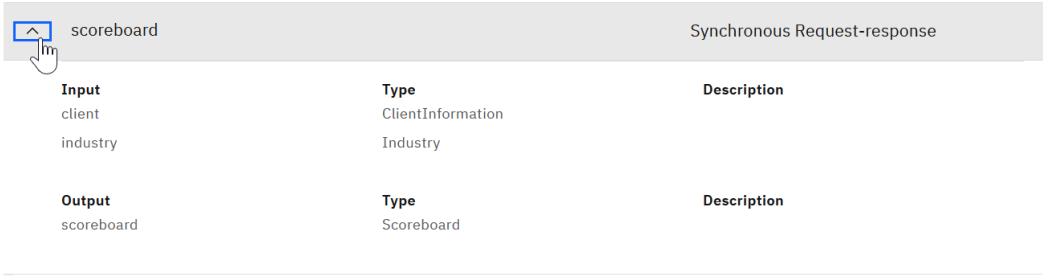
The default selection **Published automation services** shows all automation services available for consumption. The one we will be consuming as a part of this exercise is **Client Onboarding Decisions**.

2. Click on **Client Onboarding Decisions** to view its details.



An automation service can contain multiple operations. The table on the right shows the operations available along with a description for each operation. For this exercise, we will consume the **scoreboard** operation as the description matches our goal of scoreboarding the client.

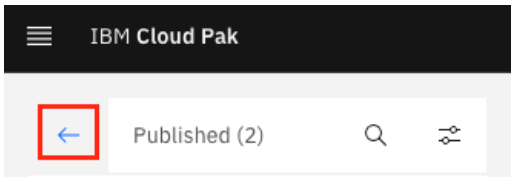
3. Click on the **twisty** icon next to **scoreboard** to view more details about the operation.



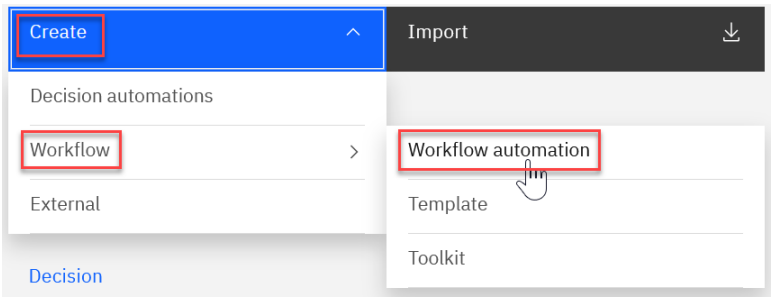
Here, we can see the inputs and outputs that are specified for this operation. This means that anyone consuming this automation service will need to provide an **industry** and **client information** and will receive the **scoreboard** in return. You can also see the interaction style of the operation, in the case of the **scoreboard** operation, synchronous request-response.

We will be consuming this automation service in a Workflow.

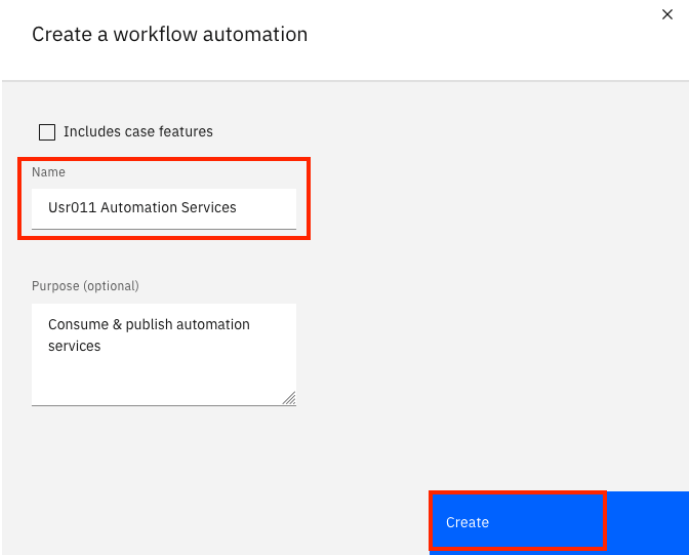
4. Click on the **Back** button in the upper-left corner.



5. Click on **Create → Workflow → Workflow automation**.



6. In the **Name** field, enter ***UsrNNN* Automation Services** where *UsrNNN* is your username.
7. Provide an optional purpose.
8. Click on **Create**.

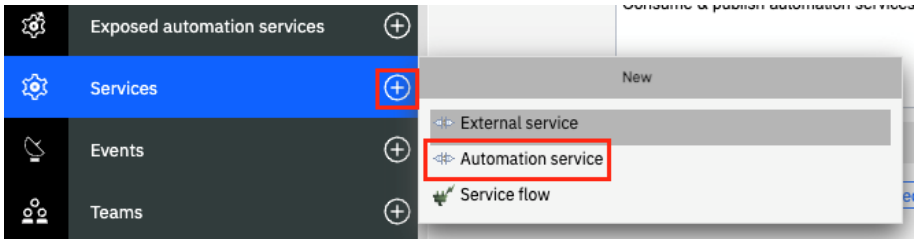


This opens the IBM Process Designer which is the primary modeling and designing tool in IBM Business Automation Workflow.

The left-hand side pane is the library panel where you can create and access different artifacts.

Note: If the IBM Process Designer window does not load the first time, click on the browser's address bar and press Enter to reload the page.

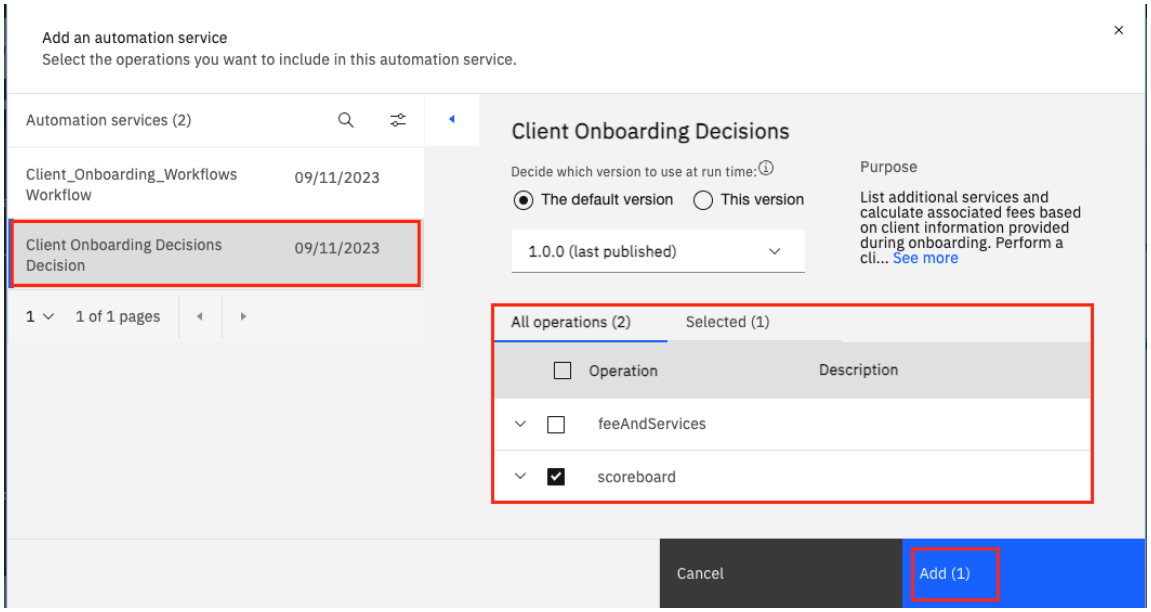
9. In the library panel, hover over **Services**, click on the **+** button and select **Automation service**.



This brings up the list of published automation services where you can select which one you want to consume.

10. Click on **Client Onboarding Decisions**.

11. Select only the **scoreboard** operation.

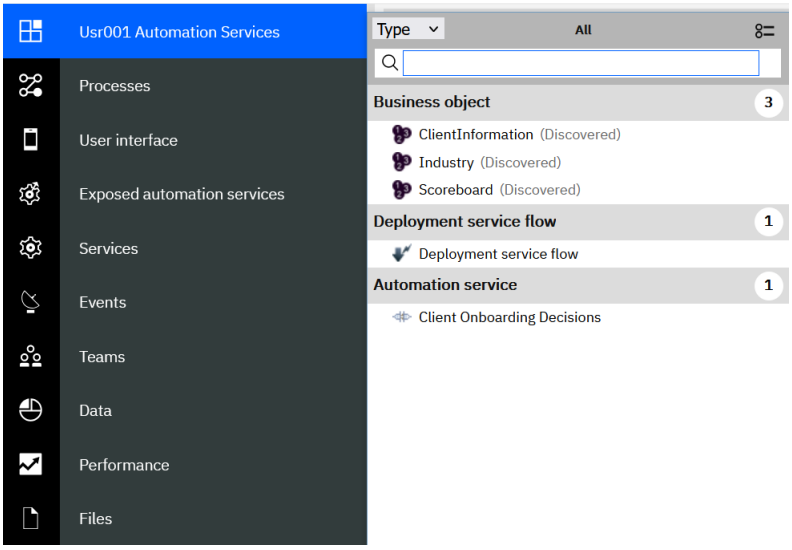


At the top, you can select which version of the automation service you want to consume. By default, the last published version is always chosen. We will leave that selection as is.

12. Click on **Add (1)**.

This creates the artifacts necessary to create the automation service in your workflow project and opens the Automation Service. This includes any business objects that are required to call the service. Let's look at the objects created.

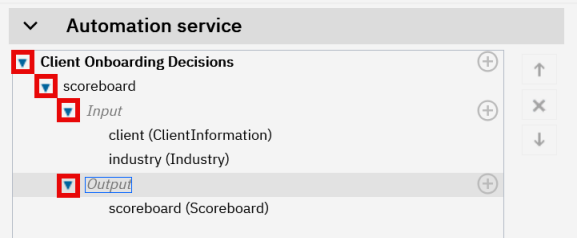
13. In the library pane on the left click on the title of your project to show the list of artifacts.



As you can see, the **ClientInformation, Industry** & **Scoreboard** business objects are automatically discovered as they are the inputs and output required to invoke the service.

Next, we will take a deeper look at the automation service.

14. Click on the **twisty** icon for the **scoreboard** operation to see its details.
15. Click on the **twisty** icons for the **Input** and **Output**.

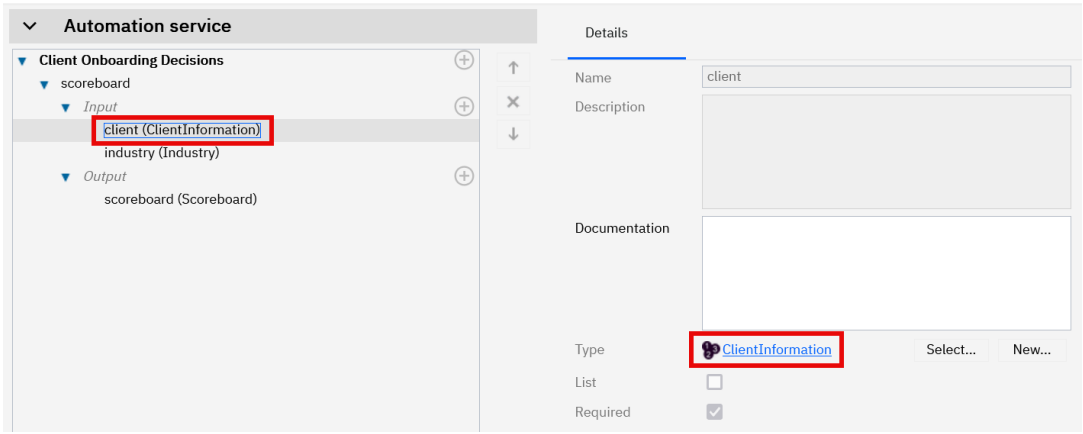


You can see the business objects used as input and output by the scoreboard operation.

16. Click on **client** under **Input**.

    On the right, the details for the parameter are shown including its type: ClientInformation.

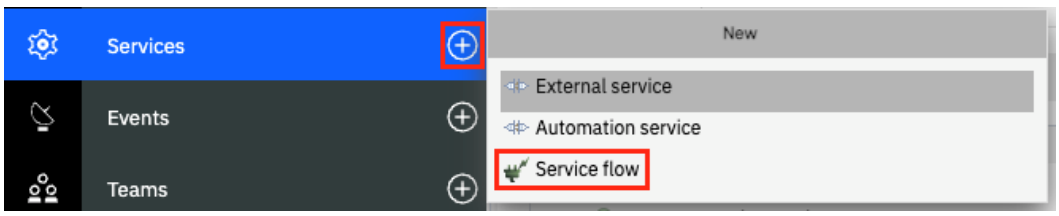
17. Click on **ClientInformation** to open the business object and see its parameters.



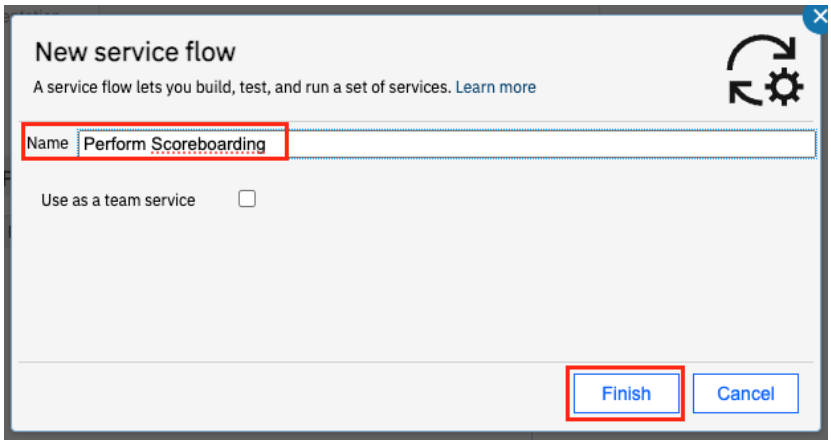
18. In the **Parameters** section, you can see the different parameters within this business object:

You can similarly explore the other input and output for the automation service. Next, we will create a Service Flow that can invoke this automation service.

19. In the library pane on the left, hover over **Services**, click on the **+** button and select **Service flow**.



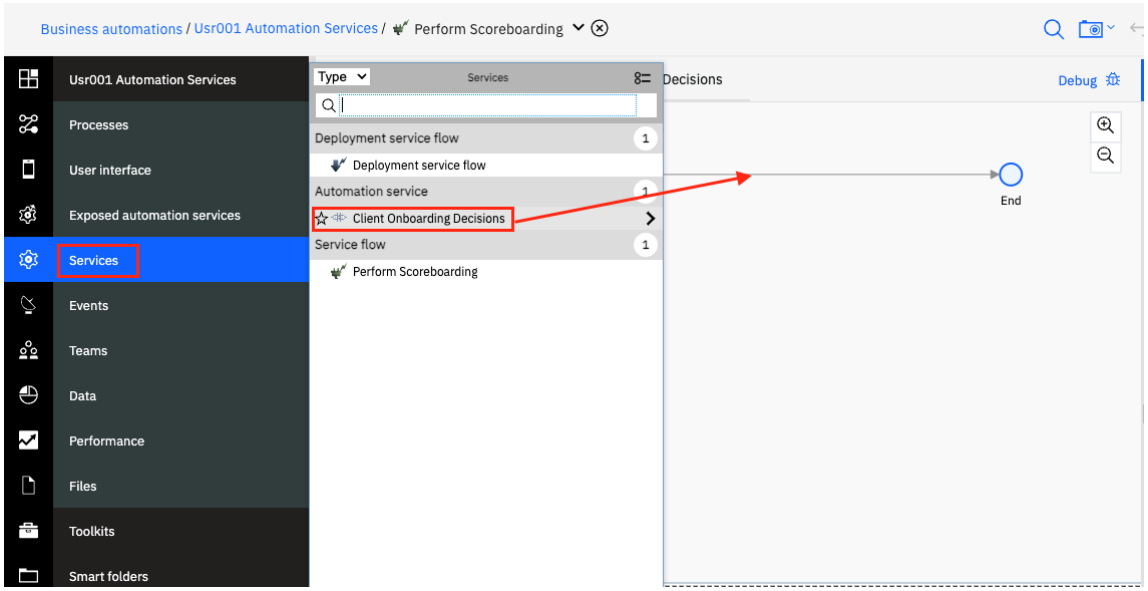
20. In the New Service Flow wizard, enter **Perform Scoreboarding** as the name.
21. Click on **Finish**.



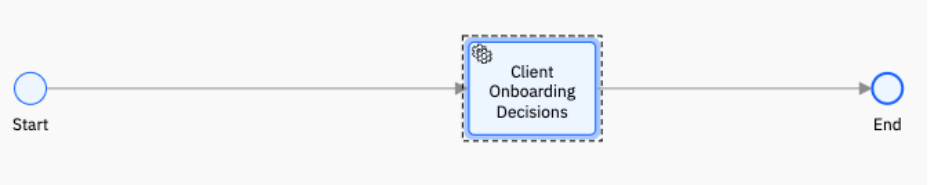
The service flow editor should now open with a default diagram:

Now, we want to add a call to the automation service between the line connecting the **Start** and **End** nodes.

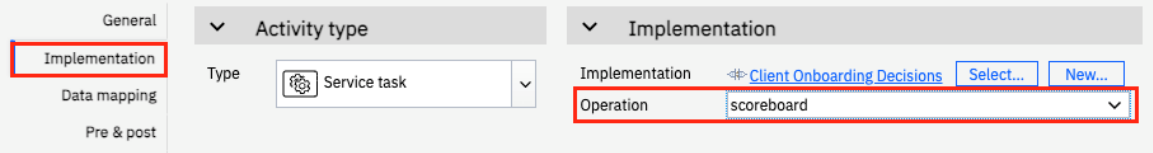
22. In the library pane on the left, click on **Services** and drag the **Client Onboarding Decisions** automation service on the line connecting the **Start** and **End** node.



Your diagram should then look as follows:



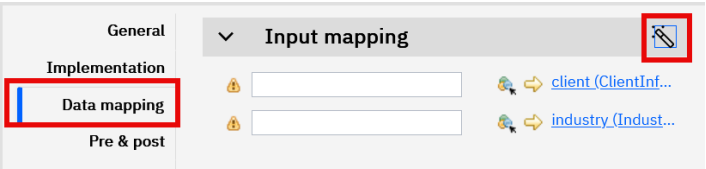
23. Click on the **Client Onboarding Decisions** service task between the **Start** and **End** nodes.
24. In the properties pane at the bottom, under **Implementation** select the **scoreboard** operation.



25. Switch to the **Data mapping** tab

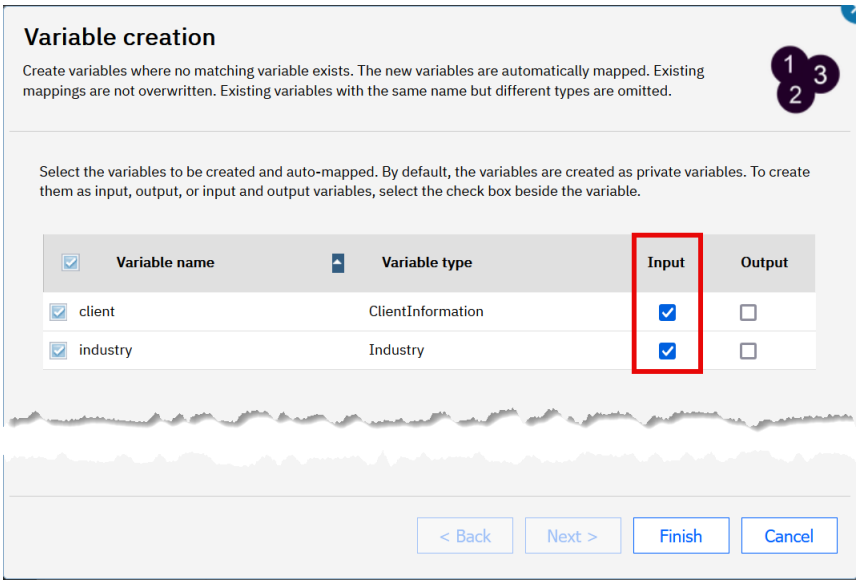
    The contents of this tab allow you to map constant values and/or variables to the input and output of the automation service.

26. Click on the **auto-map** icon for the **Input Mapping** section



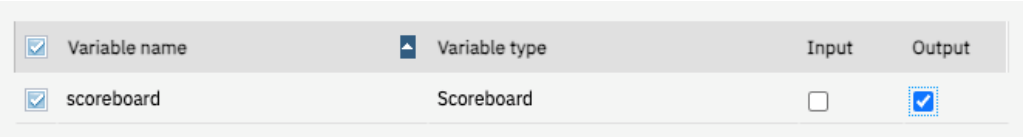
    This brings up the variable creation wizard which allows us to automatically create the required variables. We want this service flow to be reusable so that it can be called by other artifacts (such as a human service). To do that, we can select the **client** and **industry** as inputs to this service flow. This means that anyone calling the **Perform Scoreboarding** service flow can provide these two variables as inputs.

27. Select the **Input** checkboxes for both **client** and **industry**.



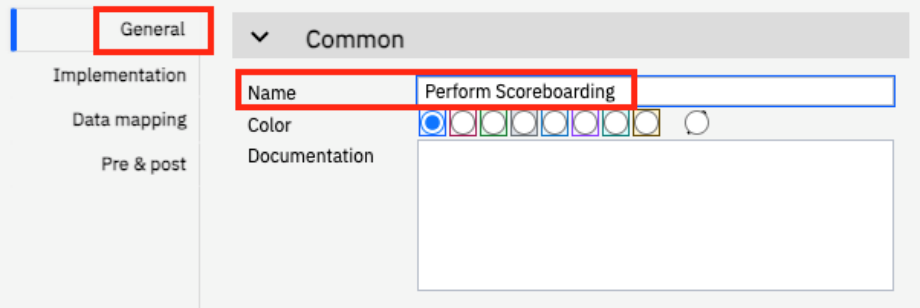
    We would check the output checkboxes if we were modifying the input. This way any artifact calling the service flow would be able to get the updated values as the output to the flow.

28. Click on **Finish**.
29. Repeat the steps above to auto-map the output variable **scoreboard**. In this case however, select the Output checkbox



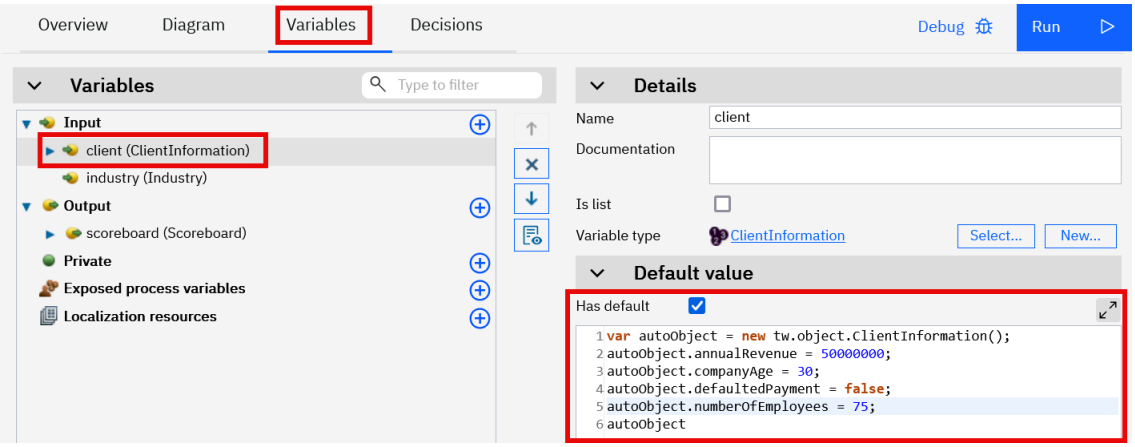
Your data mapping section should now look as follows:

30. Switch to the **General** tab.
31. Change the name of the task to **Perform Scoreboarding**.



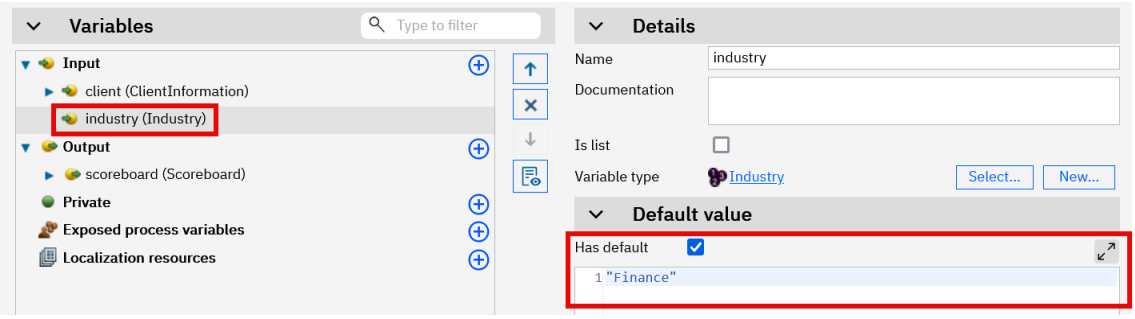
Now, to test this service flow, we will need to provide some default values.

32. Click on the **Variables** tab at the top.
33. Select the **client** input variable.
34. On the right-hand side, **check** the checkbox for the **Has default** field.
35. Updated the following values in the autogenerated script:
    a. **annualRevenue:** 50000000
    b. **companyAge**: 30
    c. **numberOfEmployees**: 75



36. Click on the **industry** input variable.

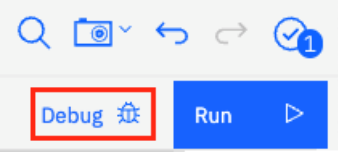
37. **Check** the **Has default** checkbox.
38. Update the industry in the autogenerated script to **Finance**.



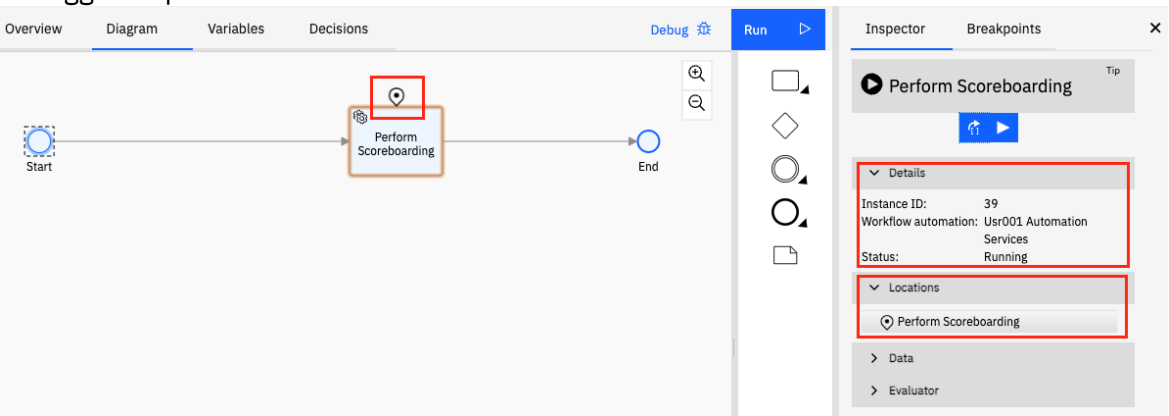
With the default values added, we are now ready to test the automation service.

39. Click on the **Diagram** tab at the top.

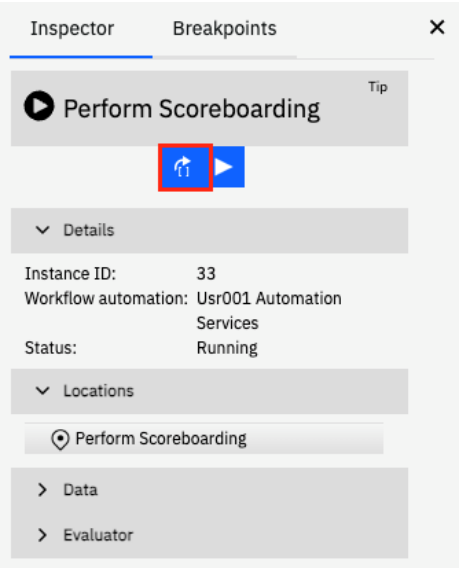
40. Click on the **Debug** icon in the upper-right corner.



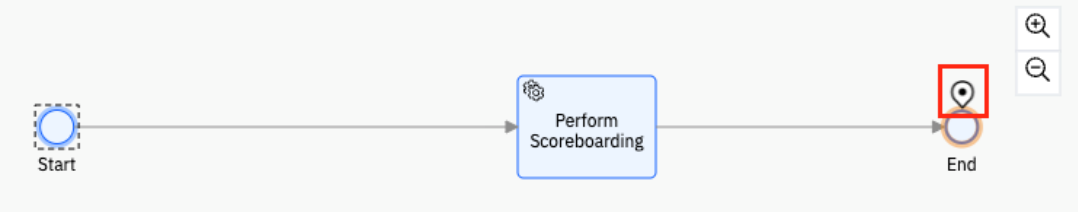
Notice how the **Inspector** panel is opened to the right containing the controls and information about your debugging session. The diagram now also shows a location pin indicating the current debugger step.



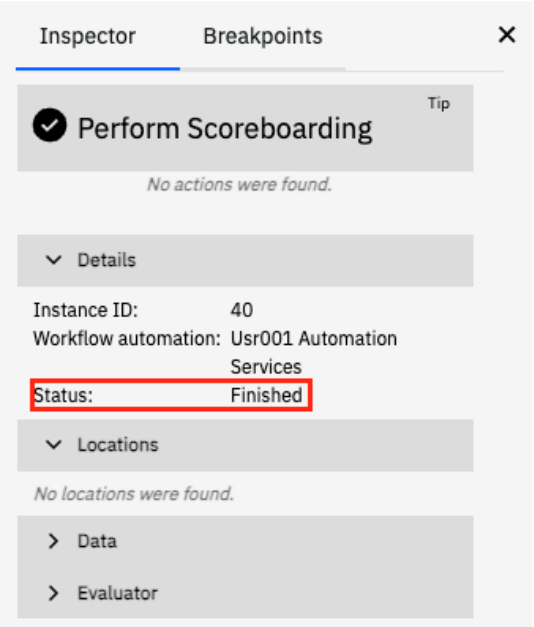
41. Click on the **Step over** button to invoke the automation service.



Notice the location pin changed from the **Perform Scoreboarding** to the **End** node.



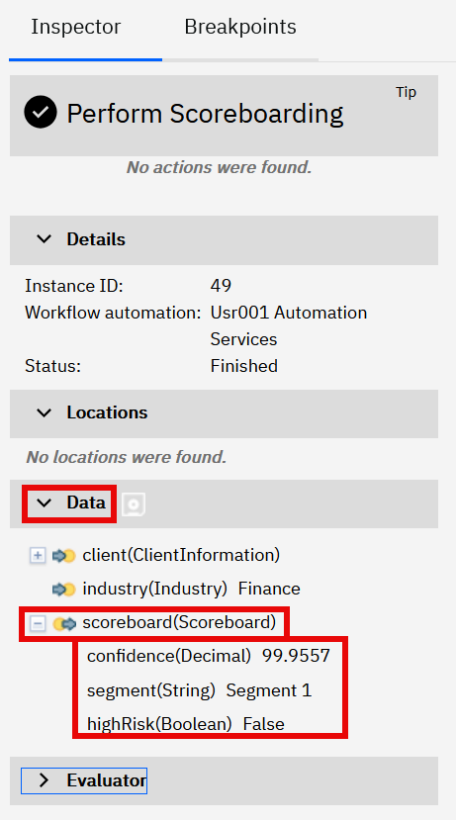
42. Click the **Step over** button one more time to complete the execution of the Perform Scoreboarding service flow.



The Inspector status should now be updated to **Finished**.

43. Click on the **twisty** icon to expand the **Data** section.
44. Click on the **+ sign** in front of the **scoreboard** variable, which holds the result of invoking the external service.

45. Verify that the values shown match the screenshot below (there might be small variations in the confidence).

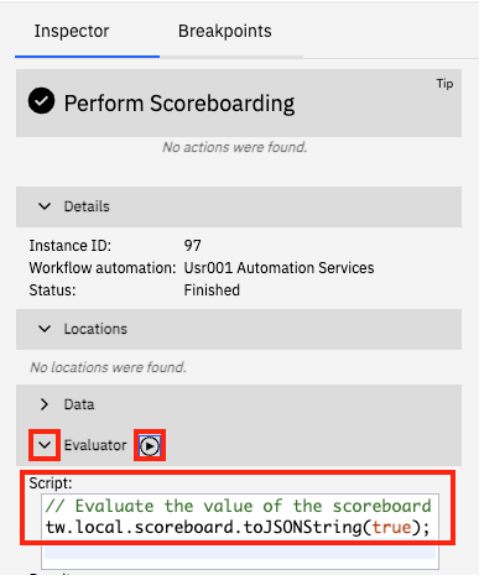


An alternative way to see the variable results and do much more, is using the Evaluator section.

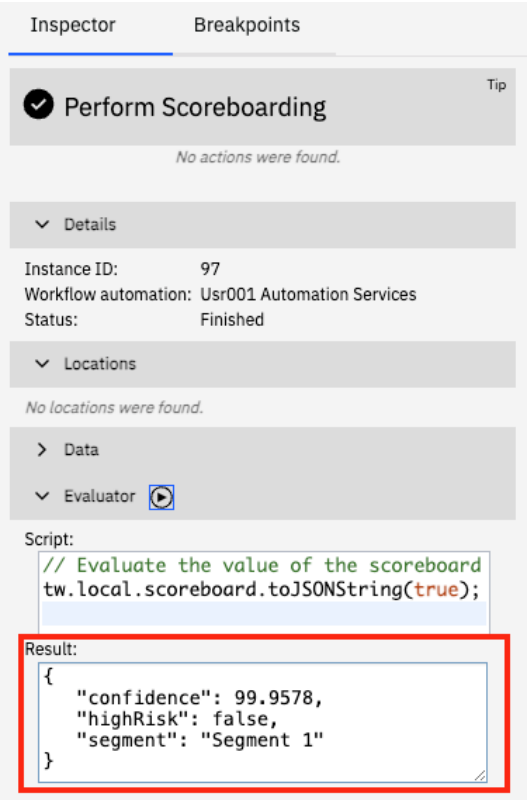
46. Click on the **twisty** icon to expand the **Evaluator** section.
47. Enter the following JavaScript expression in the **Script** field to inspect the scoreboard value returned by the **Perform Scoreboarding** service flow.

```
// Evaluate the value of the scoreboard output variable
tw.local.scoreboard.toJSONString(true);
```

48. Click the **Run the script** button.



49. Verify that the values shown in the **Results** section match the screenshot below (there might be small variations in the confidence):



With that, you have successfully completed this exercise and learned how to consume an automation service and debug it from Process Designer. The service flow that encapsulates this automation service can now be reused throughout the project to call the decision service. If you want to learn more about this along with the basics of IBM Business Automation Workflow, look at the **Introduction to IBM Business Automation Workflow** lab.

In the next exercise, we will create an external service that calls out to a Java application to send emails.

3 Exercise: Create an External Service

3.1 Introduction

External services support various bindings like Java, REST API, Web Service, etc. In this exercise, we will create an external service that calls a Java application (.jar file) that sends an email.

Note: A single external service can only have one type of binding.

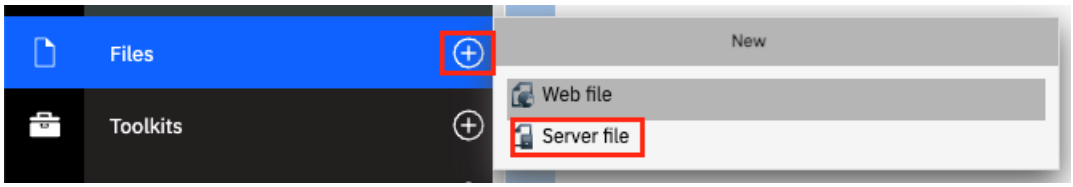
3.2 Exercise Instructions

1. Open the ***UsrNNN* Automation Services** workflow project if not already open.

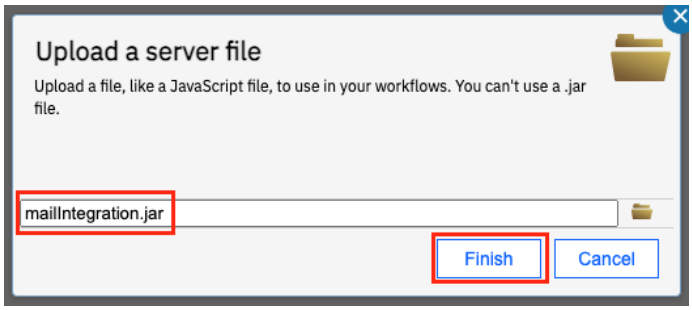
   You can do this by going to the Business automation repository in **IBM Business Automation Studio**.

   We first need to add a jar file to the project. This file contains the Java implementation to send an email. The integration samples page contains additional workflow project exports and the sample Java code that can be used to interact with emails.

2. In the library pane on the left, hover over **Files**, click on the **+** button and select the **Server file** option.

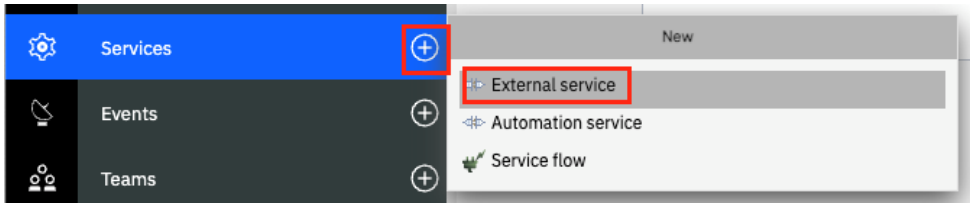


3. Select the **mailIntegration.jar** file downloaded as a part of the lab setup instructions.
4. Click on **Finish**.



   Next, we will create the external service that uses this jar file.

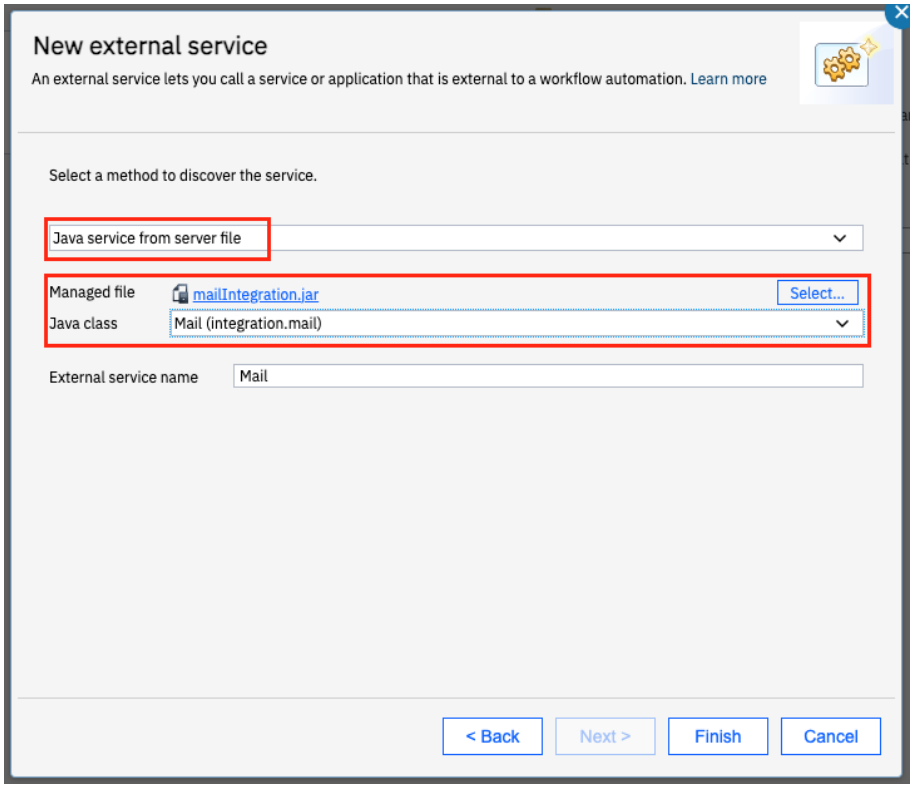
5. In the library pane, hover over **Services**, click on the **+** button and select the **External service** option.



   The **New external service** wizard pops up with two options. As we are integrating with a Java application, we will use the default selection.

6. Click on **Next**.

7. For the **Select a method to discover the service** field, select **Java service from server file** option.
8. In the **Managed file** field, click on **Select** and pick the **mailIntegration.jar** file.
9. For the **Java class** field, select the **Mail** class.

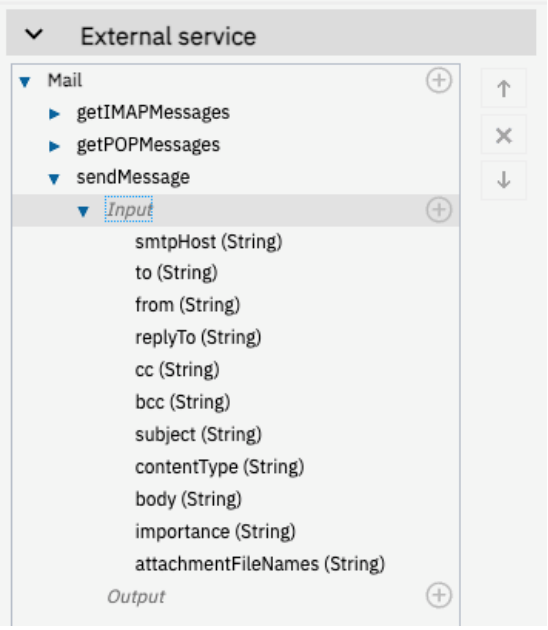


The external service name is automatically updated to match the name of the Java class.

10. Click on **Finish**.

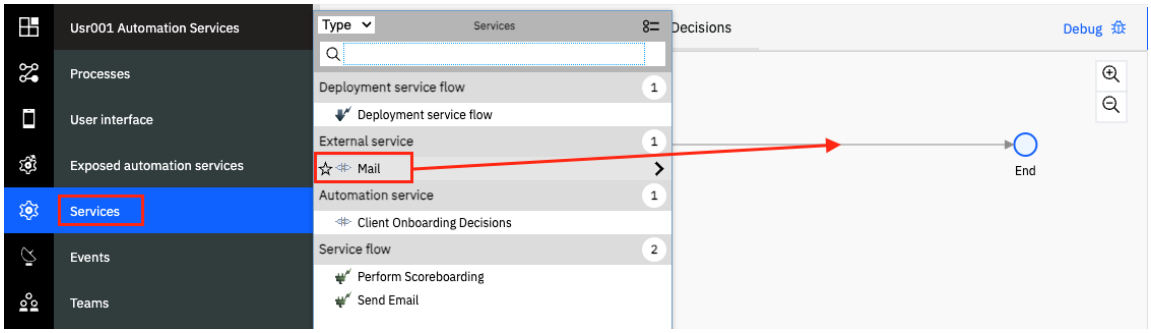
    This opens the external service editor with a similar look and feel to the automation service editor from the previous exercise.

11. Expand the **sendMessage** operation and the **Input** section to view its details.

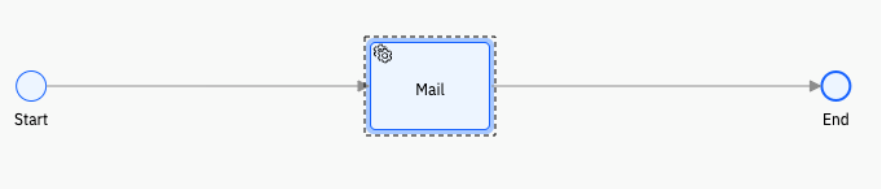


Here you can see the inputs that can be used to send an email. Next, we will create a service flow just like the previous exercise to test this external service and make it reusable. In the next exercise, we will see how to publish an automation service that calls this service flow.

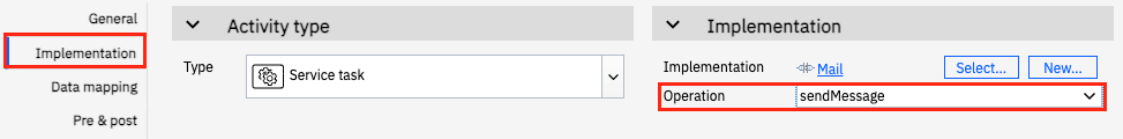
12. In the library pane on the left, hover over **Services**, click on the **+** button and select **Service Flow**.
13. In the New Service Flow wizard, enter **Send Email** as the name.
14. Click on **Finish** to open the service flow editor.
15. In the library pane on the left, click on **Services** and drag the **Mail** external service on the line connecting the **Start** and **End** node.



Your diagram should now look as follows:

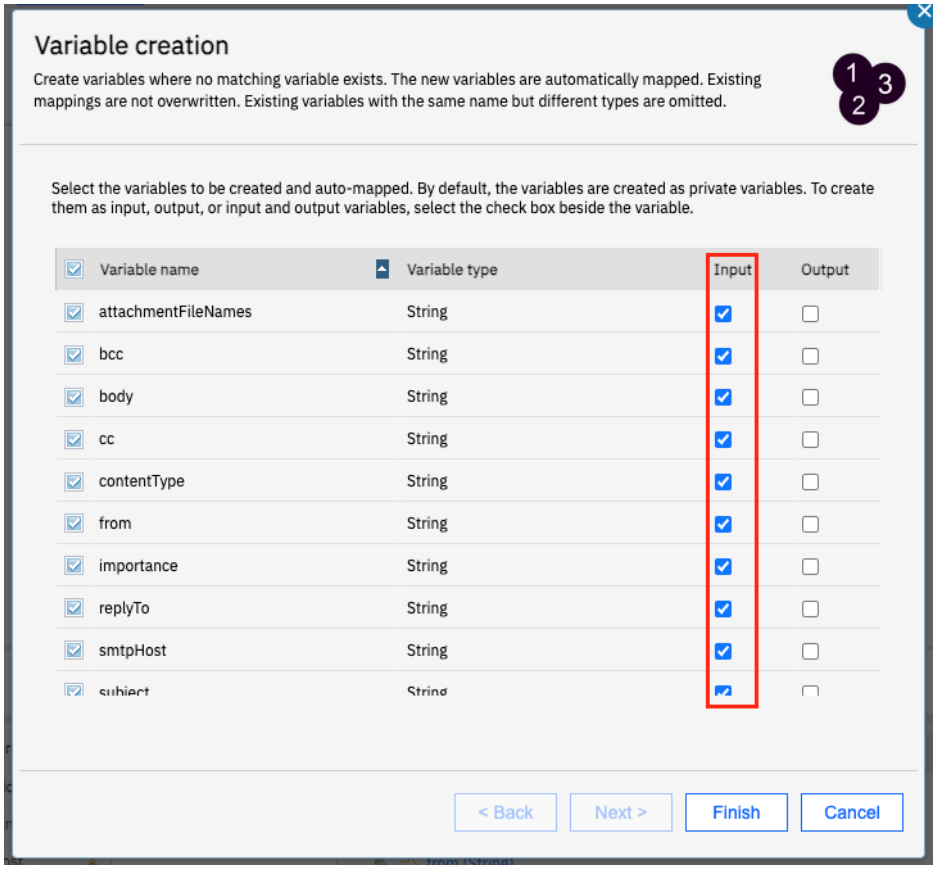


16. In the properties pane, under the **Implementation** section, select the **sendMessage** operation.



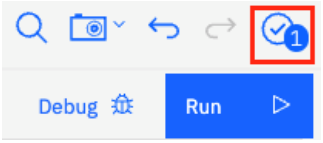
17. Switch the **Data Mapping** tab.
18. Click on the **auto-map** icon for the **Input Mapping** section.

19. In the variable creation wizard, select the **Input** checkboxes for **all** variables.



20. Click on **Finish**.

21. Click on the **Finish editing** button in the upper-right corner.



This completes the exercise.

You can optionally choose to test this service flow by providing default values to the input variables like you did when testing the automation service. For that you will need access to an email account with an SMTP server.

4 Exercise: Create and Publish an External Service

4.1 Introduction

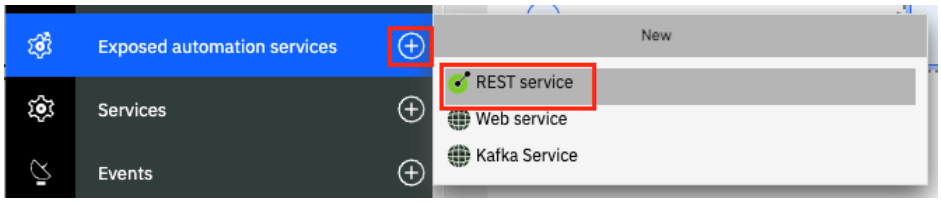
In this exercise, we will create an automation service containing an operation that invokes this service flow. We will then see how to publish this automation service.

4.2 Exercise Instructions

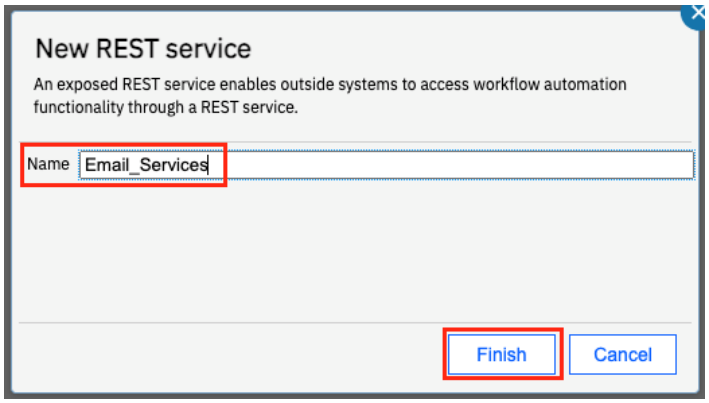
1. Open the ***UsrNNN* Automation Services** workflow project if not already open.

   You can do this by going to the Business automation repository in **IBM Business Automation Studio**.

2. In the library pane on the left, hover over **Exposed automation services**, click on the **+** button and select the **REST service** option.

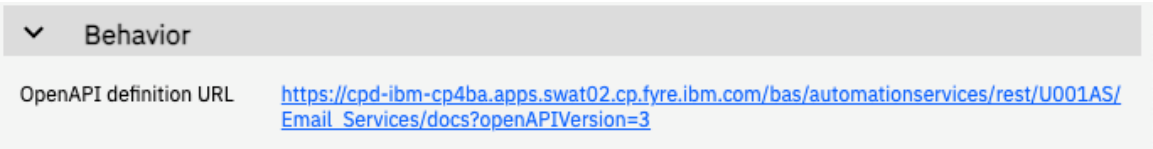


3. In the **Name** field, enter **Email_Services**.
4. Click on **Finish**.



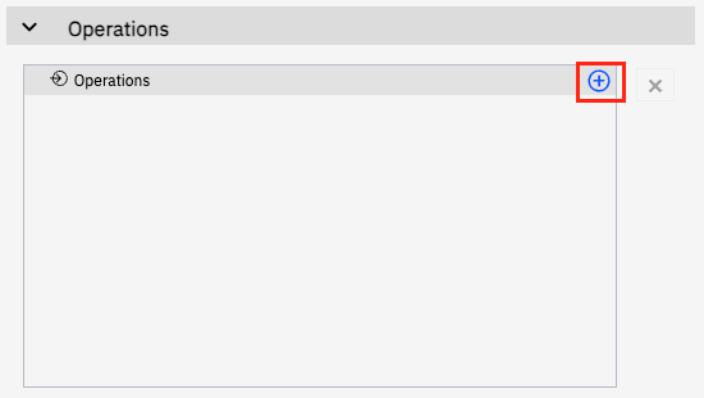
This opens the **REST service** editor where you can add multiple operations. In this exercise, we will only add one operation to send emails.

REST Services also provide an OpenAPI definition URL. The OpenAPI spec defines a standard, language-agnostic interface for REST APIs.

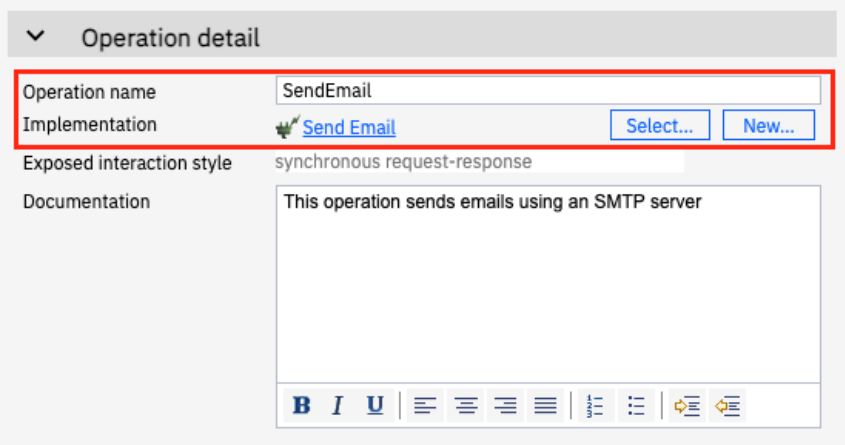


Note: The URL you see will be different compared to what's in the screenshot based on your lab environment.

5. In the **Operations** section, click on **+** to add a new operation.

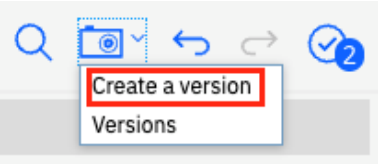


6. In the **Operation detail** section on the right, enter **SendEmail** in the **Operation name** field.
7. For the **Implementation** field, click on the **Select** button and select the **Send Email** service flow created in the previous exercise.



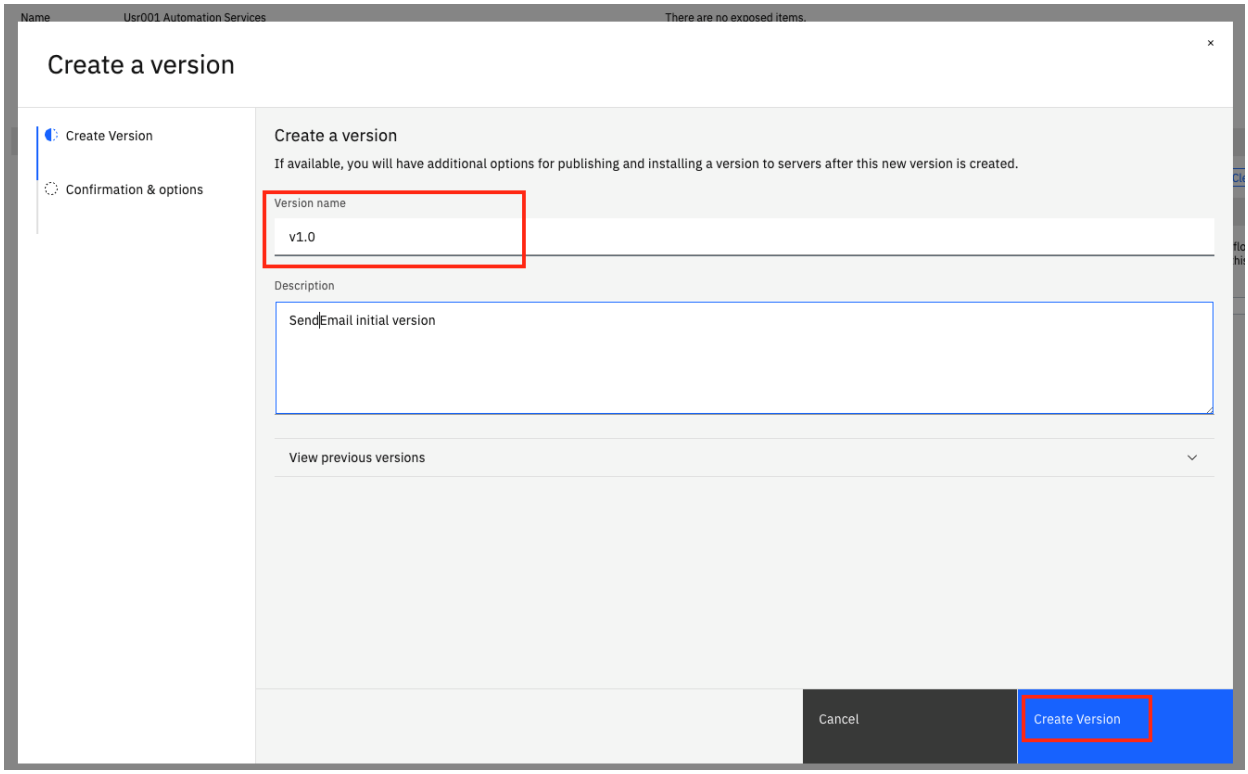
Next, we will need to create a version of this workflow project so that the REST service can be published as an automation service.

8. Click on the **Version** button in the upper right corner and select **Create a version**.



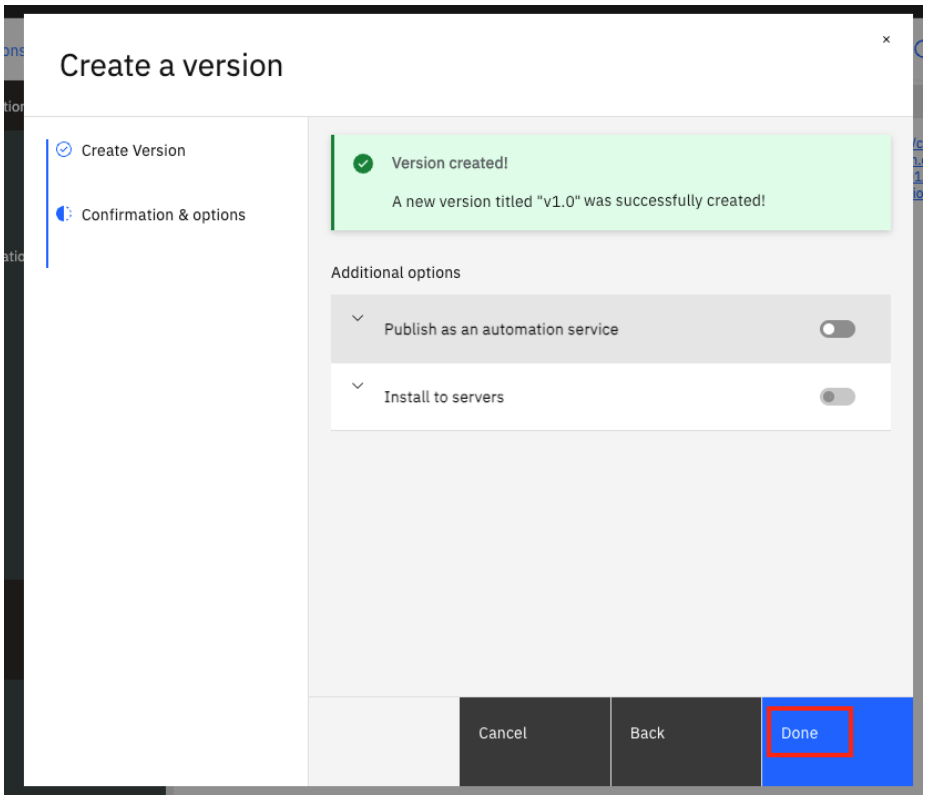
9. In the **Create a version** wizard, enter **v1.0** in the **Version name** field and an optional description.

10. Click on **Create Version**

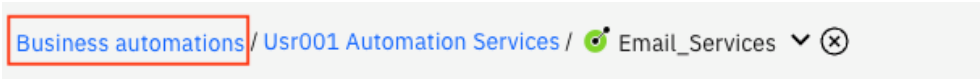


This will create the new version of your service and will take you to the **Confirmation & options** panel of the **Create a version** wizard. You can control access, modify permissions, and publish the automation service from this panel. However, for this lab we will publish the automation service from Business Automation Studio.

11. Click the **Done** button to exit the **Create a version** wizard



12. Click on **Business automations** in the upper-left corner to go back to **IBM Business Automation Studio**.

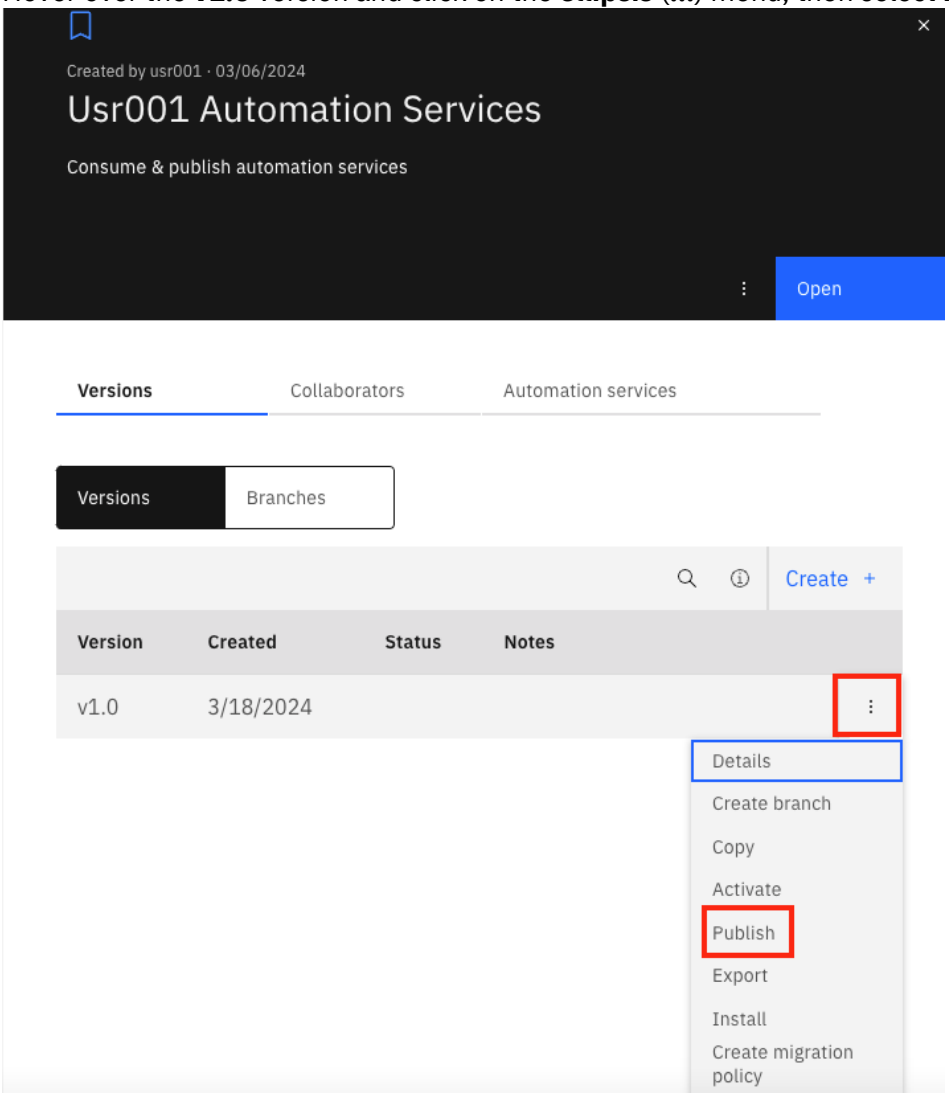


13. Click on your Workflow project ***UsrNNN* Automation Services**. Do **NOT** click on **Open** but on the tile itself.

    Hint: You can use the search for your project by clicking on the **search** icon the upper-right corner.

    The project details view opens on the right. From this view you can fully manage the different versions of your project and publish or unpublish the capabilities provided by your service. Notice the version we previously created from the IBM Process Designer is listed.

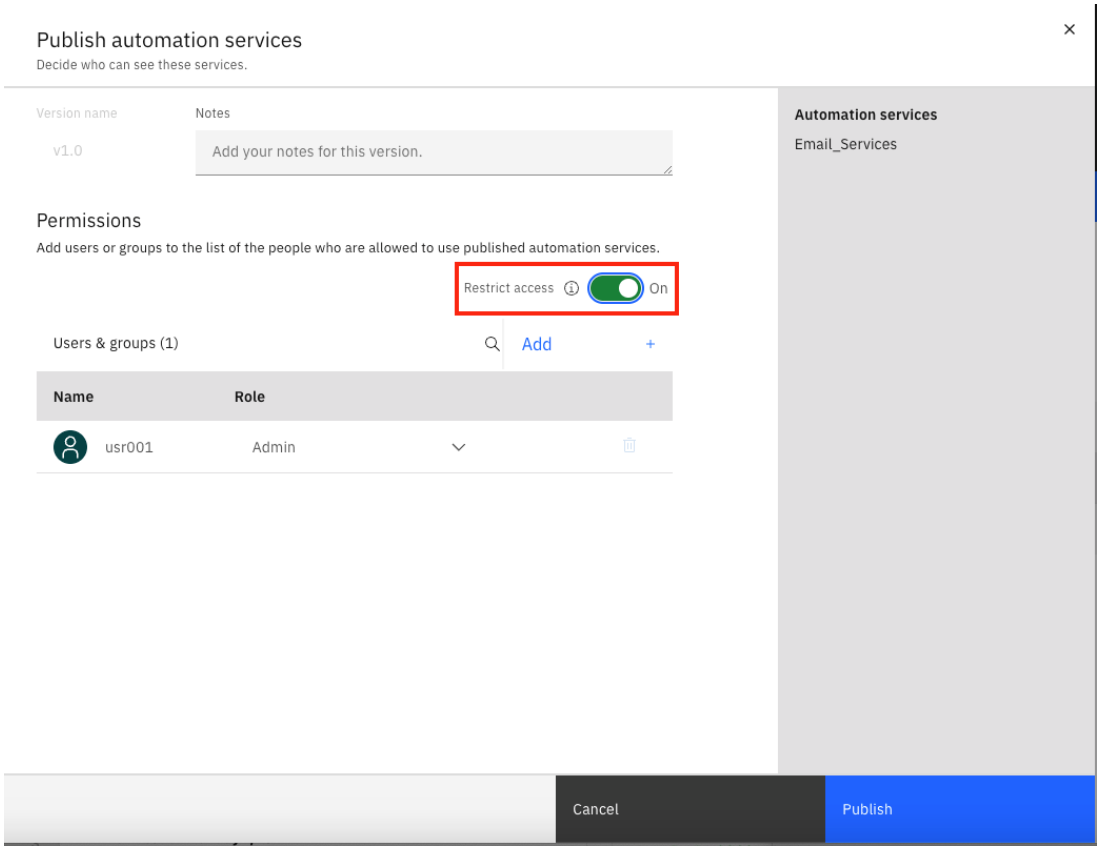
14. Hover over the **v1.0** version and click on the **ellipsis** (**...**) menu, then select **Publish**.



    This brings up the **Publish automation services** dialog.

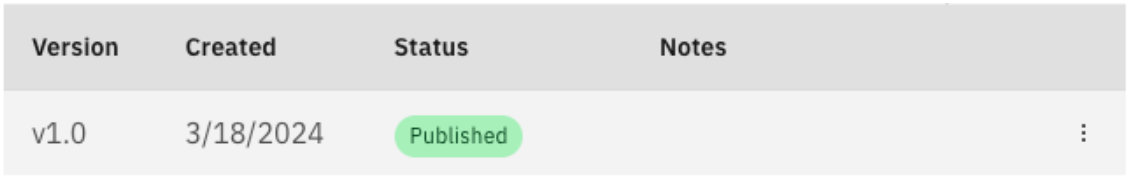
15. Click on the **Restrict access** toggle to turn on access control.

    Keep the default settings, which helps to keep the environment clean for other participants, in case of a multi-user event.



Notice that you can assign different roles to the users and groups for this automation service and that you can add additional users and groups using the **Add** button.

16. Click on **Publish**.
    The version status will show shows **Published** after a few seconds.

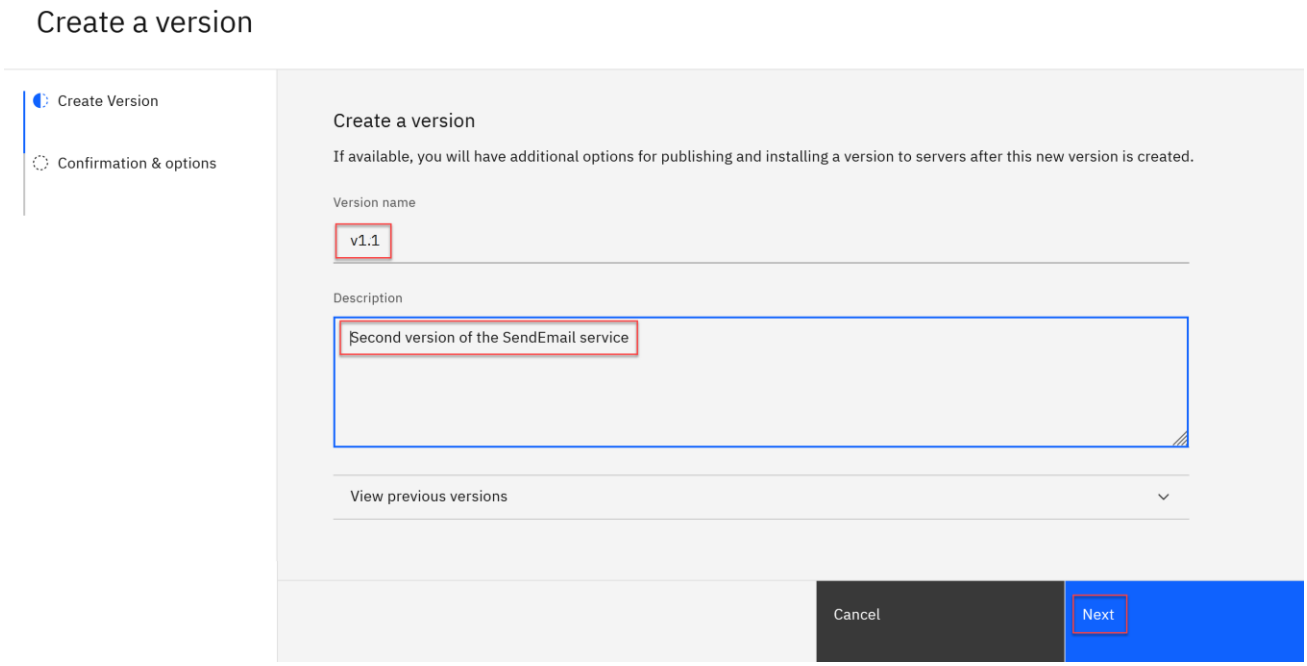


Now you will create a new version to explore additional capabilities available from this view.

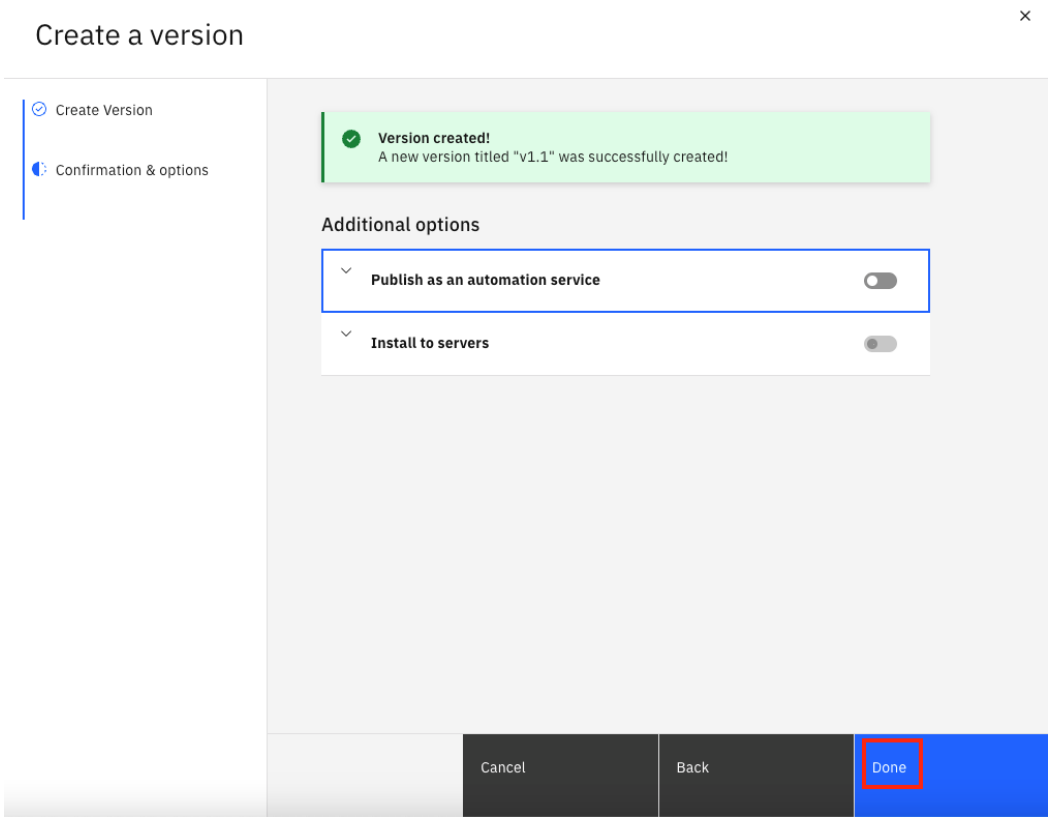
17. Click on the **Create +** button in the top-right corner of the versions table.

18. Enter **v1.1** in the **Version name** field and an optional description.

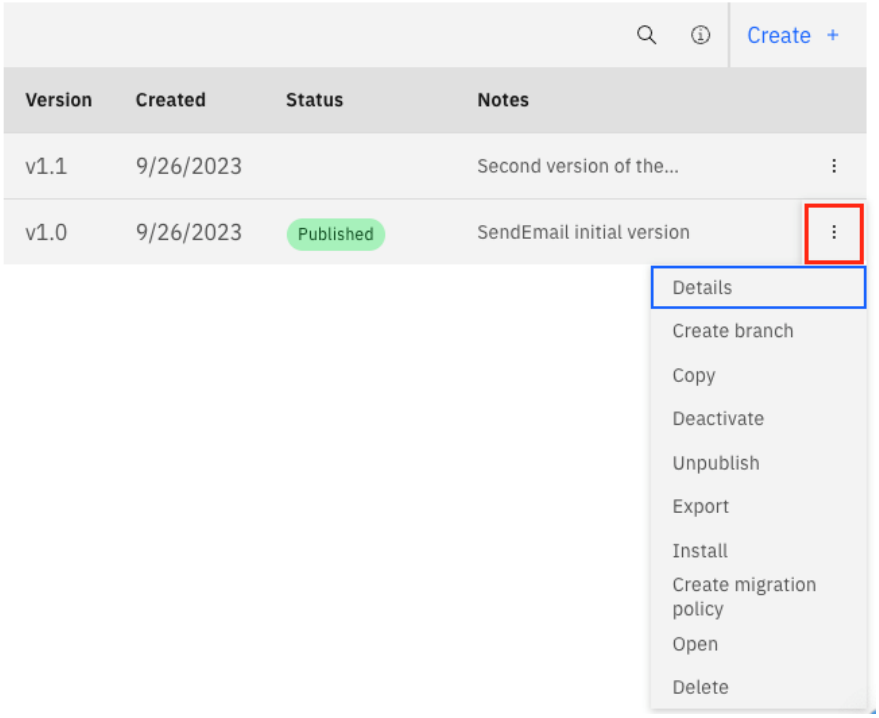
19. Click on the **Next** button.



20. You should now see the Version created message in green. Click on the **Done** button to exit the **Create a version** wizard.

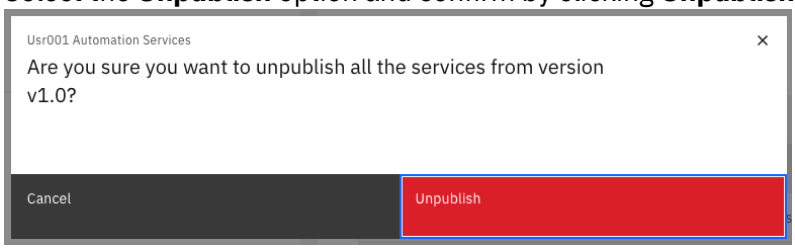


21. Click on the **ellipsis** (**...**) menu next to the version that has already been published (**v1.0**).



Explore the actions available for the version.

22. Select the **Unpublish** option and confirm by clicking **Unpublish** in the confirmation dialog.

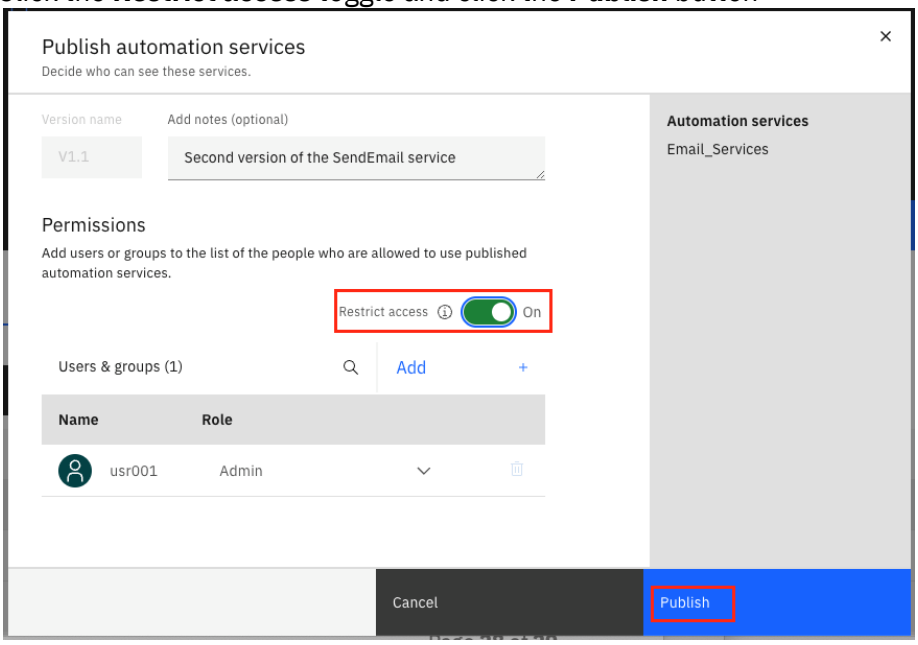


Notice once the version is unpublished, the capabilities provided by your project will no longer be available for other components in the platform.

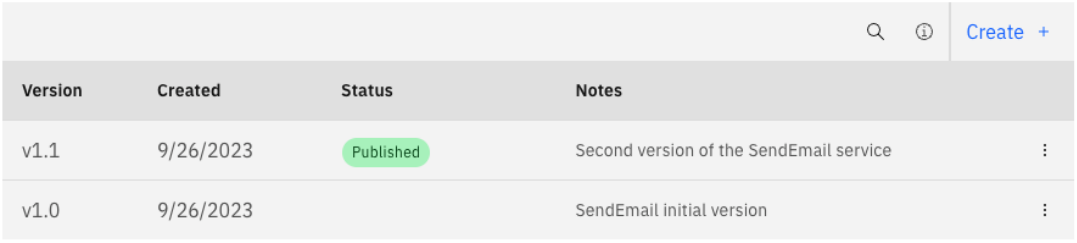
Close the confirmation dialog by clicking on the **X** icon in the top-right.

23. Finally publish the latest version of the project by clicking on the **ellipsis** (**...**) menu and selecting the **Publish** option for version **v1.1**.

24. Click the **Restrict access** toggle and click the **Publish** button

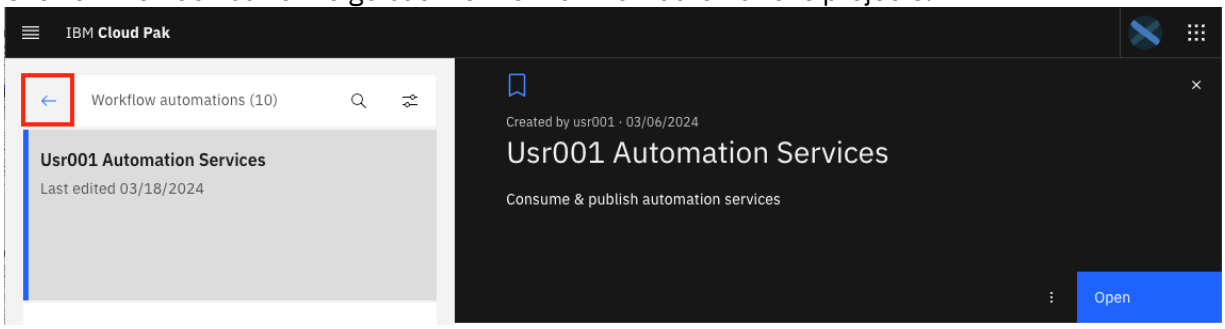


Once the new version is published, the status will show as **Published**.

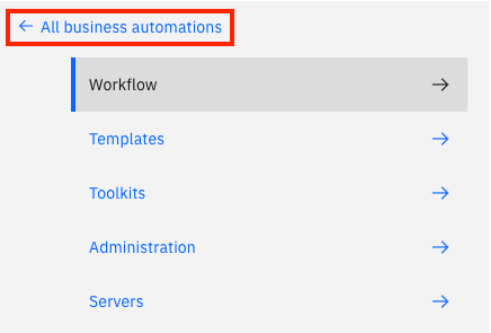


Now we will validate that the automation service is available.

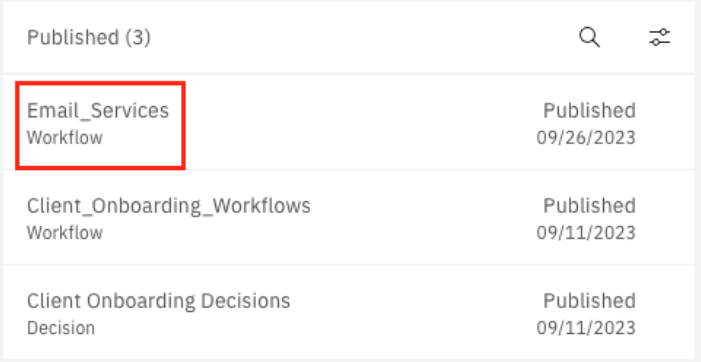
25. Click on the Back button to go back to the Workflow automations projects.



26. Click on **All business automations**.



27. The list of published automation services now shows the **Email_Services** automation service.



This indicates that the **Email_Services** is ready to be used by other components in the platform.

This concludes the Create and publish an external service exercise.

Congratulations on completing the lab!