

Watsonx Orchestrate Lab 3: Decision Automation and Skill-based Actions

Use case

In this lab we show how to work with automations and specifically how to implement a decision automation that can accomplish a task based on certain criteria. For example, if you want to approve a loan application depending on the credit score and income of the applicant, a rule-based decision to accomplish this will be very helpful. This decision can be used as a skill in IBM watsonx Orchestrate and can be made available as a skill-based action in your assistant.

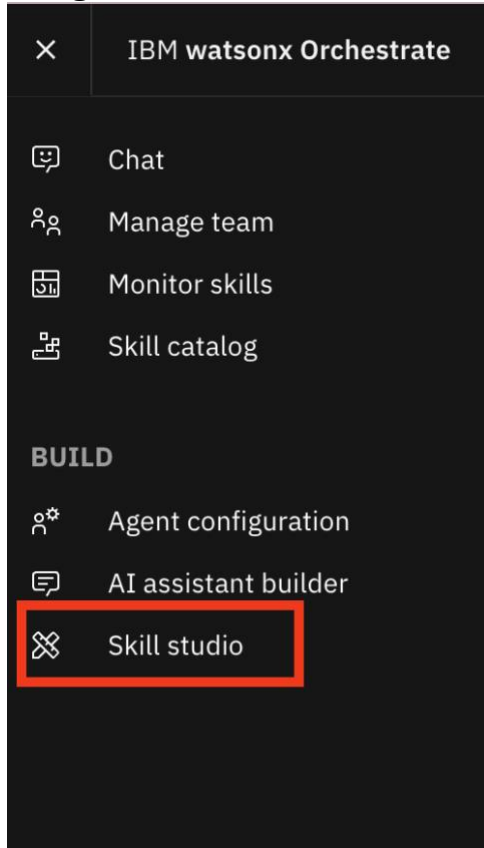
Assumptions

See lab **README** file for details

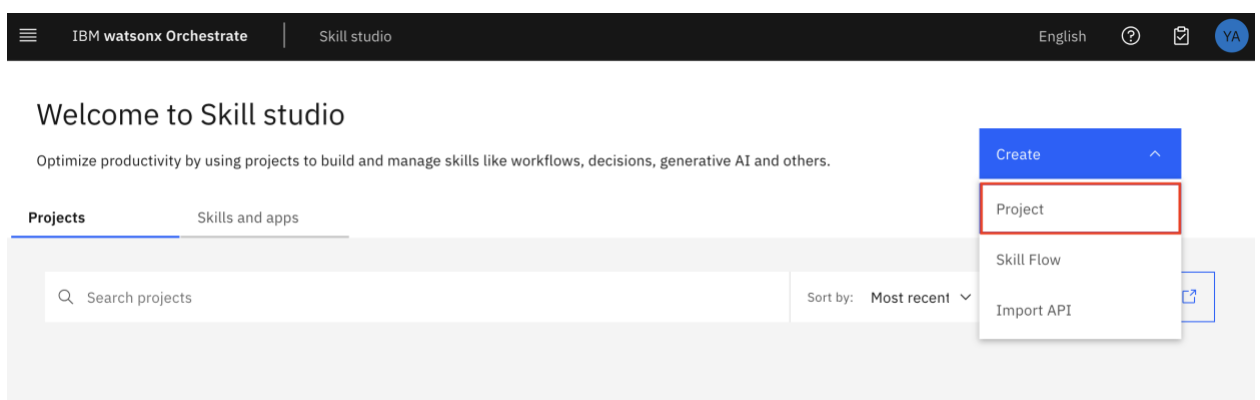
1. You have access to IBM watsonx Orchestrate (Standard Edition, which includes Automation/Projects in Skill Studio)
2. You are authorized to create automations in Skill Studio

Step 1: Create a decision automation

1. Navigate to **Skill studio** from the menu:



2. Click on the **Create** dropdown and select **Project** to create a new automation:



3. Give it a name (e.g. “YourInitials_Lending_Services”) and click **Create**:

New project

Create project

Create an empty project and build it from scratch.

Name

YA_Lending_Services

Note: This is a symbolic name that must be unique and cannot be changed later.

Description (optional)

Describe your project

Cancel Create

4. Next select **Decision** tile.

Build Data Operations History Publish

Get started by choosing a component

Decision

Automate complex business decisions with rules and decision tables.

Workflow

Model your business process and publish it to use as a skill or to generate tasks.

Generative AI

Use generative AI to analyze or create contextual content.

5. Select **Decision model** as the model for your decision.

Build Data Operations History Publish

Now choose a model

Decision model

Decompose your decision and define the data its depends on.

Ruleflow model

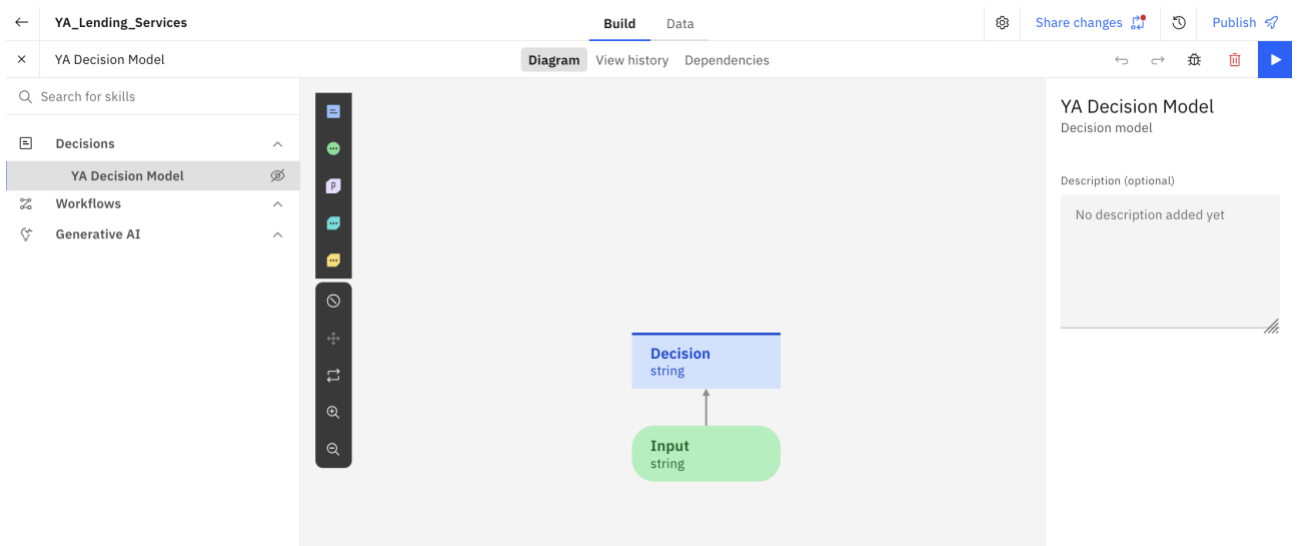
Chain together tasks and specify how and when they are run.

Prediction model

Implement predictions based on artificial intelligence.

6. Give a name to your decision (e.g., “YourInitials Personal Loan”) and click **Create**:

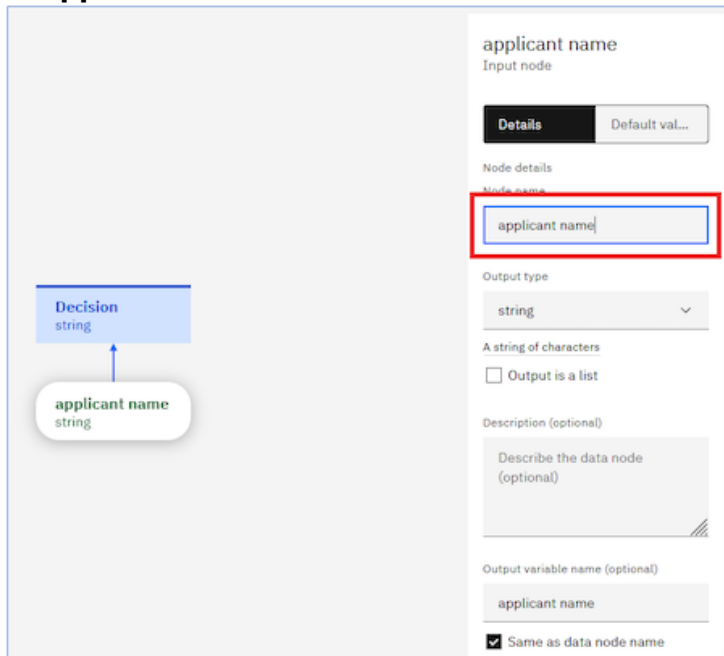
7. The **green nodes** (data nodes) represent data elements that are used by the decision’s rules. The **blue nodes** (decision nodes) represent a step in the decision. They contain rules that will execute to achieve that step. Each decision node outputs a partial decision. In complex decisions, there are many decision nodes, and the output of one decision node will flow as input into another decision node:



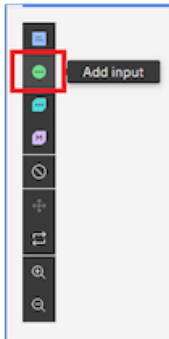
Step 1.1: Create the Data nodes

Now, let's create the data nodes.

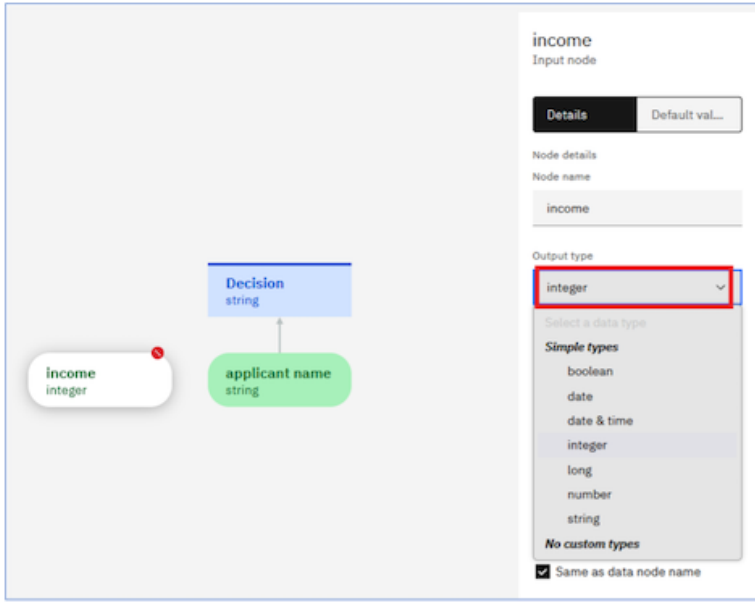
1. Click on the data node and on the right-hand side, change the name of the node to **applicant name**:



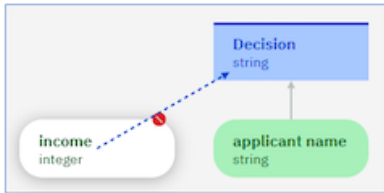
2. Create a new data node by clicking on the **Add input** button in the palette:



3. Select the newly created node, change the name of the node to **income**, and change the type of the node to **integer**:



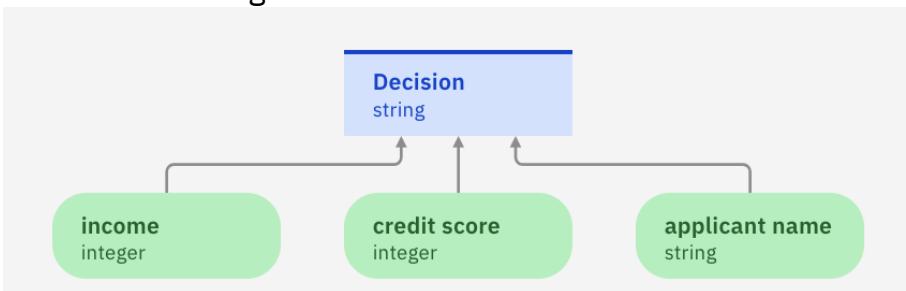
4. Hover on the node and click the **Connect to another node** button, drag the connector to the **decision node** and click on the **decision node**.
Note: If you **Add Node** from the decision node, it is automatically connected.



5. Repeat the steps 1-4 in this section to add additional nodes. They are:

Node name	Node type
income	integer
credit score	integer

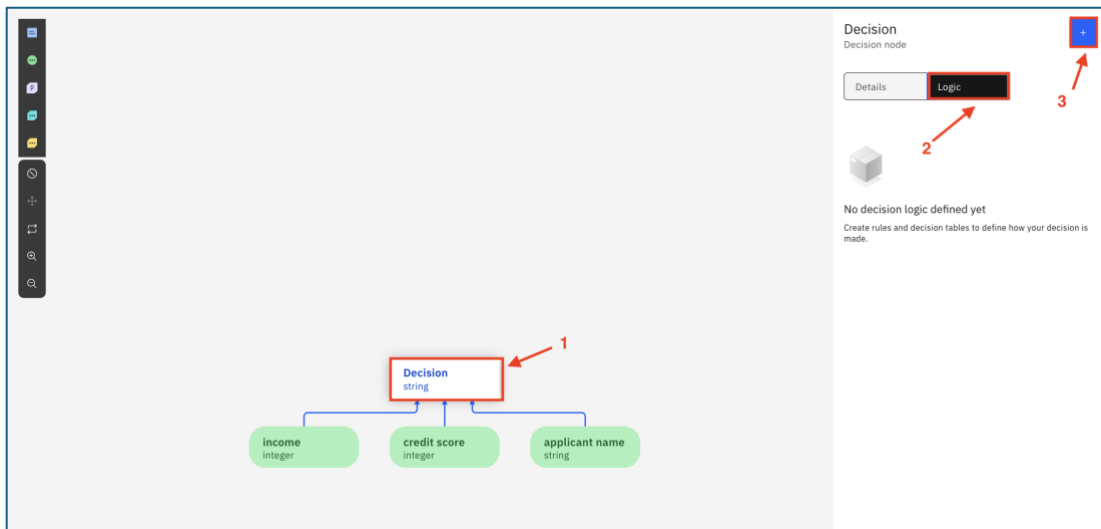
6. Your decision diagram should now look like this:



Note: this is an example created for the purposes of our hands-on lab. In a real business scenario, you would use additional parameters e.g. SSN, loan purpose, whether an applicant is employed, etc. and you could create additional rules.

Step 1.2: Create the Rules

1. You will now add some rules to the decision node. The first rule that you will add is a **default rule**. It will initialize the outcome of the decision to approved. The other rules will then specify the conditions under which the loan is declined. To create the default rule, click on the **decision node**, click the **+** button and select **Default rule**.



2. Click on the **string placeholder**, select **string** in the dropdown, and enter **approved**. You just created your first rule!

The screenshot shows the IBM Decision Center interface with the 'Logic' tab selected. The main workspace displays a rule configuration area with a dropdown menu set to 'output-default-setting'. Below the dropdown, a text area contains the rule: 'set decision to "approved" ;'. A table below the text area lists the inputs and output for the rule.

Inputs (3)	Output (1)
applicant name	string
income	integer
credit score	integer

On the right, a sidebar titled 'Decision' shows a 'Details' tab and a 'Logic' tab. The 'Logic' tab is selected, showing a message: 'Rules are applied in sequence'. Below this, a search bar and a list of rules are visible, including 'output-default-setting'.

- Now, let's add another one to state that any applicant with a credit score under 600 will be declined. Click again on the + sign and select **Business rule**. Give your rule a name, **decline low credit score**, select **credit score** in the criteria choices and click **create**.

Create business rule

decline low credit score

Select the criteria for your rule

^

applicant name

☐

'applicant name'

string

^

income

☐

'income'

integer

^

credit score

☒

'credit score'

integer

Preview your rule

```
if
  'credit score' is at least <min> and less than <max>
then
  set decision to <a_string> ;
```

Cancel

Create

- Update the rule template so that your rule looks like this:

decline low credit score ^

Type your rule using the list below as reference

```
1 if
2   'credit score' is less than 600
3 then
4   set decision to "declined" ;
```

- Add another rule to decline if income is below a certain threshold (e.g. 50,000):

decline low income ^

Name

decline low income

Description (optional)

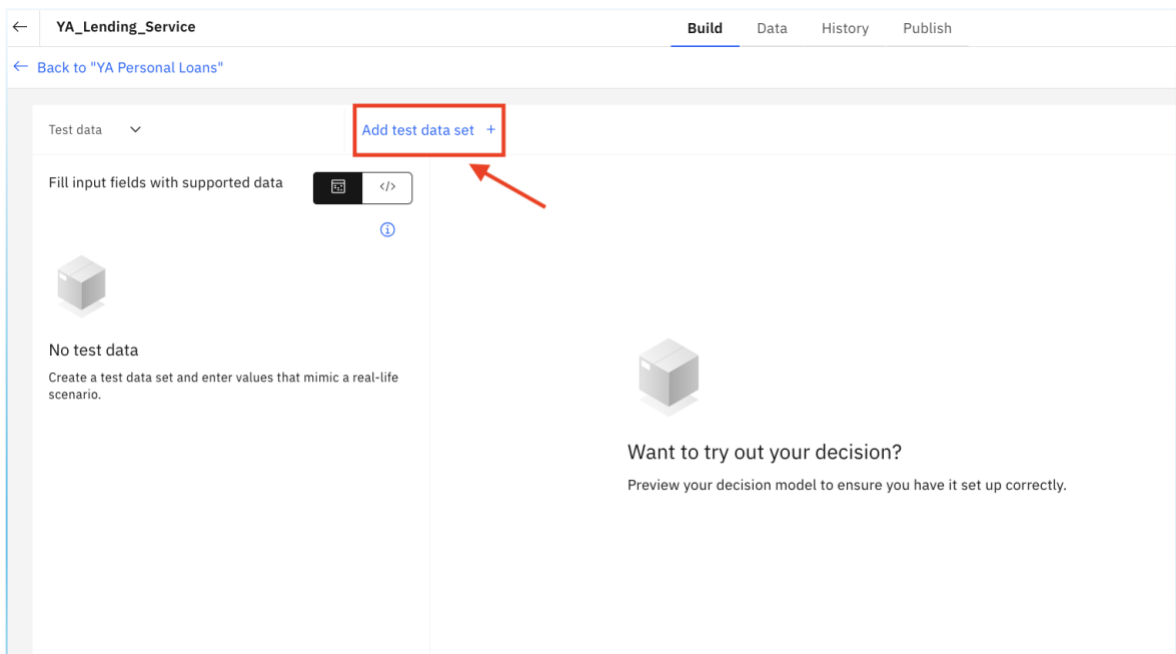
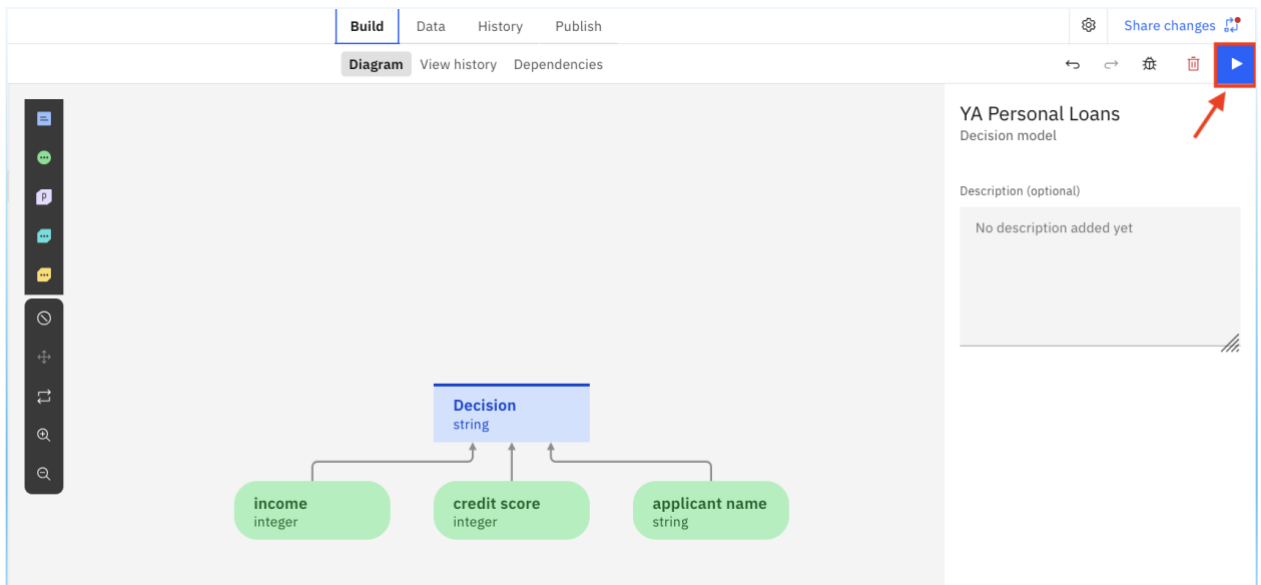
Describe this rule

Type your rule using the list below as reference

```
1 if
2   income is less than 50000
3 then
4   set decision to "declined" ;
```


Step 1.3: Test the Decision

1. You can test your rule by clicking on the **Preview** button and then selecting **Add test data set**:



2. Enter data for the input fields and click the Preview button to see the outcome of the decision. For example, enter a value of **500** for the credit score, and you should get a declined output:

← YA_Lending_Service Build Data History Publish Share changes

← Back to "YA Personal Loans"

Test data dataset ▾

dataset
Fill input fields with supported data ⓘ

creditScore
500
applicantName +
income +

dataset
12/2/2024, 4:36:12 PM

Decision output

Node Name	Result
Decision	"declined"

Messages

Message	Node name	Rule name
---------	-----------	-----------

Preview history

Node	Rules	Rule Interaction	Output
Decision	1	Sequence	"declined"
credit score	1	Not applicable	500
income	1	Not applicable	null
applicant name	1	Not applicable	null

Preview ▶

Show JSON output

Step 1.4: Deploy the decision service

1. Go back into our decision model and click on the crossed out eye icon beside your decision model name, to make your skill public.

← YA_Lending_Service Build Data History Publish

× YA Personal Loans Diagram View history Dependencies

Search for skills

Click to change the skill from private to public.

YA Personal Loans

Workflows

Generative AI

Decision string

income integer

credit score integer

applicant name string

2.

3. Share your changes, by clicking on the **Share changes** button.

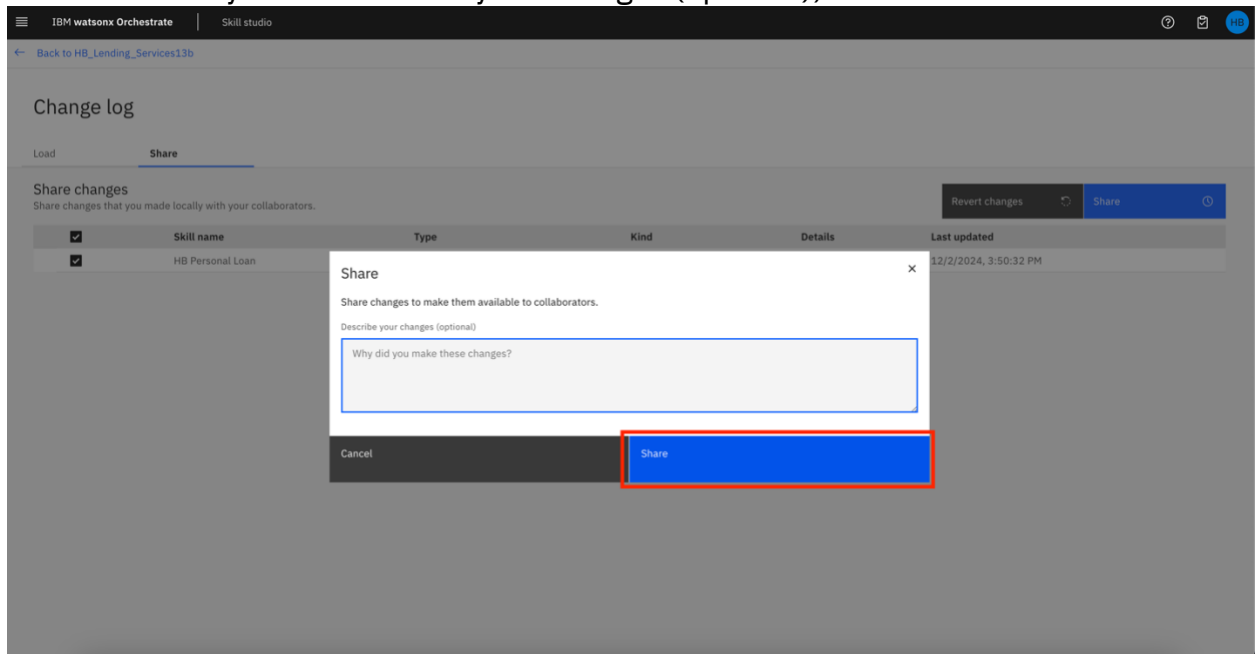
The screenshot shows the 'Build' tab in the IBM Watsonx Orchestrate interface. At the top, there are tabs for 'Build', 'Data', 'History', and 'Publish'. Below these, there are sub-tabs for 'Diagram', 'View history', and 'Dependencies'. The 'Diagram' sub-tab is active, displaying a decision model diagram. The diagram consists of a central blue box labeled 'Decision string' with three green boxes below it: 'income integer', 'credit score integer', and 'applicant name string'. Arrows point from each of these green boxes to the 'Decision string' box. On the right side of the interface, there is a panel for 'YA Personal Loans' with a 'Description (optional)' section. In the top right corner, there is a 'Share changes' button, which is highlighted with a red rectangle and a red arrow pointing to it.

4. Click on **Share**

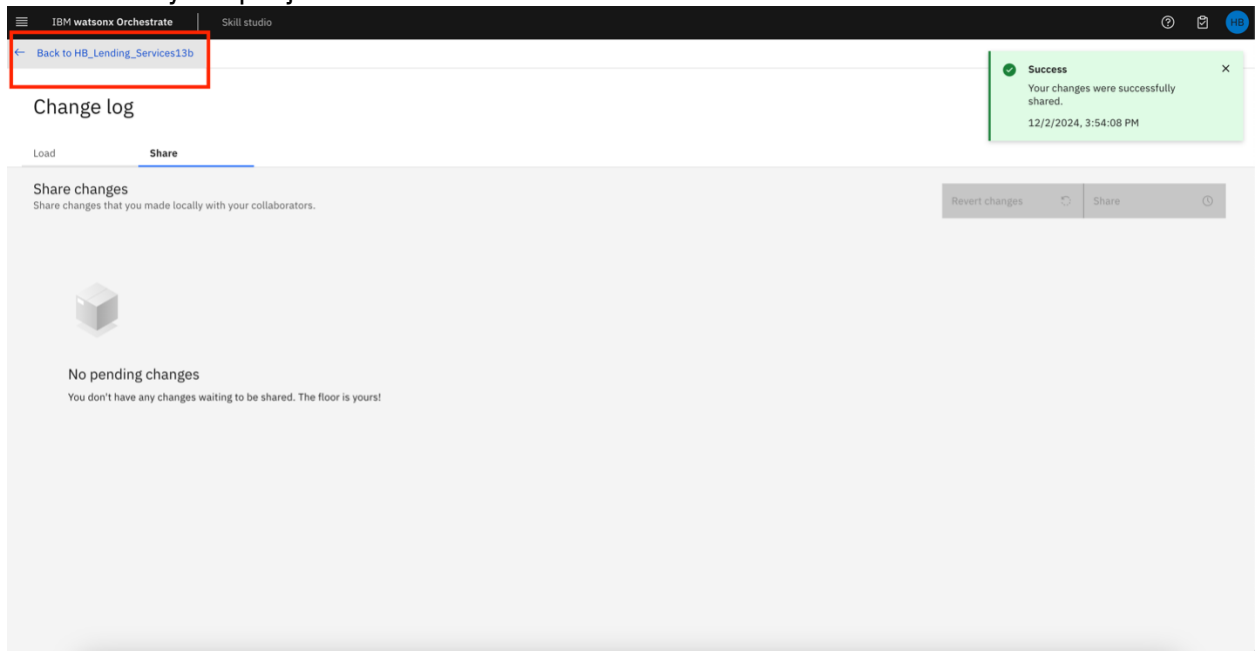
The screenshot shows the 'Change log' page in the IBM Watsonx Orchestrate interface. The page has a header with 'IBM watsonx Orchestrate' and 'Skill studio'. Below the header, there is a 'Change log' section with tabs for 'Load' and 'Share'. The 'Share' tab is active. In the 'Share' tab, there is a 'Share changes' section with a 'Revert changes' button and a 'Share' button, which is highlighted with a red rectangle. Below this, there is a table with the following data:

	Skill name	Type	Kind	Details	Last updated
<input checked="" type="checkbox"/>	HB Personal Loan	Decision model	Skill added		12/2/2024, 3:50:32 PM

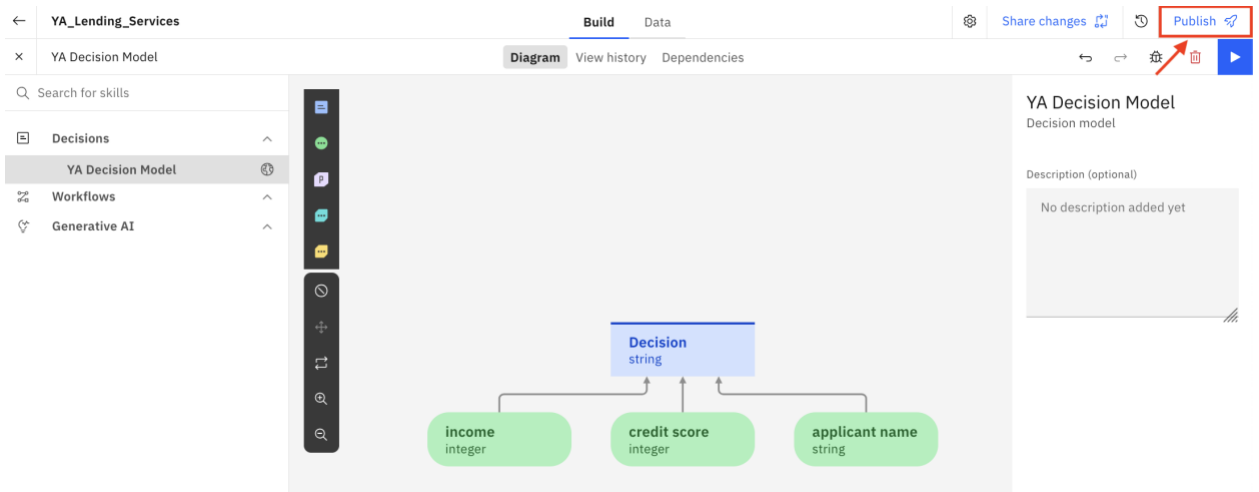
5. In a few words you can describe your changes (optional), then select **Share**



6. Go back to your project



7. Click on **Publish** in the top right corner:



8. Provide a version name and click on **Create version and publish**:

Version and publish



Create a version and publish it to the catalog.

Version name

v1

Comments (optional)

Enter details about this version

Skill visibility



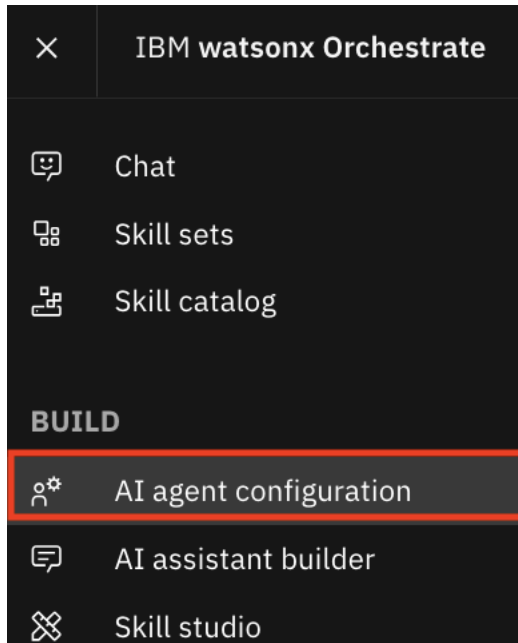
Cancel

Create version and publish

9. Your decision service is now published and operational

Step 1.5: Test Decision Flow in AI Chat

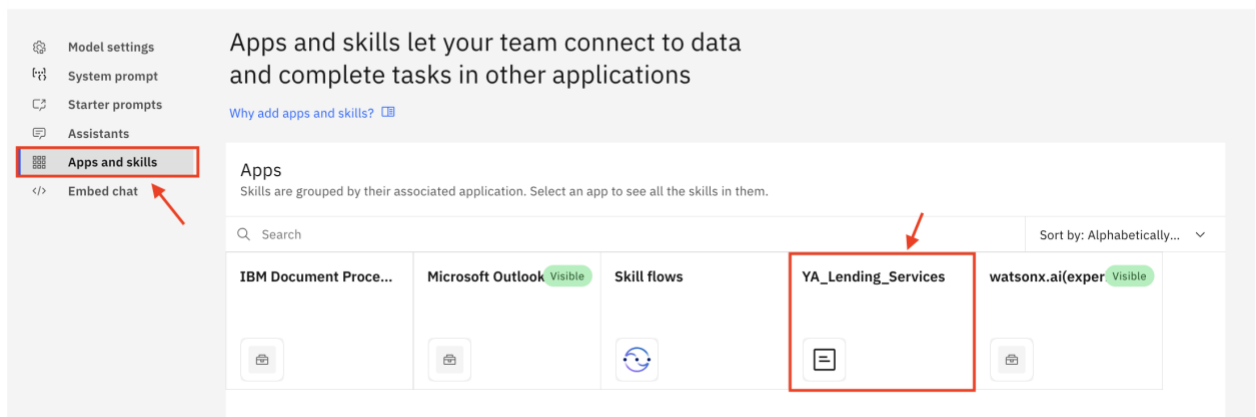
1. Navigate to **AI agent configuration** from the hamburger menu:



2. Select **Apps and skills**, then click on the tile with your decision application (**YourInitials_Lending_Services**):

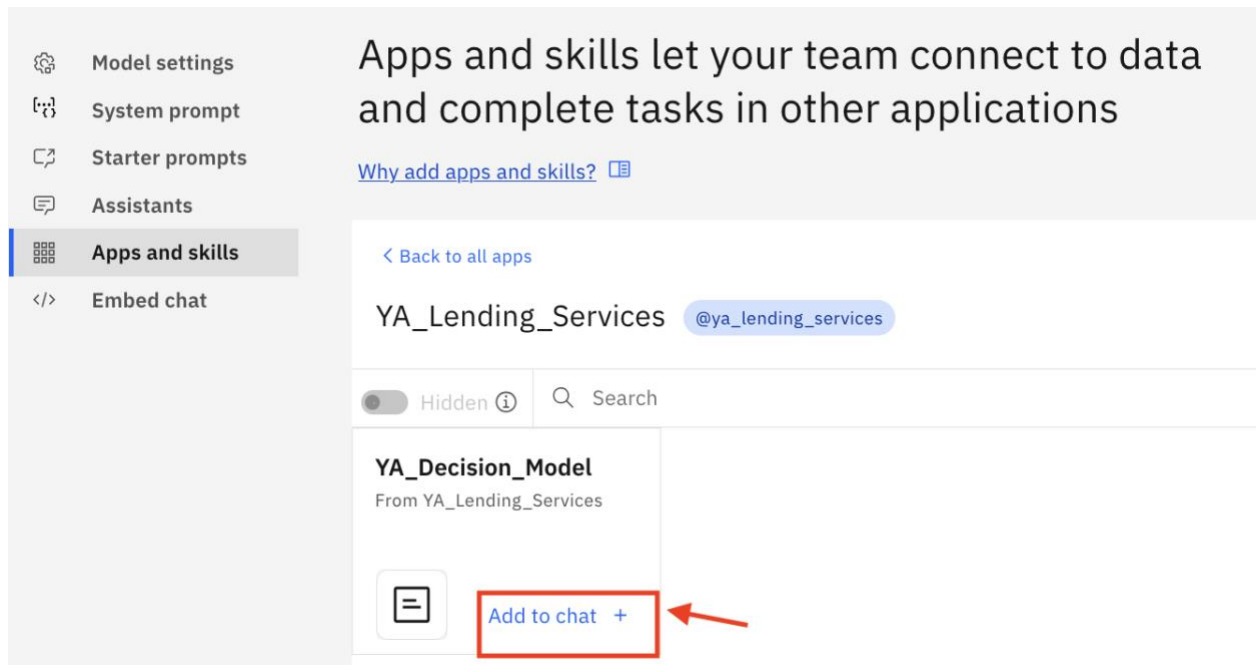
AI agent configuration

Agent configuration allows users to customize their chat experience according to their preferences.




Note: Note that your application is automatically available under **Apps and skills** in the AI Agent configuration. This is true for any published automation-based skill. If you need to add a built-in skill or custom skill you built, you will need to connect to the application from the **Main Menu -> Skill sets** first (if you are working in wxO in IBM Cloud). If you are on an AWS tenant, go under Manage Team -> Skill sets.

3. Click on **Add to chat+** to add the skill to the AI chat:




4. Enter a routing description and click on **Add skill:**



YA_Decision_Model

from YA_Lending_Services



An accurate description of the purpose and capabilities of the skill is required for the AI model to successfully activate this skill in the chat.

Routing description

Description of skill capabilities59/500

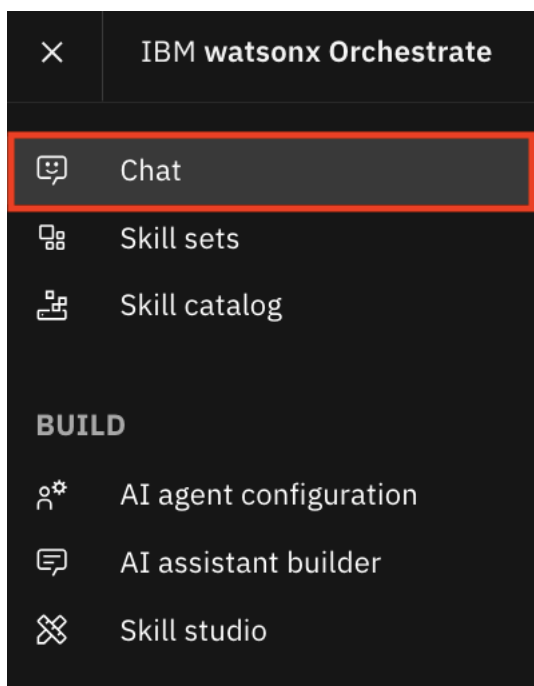
process a loan application based on income and credit score

Tip: [What makes a good description?](#)

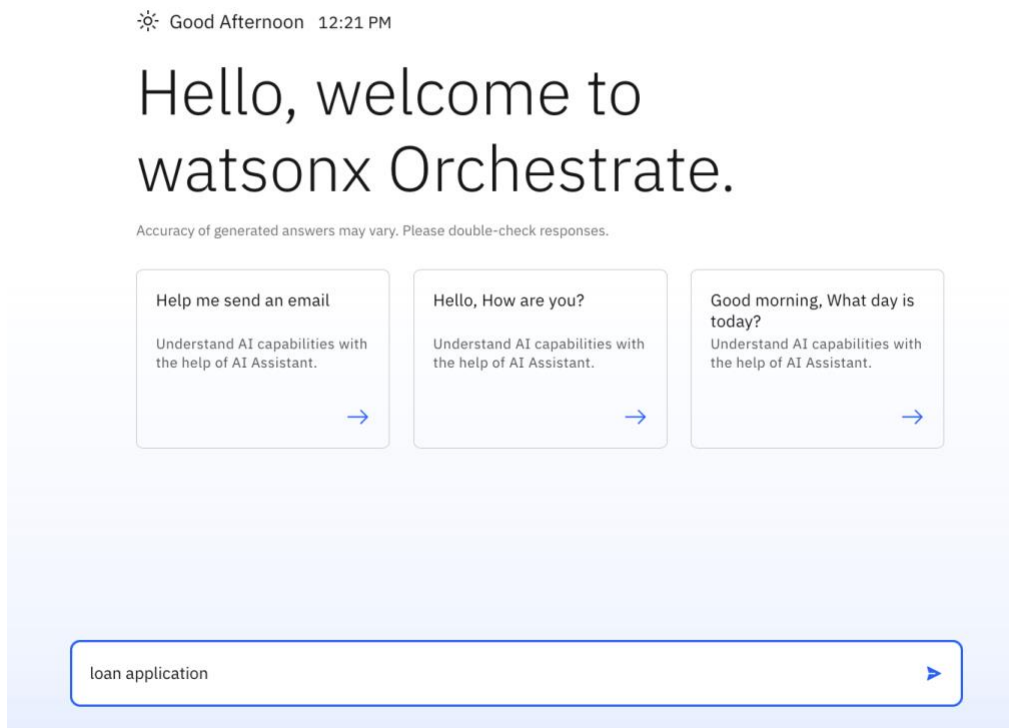
Cancel

Add skill

5. Navigate to **Chat** from the menu:




6. Type your query, e.g. **loan application** in the chat window:



7. The agent routes your query to the decision automation skill as expected:

You 12:22 PM


loan application

 watsonx 12:23 PM

Enter a value for income

You 12:23 PM


60000

 watsonx 12:23 PM

Enter a value for creditScore

You 12:23 PM


500

 watsonx 12:23 PM

Enter a value for applicantName

You 12:23 PM

John Smith

 watsonx 12:23 PM


Please confirm:
income: 60000
creditScore: 500
applicantName: John Smith

Yes

No

You 12:23 PM

Yes

 watsonx 12:23 PM

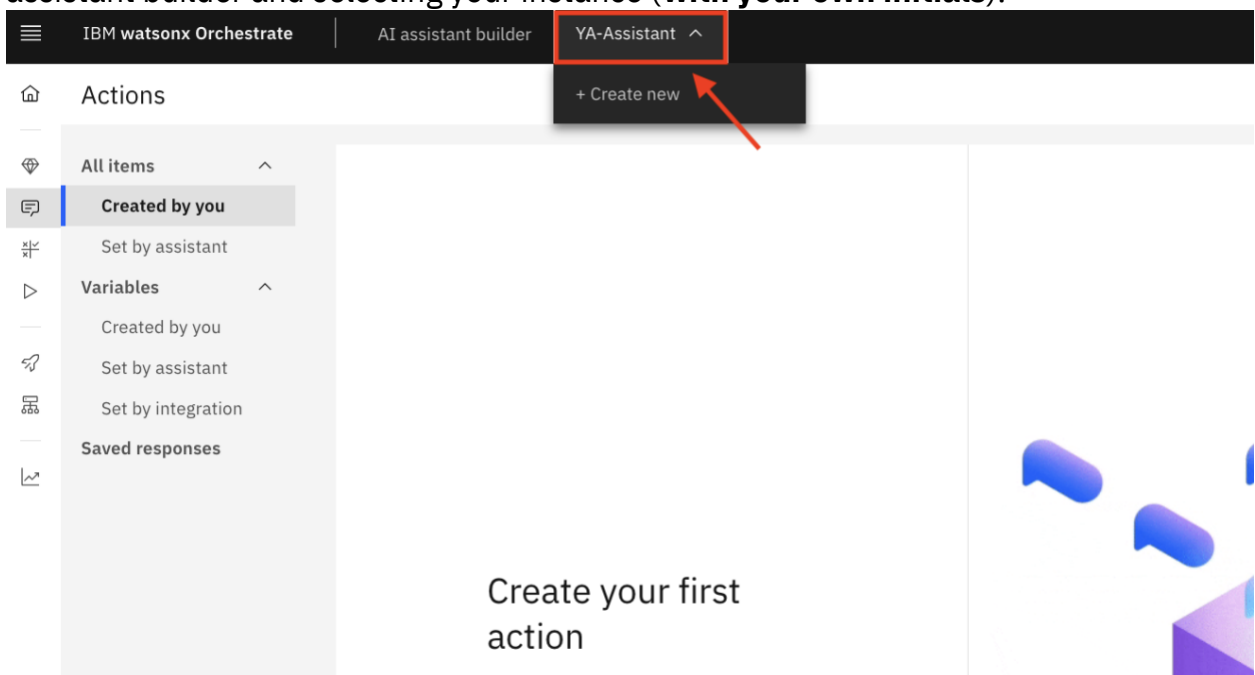
declined

The application is declined, as expected, since the credit score is < 600

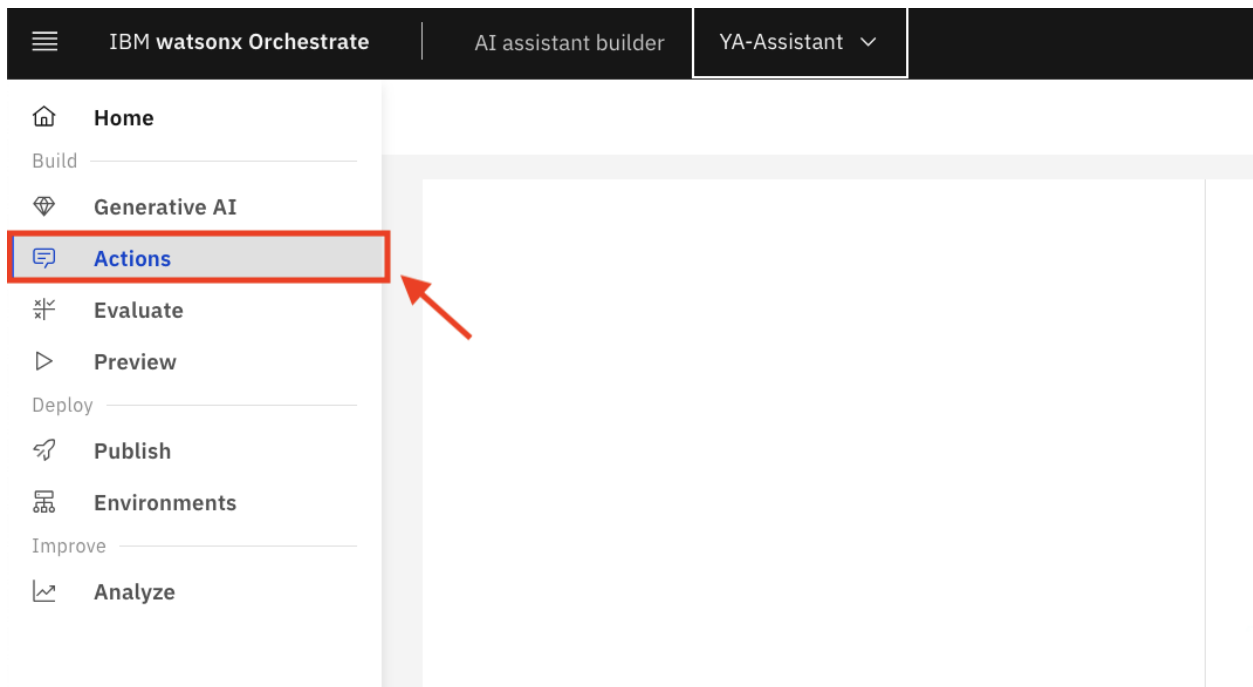
Step 1.6: Import the decision skill as an action in the assistant

As you saw in the previous step, you can bring an automation-based skill directly into the AI chat. Another option, if you want to make this skill a part of a conversation flow, is to make it available in an assistant. For example, we can use the assistant we created in lab 2 and create a skill-based action using the decision automation we just built. Once our assistant is implemented and published, we can add it to the AI chat instead of adding an individual skill (see Lab 4).

1. Open the assistant builder instance you created earlier by going into the AI assistant builder and selecting your instance (**with your own initials**):



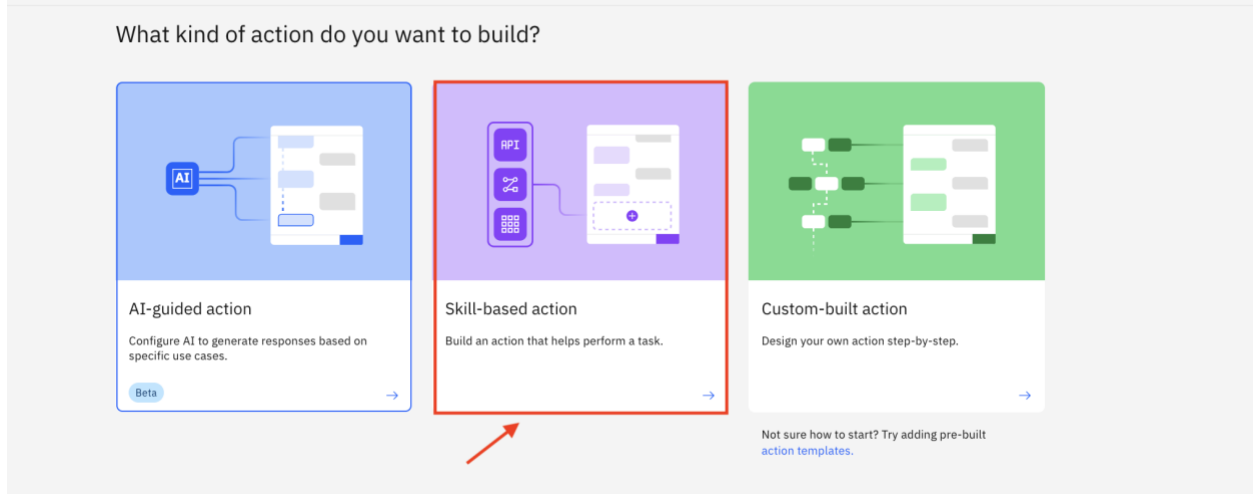
2. Open the actions tab:



3. Create a new action:

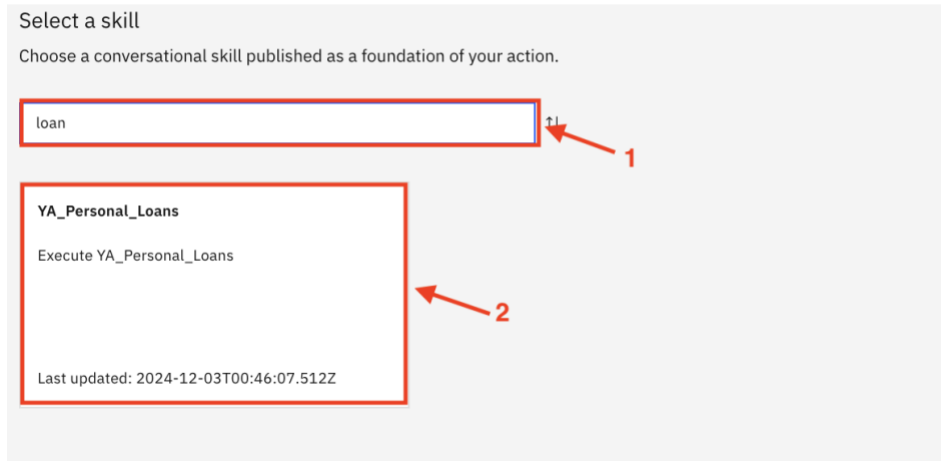


4. And make it a skill-based action:



5. Search for your decision skill and select it:

Build an action from a skill



6. Click **Next:**

Build an action from a skill

Cancel Next

Select a skill

Choose a conversational skill published as a foundation of your action.

loan

↑↓

YA_Personal_Loans



Execute YA_Personal_Loans

Last updated: 2024-12-03T00:46:07.512Z

7. Provide an utterance (a phrase a user might type to invoke the action), e.g. **Apply for a loan** and click on **Save**:

New action



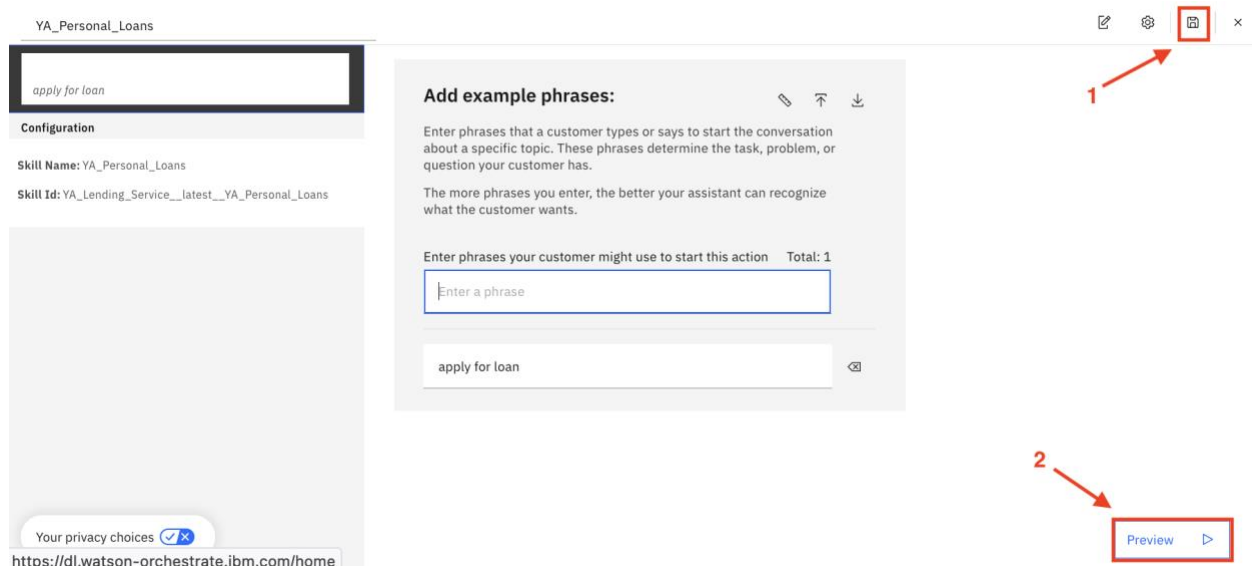
What does your customer say to start this interaction?

apply for loan

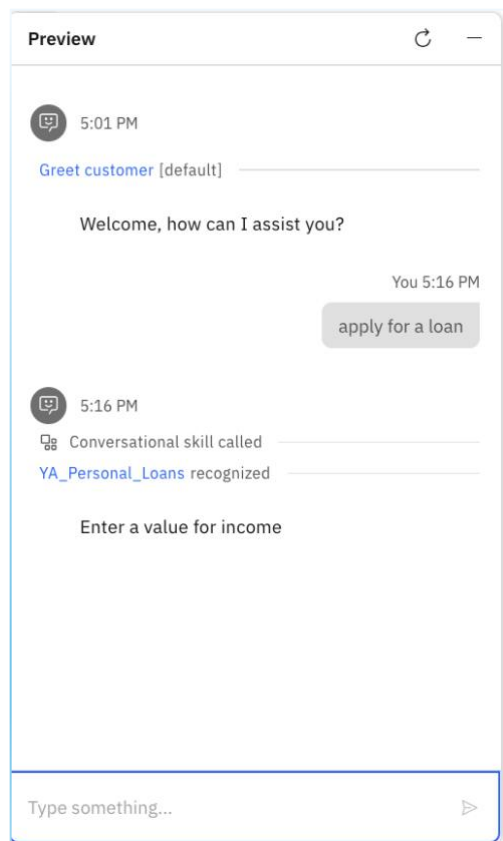
Cancel

Save

8. Click on **Save** and then on **Preview**:



9. Once the changes have been added, type the suggested utterance in the assistant preview window and hit enter. You will now be asked to provide some of the information required as input for the decision automation:





10. Provide answers as follows:

Income = 100,000

Credit score = 500

Your name


The conversation should look similar to:


Preview  

Enter a value for income

You 5:19 PM

100000


 5:19 PM


 Conversational skill called _____

Enter a value for creditScore

You 5:19 PM

500

 5:19 PM

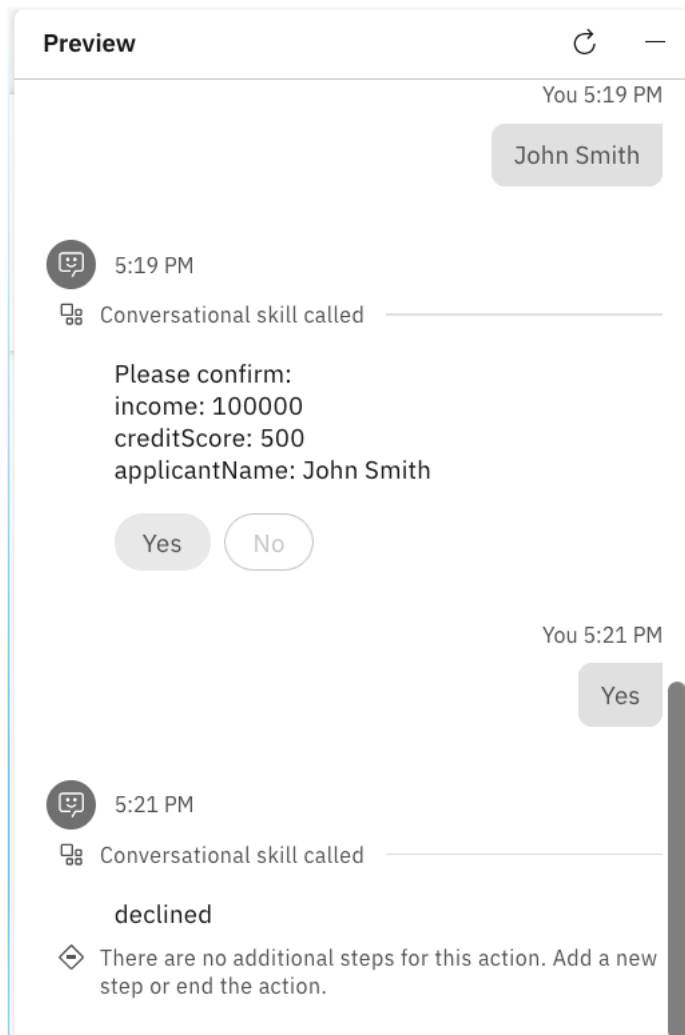
 Conversational skill called _____

Enter a value for applicantName

You 5:19 PM

John Smith

11. At this point you will be asked to confirm the entered data, and you will be notified of the loan approval decision:



Now you have implemented a skill-based action in your assistant. It uses the decision automation skill you built earlier for the loan approval process. Of course, this is just a very simple version and much more can be done in watsonx Orchestrate! You can also bring other available skills into your assistant. Additionally, it is very easy to further configure how the questions and answers are displayed in the chat.

This concludes the lab