IBM Cloud Pak for Business Automation Demos and Labs 2022

IBM Automation Decision Services
Manage Decisions and infuse Machine Learning

V 1.0

Decisions
dba_decisions_l3_services@wwpdl.vnet.ibm.com

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

In this Lab, you build business operational decisions to implement services evaluation and client risk assessment for a Client Onboarding solution. You capture and automate these decisions using IBM Automation Decision Services.

It includes three exercises. Each exercise can be done separately.

Duration: About 3 hours (each exercise is about 1 hour).

Audience: Anyone who wants to learn how to use Automation Decision Services.

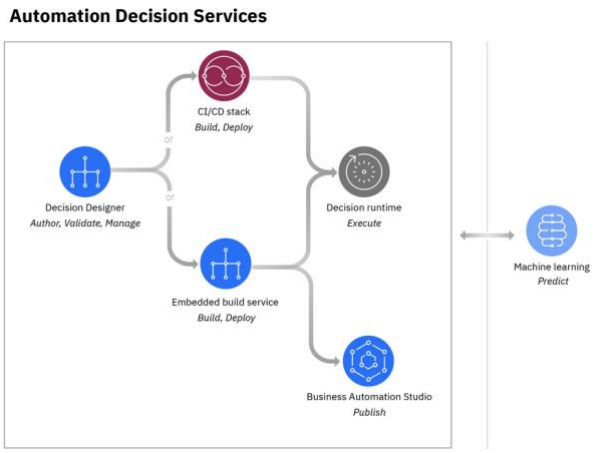
1.1 IBM Automation Decision Services

IBM Automation Decision Services provides a comprehensive environment for authoring, managing, and running decision services.
Operational decisions apply business policies, which are often influenced by numerous factors that can be both internal and external to an organization.

Part of the IBM Cloud Pak® for Business Automation platform, Automation Decision Services delivers advanced decision automation capabilities. Using Decision Designer in Business Automation Studio, business experts can model, author, and validate decisions in one development environment. They can also infuse intelligence into business decisions by combining decision models and predictive models into decision services. They can collaborate with others in their organization by sharing projects through a central Git repository.

Automation Decision Services integrates with a continuous integration and delivery (CI/CD) stack.

You can build and deploy decision services directly from Decision Designer. Deployed decision services can then either be published as automation services in Business Automation Studio, or invoked through the decision runtime.



For more information, see IBM Documentation
IBM Automation Decision Services
What is Automation Decision Services

1.2 Lab Overview

The end-to-end Client Onboarding solution explores a generic use case for onboarding new services to an existing client. Automation Decisions Services provides two decisions services that are integrated in the solution.

The first decision is to define the fees for the services the client is being onboarded to, and to suggest additional services the client could be interested in, based on the information gathered. It is consumed in the end-to-end solution via an automation service from a Business Automation Application in the front-office in-take app.

The second decision provides input to determine the onboarding approval, the scoreboard. It implements a client risk evaluation based on multiple information such as the client profile, industry and a client classification based on the client revenue. It combines descriptive and predictive decisions. It is consumed in the end-to-end solution as an automated service in Workflow, during the approval step.

In this Lab you learn how to:

1. Manage business decisions based on multiple data: client, industry information.
2. Infuse intelligence into business decisions by adding a predictive model into the decision service for the scoreboard of the client.
3. Collaborate by sharing projects through a GitHub repository, build and deploy decision services from Decision Designer. Learn how to publish automation services in Business Automation Studio.

Integrating the automated services in other applications is not covered in this Lab, look at the Workflow and Business Automation Application labs for this.

1.3 Lab Setup Instructions

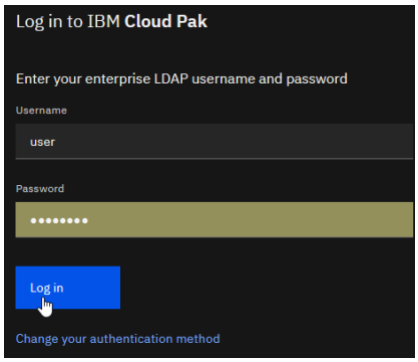
1.3.1 Prerequisites

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** to access Decision Designer.
   - **ADS ML Service** (Machine learning providers details).
2. Download the file **ClientOnboardingLab.zip** from the Lab Data folder onto your computer.

   GitHub account is needed for Exercise 3. You use a GitHub account to create a repository. If you do not have an account, create an account in GitHub. Your work will be saved in your repository.

1.3.2 Log in to the environment

1. Launch Business Automation Studio (URL listed in the Lab access document, see prerequisites)
2. Select your authentication type: **Enterprise LDAP**
3. Enter your *Username* and *Password* then click on **Log in**



2 Exercise 1: Modeling Decisions

2.1 Introduction

This exercise is an introduction to the authoring environment of Automation Decision Services. You navigate in Decision Designer which is the development environment for creating decision services. You discover the main concepts of Automation Decision Services by exploring and modifying a given decision service. The scenario of this decision service is to define the fees for services the client is being onboarded to and to suggest additional services the client could be interested in, based on the information gathered.

2.2 Exercise Instructions

In this exercise you import a decision service and explore it:

- Create a **project** and import a **decision service**
- Explore the **data model** and the **decision model**
- Add a business rule and edit a decision table to update the **logic** used in the model
- Validate the decision service

Prerequisites: you need to download the provided file **ClientOnboardingLab.zip** from Lab data folder.

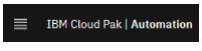
2.2.1 Creating a project and importing a decision service

In this step, you create a project in Business Automation Studio. You import a decision service from the zip prepared for this exercise.

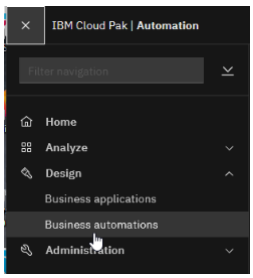
A project is a set of artifacts that share the same lifecycle and are grouped to solve a particular business problem. In this Lab you work with a decision automation project, it provides decision-modeling capabilities to help business experts to capture and automate repeatable decisions.

If you are already connected to Automation Decision Services and have your project opened following a previous exercise, you can skip this section and pursue in next section Exploring the decision service: data model and decision model.

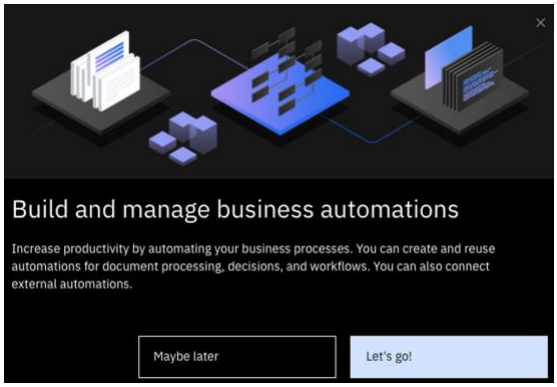
If you begin the Lab start with this exercise, follow the steps below.

1. Log in to Business Automation Studio.
2. Click on the Navigation Menu on the top left corner 

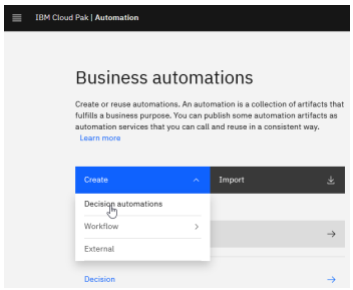
.
3. Expand Design, and then click on Business automations.



4. An invite to a guided to a guided tour to Business automations may be presented. You can choose to follow it or select Maybe later.

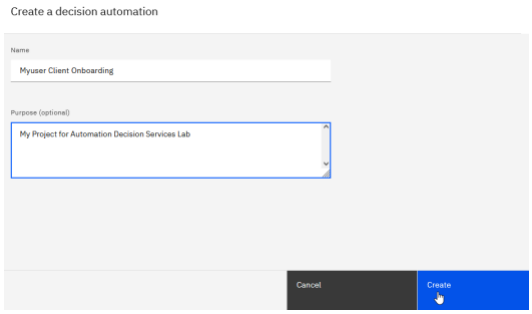


5. Click on **Create** and select Decision automations.

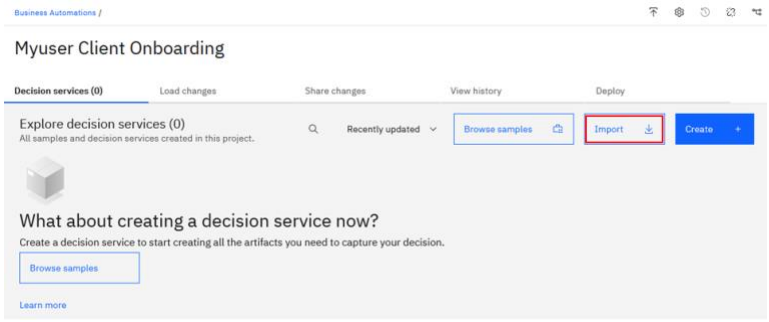


6. Enter a name for your project. Enter **UsrNNN Client Onboarding Decision** where *UserNNN* is your assigned username (a name starting by your username to avoid conflicts with other projects).

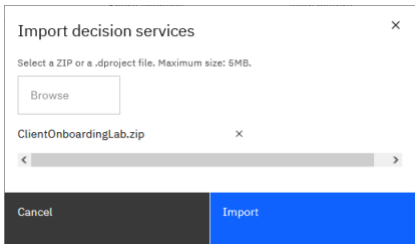
7. Click on **Create**



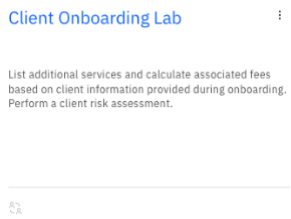
8. Your new project opens in Decision Designer. Click on **Import** to import the decision service provided by the Lab team.



9. Browse to select the project prepared for the Lab **ClientOnboardingLab.zip** and click on **Import**



10. A tile of a decision service named **Client Onboarding Lab** appears on the project page.



A decision project can be composed of several decision services. You create them or import them from the samples or from a zip file. A project can be shared with other users and can be connected to a Git repository. These features will be covered in Exercise 3.

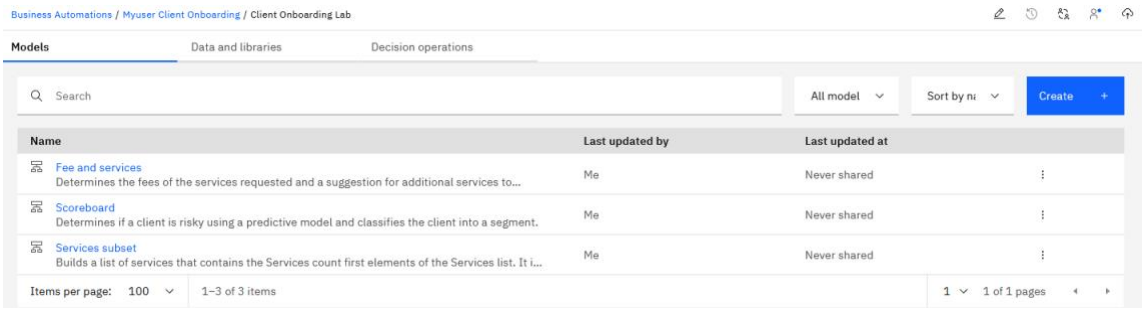
2.2.2 Exploring the decision service: data model and decision model

In this step, you explore the decision service you imported. A decision service uses decision artifacts to define the business decision.

- **Decision models**: Represent a diagram that expresses the business logic. You can reuse the output of a decision model in another decision model.
- **Predictive models**: Apply data from a machine learning model to make a prediction.
- **Data model**: Represents the data structure used by the business logic. You can use the data model vocabulary in your decision models and predictive models.
- **External libraries and data sources**: To extend the data models using java code or contain data models and functions that you can use in decision models.
- **Decision operations**: Define entry points for decision services execution.
- **Task models**: Lets business experts define the decision logic outside the diagram (Note: task models are not covered in this lab, but you can see them in the samples proposed in Additional information)

For more information, see IBM Documentation Building decision models

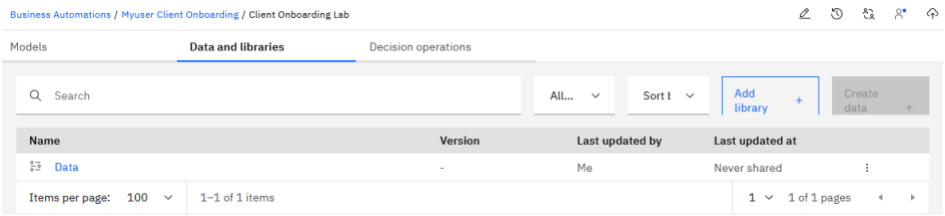
1. Click on **Client Onboarding Lab** to open your decision service.
2. Your decision service is displayed.



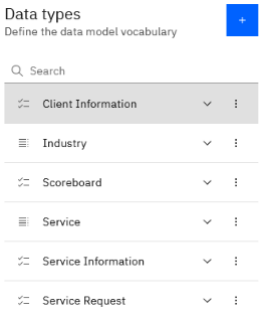
2.2.2.1 Exploring the data model

A data model defines the data that is needed to make your decisions. In a data model, you use predefined and custom data types to create a vocabulary that you can use to populate decisions. Basic data types, including string and integer are predefined by default in Automation Decision Services. You create a data model to define custom data types to match specific needs for your service. These types are defined in a data model and can be used in one or more decision models in a service.

1. Open the Data and libraries tab and click on Data



The data model is a collection of **Data types** that represents the data needed to make the decision. This collection is used by all the decision models defined in the decision service.
This model for Client Onboarding contains 6 **Data types**: Client Information, Industry, Scoreboard, Service, Service Information, Service Request.



In the decision model for this exercise, 5 of these data types are used: Client Information, Industry, Service, Service Information and Service Request.

Two categories of data types can be defined:

- the **enumeration type** such as Industry and Service that contains a list of possible values.
- the **composite type** such as Client Information, Service, Service Information, Service Request, that contain a set of attributes with a name and a type. The type can be a basic type predefined in Automation Decision Services such as string or integer or a custom data type.

The types defined as input of the model for this exercise (Fee and services) are:

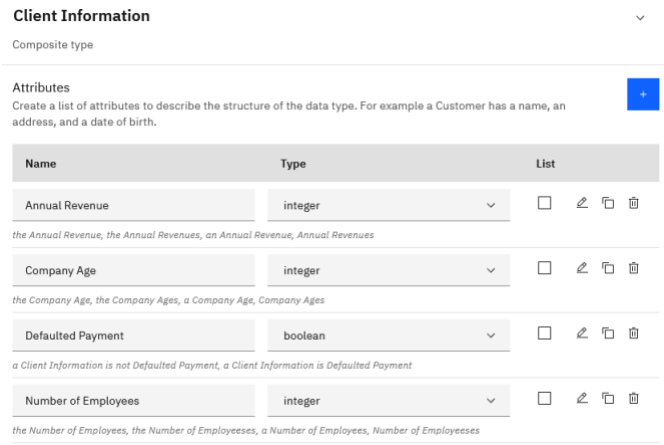
- the enumeration **Industry**: lists the available industries
- the enumeration **Service**: lists the available services
- the composite type **Service request**: describes in which industry the client is and what are the services requested
- the composite type **Client Information**: describes the characteristics of the client: annual revenue, number of employees, company age, defaulted payment.

The type defined as output of the model for this exercise (Fee and services) is:

- Service Information: gives a list of additional services suggested to the client and the fee for the services requested.

Automation Decision Services allows you to import sets of values extracted from an Excel file to populate enumeration data types. This is not covered in this Lab. For more information, see IBM Documentation Extracting values from an external data source

2. Click on **Client Information** to explore the data types. It is a Composite type composed of 4 **attributes**. Attributes and values allow you to define the characteristics of a data type

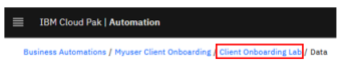


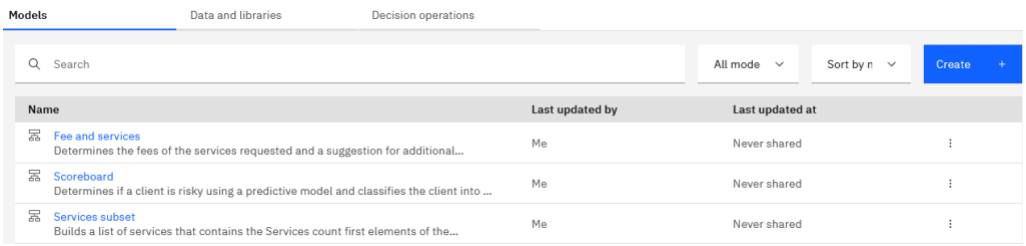
The **verbalization** defines how a data type is referenced in business rules. It depends on the type of element. Each attribute comes with a set of automatically generated expression and action phrases.
For more information, see IBM Documentation Default verbalization

3. Explore each **data type** and **attributes** to learn more about the data types.

   For more information, see IBM Documentation Creating a data model

4. Click on **Client Onboarding Lab** in the breadcrumb to navigate back to your decision service.





Working with external libraries to enrich your data model is not covered in the Lab. For more information see IBM Documentation Working with external libraries, or the tutorial Using and external library tutorial available on GitHub.

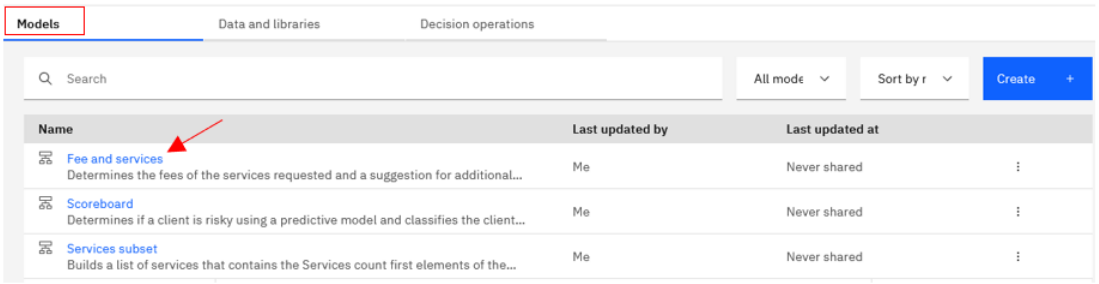
2.2.2.2 Exploring the decision model

You define the structure of a decision in a diagram, in a model. The primary part of the diagram are nodes:

- **Decision nodes**: represent the decision and contain a decision logic that defines how each decision is made.
- **Input data nodes:** represent the data used to determine the decision output. They are associated with a data type.
- **Function nodes:** represent values that are computed from other decision models.
- **Prediction nodes**: represent values that are computed in machine learning predictive models (prediction nodes are not used in this exercise, they are used in Exercise 2)
- **Links** represent the relationship between the decisions and input data, the invocation of a function or prediction

The rules define the logic of the decision and are expressed in business rules or decision tables. The rules determine the output of the model.

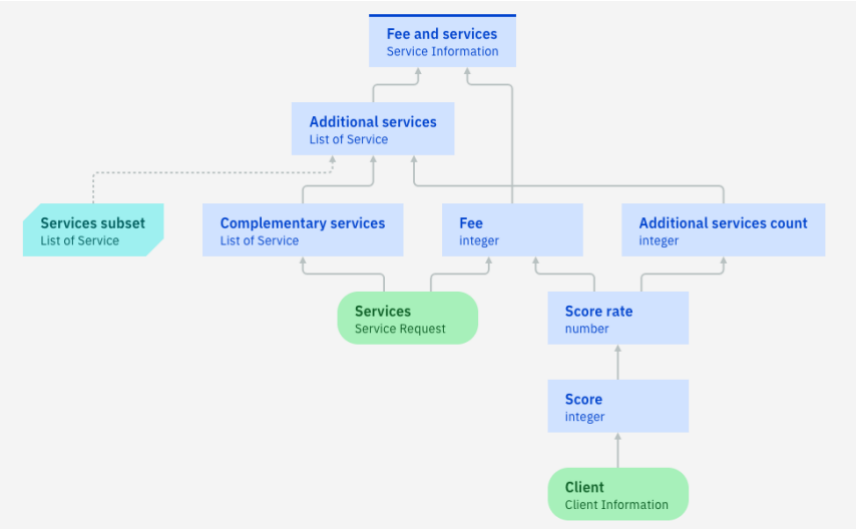
1. In the Models tab, click on **Fee and services.**



It opens the Decision model Fee and services.

In Automation Decision Services, you implement your decision by building a decision model diagram. The diagram shows how the decision depends on several sub-decisions.
The modeling diagram is based on the Decision Model and Notation (DMN) standard.

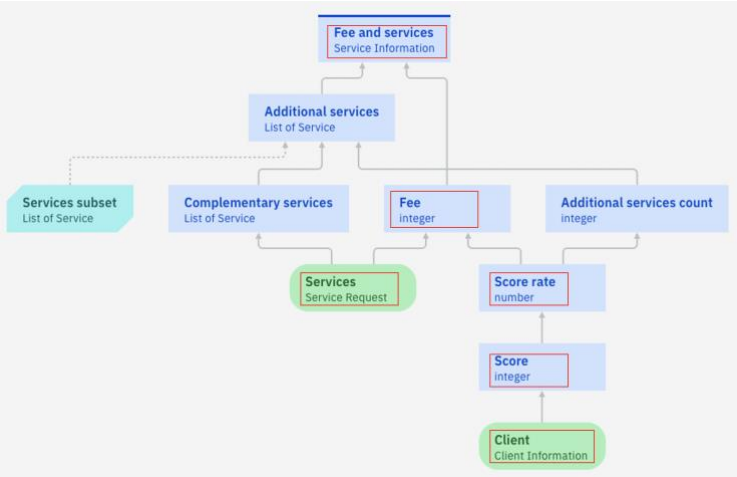
The diagram below shows the decision model for **Fee and services.** As explained in the [Exploring the decision mode introduction], the diagram is composed of nodes and links. The decision nodes are blue, input nodes are green and function nodes cyan. The links represent the relationship between the decisions and the inputs. A link shows the input that is available in a node.



In this exercise, the top decision **Fee and services** is the final decision. It calculates the fees for the services requested by the client and suggests additional services to offer.
It is based on the **Client** information and the **Services** requested as input. To provide the outcome, it depends on intermediate steps, sub-decisions.

The final decision node **Fee and services** depends on two sub-decisions: **Fee** and **Additional services.**

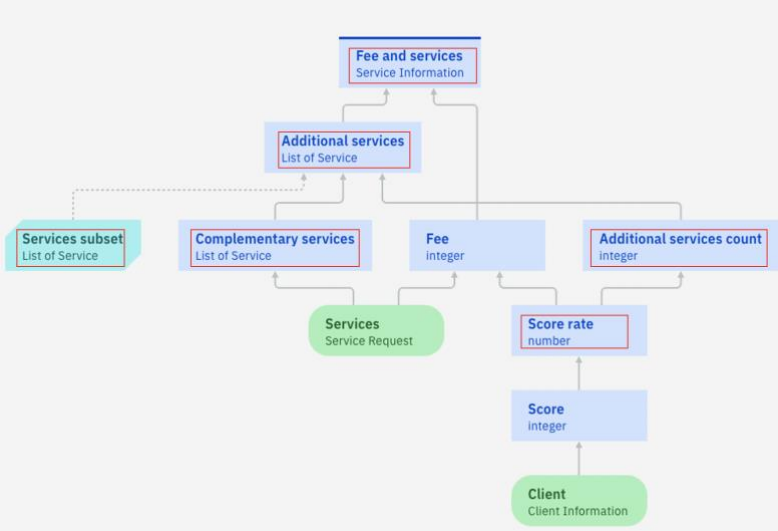
Fee
The fee is computed by the **Fee** decision node, based on the service list requested by the input node **Services** and the **Score rate** decision node.
The **Score rate** decision node computes a rate to apply to the default fee for the client based on a score computed by the **Score** decision node.
The **Score** decision node computes the client score from its different characteristics described by the **Client** input node.



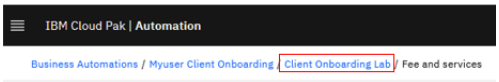
Additional services

The additional services list is computed by the **Additional services** decision node. It takes a subset of the list computed by the **Complementary services** node.

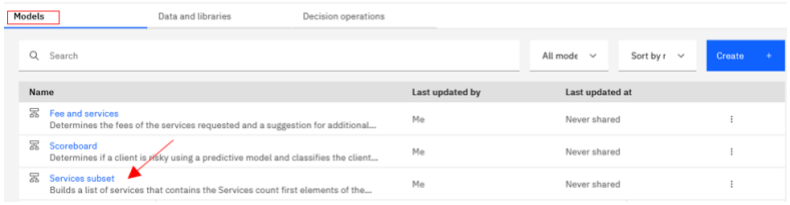
To extract the subset of services it depends on the function node **Services subset.** The subset of the list is computed by taking the count of elements from the list. In the exercise, this computation is arbitrarily simplified. The count is computed by **Additional service count** decision node from the **Score rate** decision node.

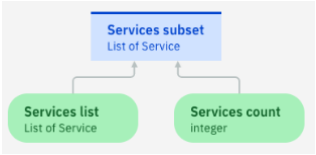


2. Services subset is a function node. It is computed from another decision model named Services subset. Click on **Client Onboarding Lab** in the breadcrumb to navigate back to the **Models** tab.



3. To see its diagram, select **Services subset** in the Models list:

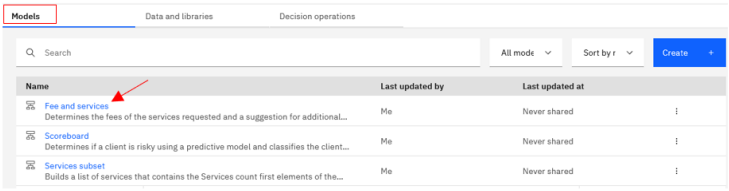




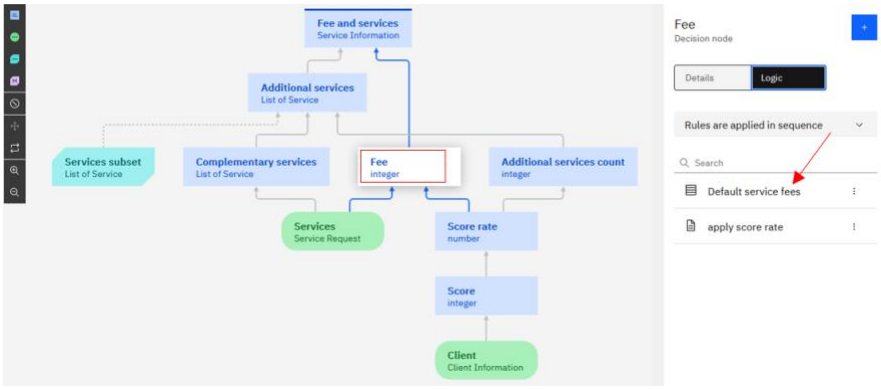
For more information, see IBM Documentation Creating decision model

4. Return to the Decision model Fee and services. Click on **Client Onboarding Lab** in the breadcrumb to navigate back to the Models tab.

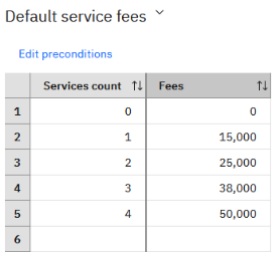
5. In the Models tab, click on **Fee and services**



6. Click on the **Fee** node, and then select the **Logic** tab to explore the decision logic. It displays the decision table **Default services fees** and the rule **apply score rate** that implement the business logic of the fee calculation. In Automation Decision Services, you express the decision logic with a set of business rules and/or decision tables.

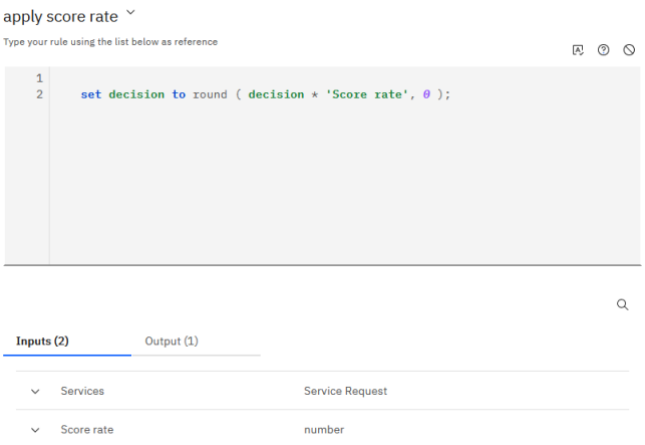


7. Click on Default services fees in the **Logic** tab. It opens the decision table.



Each row represents a single decision where the **Services count** column represents a condition parameter and the **Fees** column the value of the action when the conditions are met.

8. Click on apply score rate in the **Logic** tab. It opens the business rule.



It is a simple rule statement that determines a rate to apply to the fee calculation.

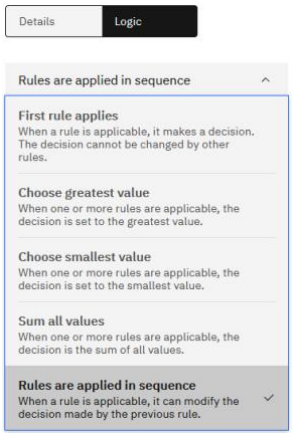
9. Click on **Inputs,** then **Output** at the bottom of the screen.

   The rule uses the vocabulary of the input data **Services** and **Score rate** and selects a value for the output of the decision **Fee**.

In this decision logic, the rules are applied in sequence. A fee is assigned according to the number of services. Then, a rate is applied to calculate the final fee for the requested services.

To define how the rules interact with each other, you select a **rule interaction policy**.

10. On the Logic tab, expand Rules are applied in sequence.



The interaction policy choices differ according to decision node type (number, list, integer...) In this exercise, the decision node type Fee is an integer, the options above are displayed for the integer type.
For more information, see IBM Documentation Choosing an interaction policies

2.2.3 Updating a decision logic: adding a node, a rule, editing a decision table

You change the decision logic for the Score computation to take into account the number of services requested by the client. A higher number of services leads to a higher score. The change in the decision logic implies:

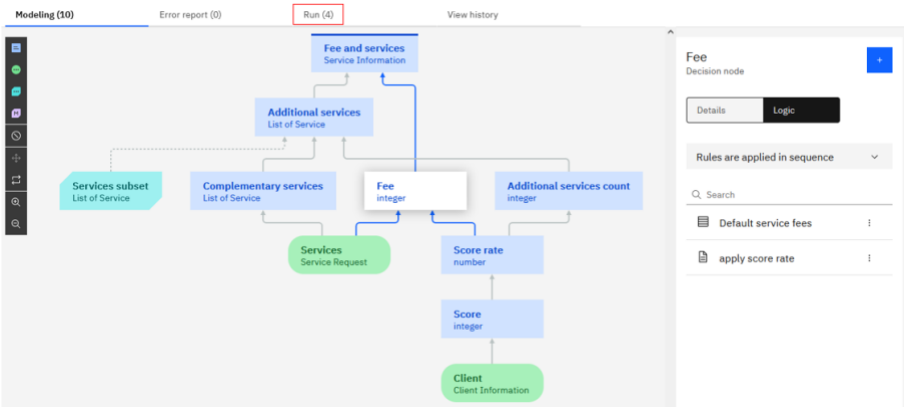
- knowing the number of requested services from the score node. That requires adding a link from the services input and the score node.
- adding the computation from the number of services. That requires adding a rule to compute the new score.
- adapting the score rate computation to the new score range.

Before modifying the decision logic, you check the current behavior of the rule execution.

2.2.3.1 Test the decision logic

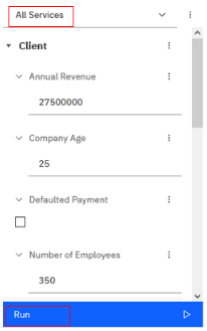
1. Click on **Back to the diagram** to return to the decision diagram

2. Click on the **Run** tab.

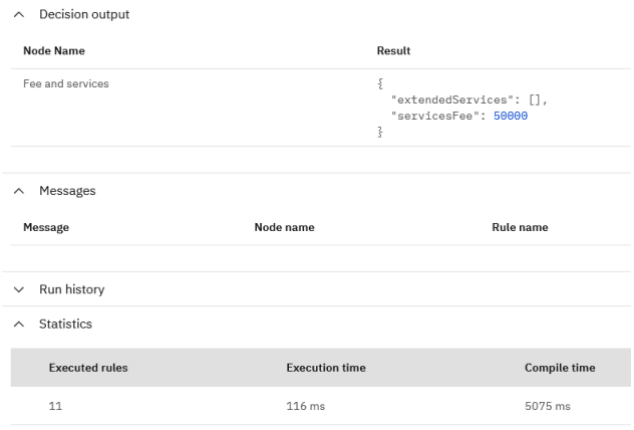


On the left side of the screen, the **Test data** pane allows to select the set of data to submit to execute the rules. In this Lab, 4 sets of data are pre-defined: All Services, Federal 2, Federal 3, Telecom 1.

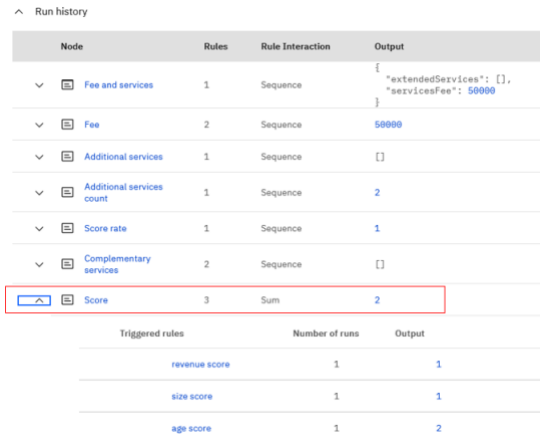
3. Select **All Services** and click on **Run.**



It generates a report and displays the input and output of each node in the decision model. The final decision is displayed in the top of the report: for Fee and services the result is "servicesFee": 50000.



4. Expand **Run history**, and then expand **Score** node



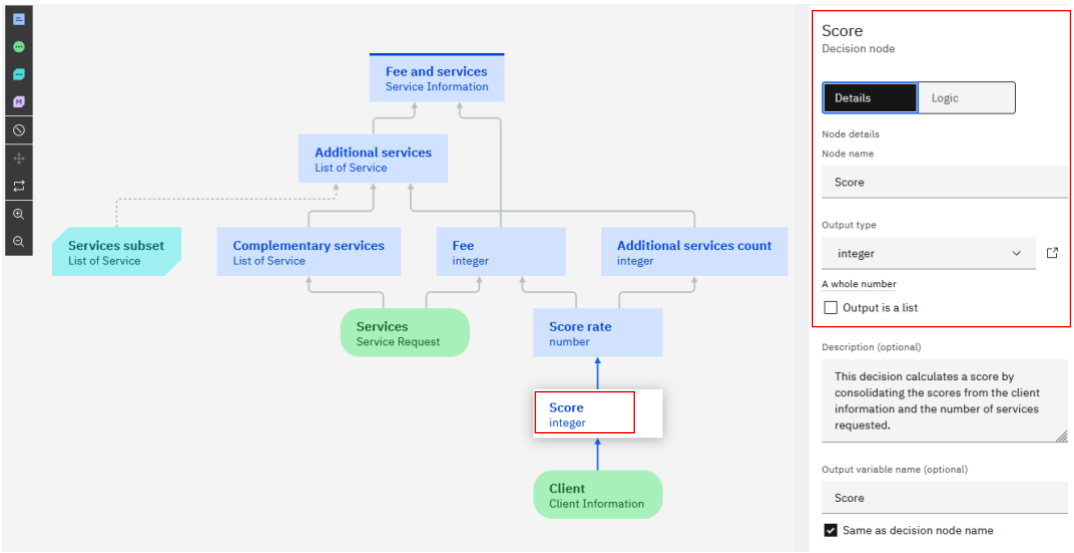
The output displays the result for the decision node Score, it is a consolidated result. In this example, Score decision node output is 2:
- revenue score output is 1,
- then added to size score (size score is 0) it remains 1,
- then added to age score result is 2 (age score is 1).

5. Click on the **Modeling** tab to return to your decision

2.2.3.2 Exploring the current logic

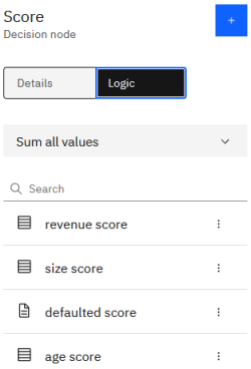
The score node returns an integer that is used by the score rate node to compute a rate from the score.

1. Click on the **Score** node, and then the **Details** tab.



It returns an integer computed by adding all the values returned by its defined rules and decision tables.

2. Click on the **Logic** tab to explore the logic of the business rules.



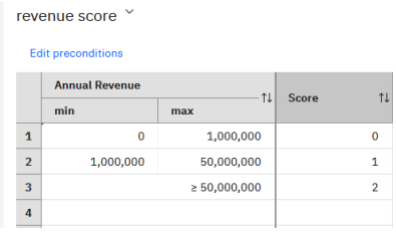
The logic contains one rule (defaulted score) and 3 decision tables (revenue score, size score, age score).

3. Click on **defaulted score** to explore the logic.

The defaulted score rule adds -1 to the decision in case of a defaulted payment (defaultedPayment).

4. Click on the **revenue score** decision table

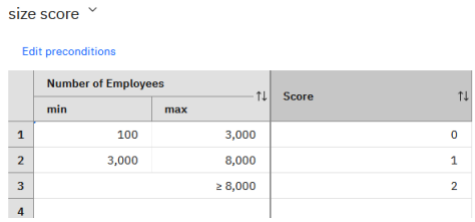
The revenue score decision table computes a score from the annual revenue.



	Annual Revenue		Score
	min	max	
1	0	1,000,000	0
2	1,000,000	50,000,000	1
3		≥ 50,000,000	2
4			

5. Click on the **size score** decision table.

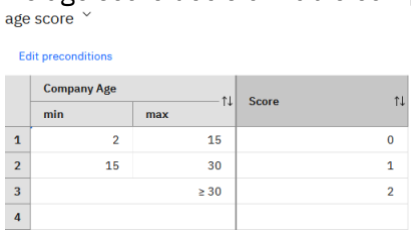
The size score decision table computes a score from the number of employees.



	Number of Employees		Score
	min	max	
1	100	3,000	0
2	3,000	8,000	1
3		≥ 8,000	2
4			

6. Click on the **age score** decision table.

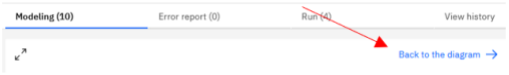
The age score decision table computes a score from the company age.



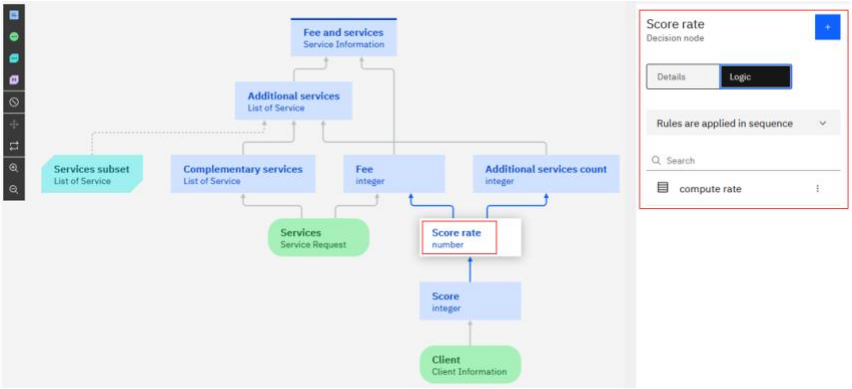
	Company Age		Score
	min	max	
1	2	15	0
2	15	30	1
3		≥ 30	2
4			

The score output is used by the score rate node to compute a rate from the score.

7. Click on **Back to the diagram** to return to the decision diagram

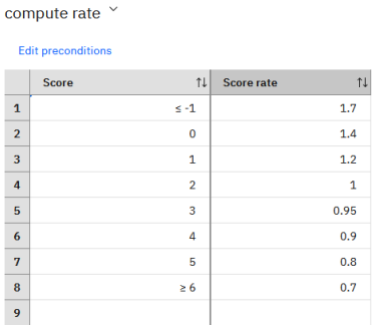


8. Click on the **Score rate** node, and then on the **Logic** tab.



9. Open the **compute rate** decision table

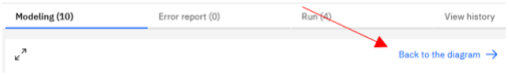
   You see the score values.



	Score	Score rate
1	≤ -1	1.7
2	0	1.4
3	1	1.2
4	2	1
5	3	0.95
6	4	0.9
7	5	0.8
8	≥ 6	0.7
9		

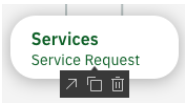
   In the next step, you change the decision logic.

10. Click on Back to the diagram

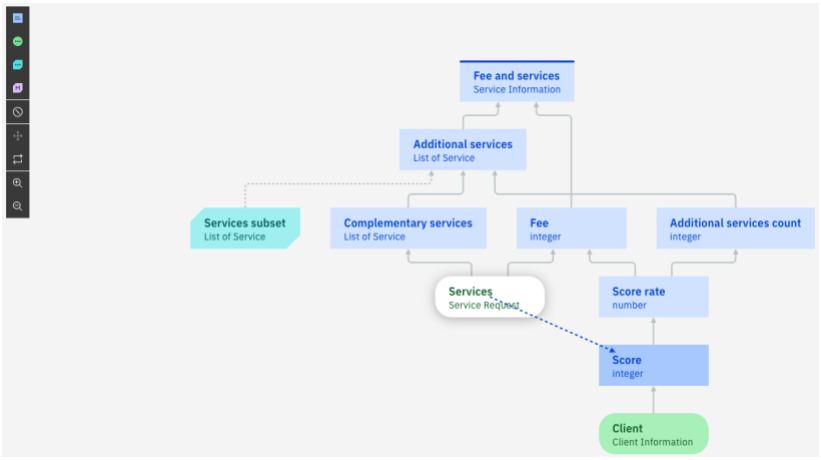


2.2.3.3 Adding a business rule

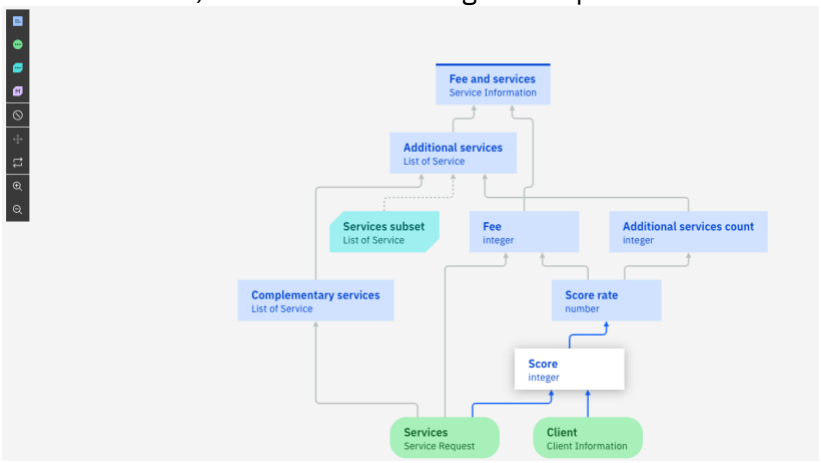
The score of the client is calculated based on 3 decision tables: revenue score, size score, and age score, and 1 rule: defaulted score. The score should also be evaluated based on the number of services the customer requested to be onboarded. You modify the business rule that calculates the score to add the number of services to its logic. A higher number of services results on a higher score.
To add the number of services as an input to the score computation, you update the decision node **Score** to add **Services** as input and add a new rule to determine the score.

1. Click on the **Services** input node, then hover over the icon 

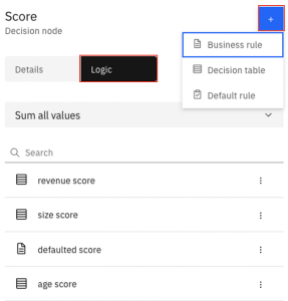
, and then click on **Connect to another node** (arrow) to add a link to connect to another node. Drag the link and drop it on the **Score** decision.



It creates a link, and the decision diagram is updated to reflect the new dependency:

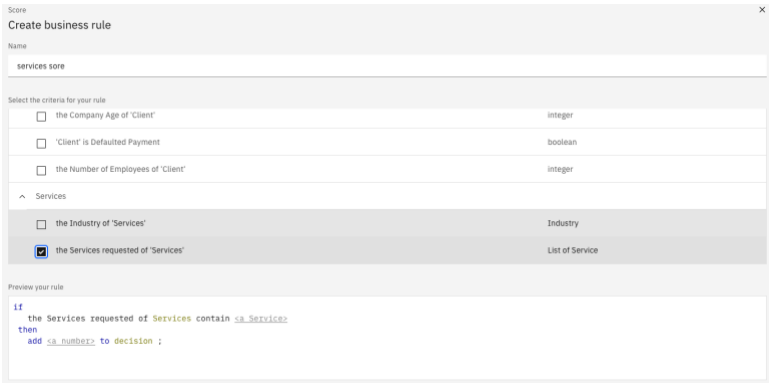


2. Click on the **Score** decision node, then select the **Logic** tab in the right pane, and then click on  > Business rule



It opens a wizard to help you select the criteria of your rule.

3. Enter **services score** as the name for the rule, and then scroll down to select the criteria **the Services requested of 'Services'**.



4. Click on **Create** and update your rule. You see a red icon that indicates an error. You can hover over the icon to see the details.

Important: Click the Refresh button in your browser if you see two red icons instead of just one:

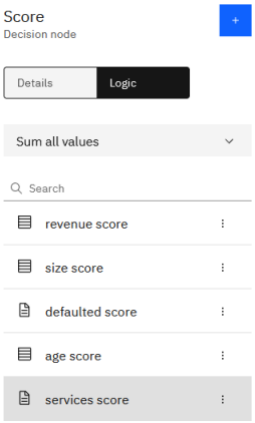
5. In the rule editor, update the rule. Replace 'the Services requested of Services contain <a Service>' by 'the number of elements in the Services requested is more than 3 then add 1 to the decision.

```
if
  the number of elements in the Services requested of Services is more than 3
then
  add 1 to decision ;
```

While you are typing, the auto-completion guides you through writing your rule. You can trigger the auto-completion by typing Ctrl-Space in the editor.

Once finished the rule appears as follows:

In the decision logic, the **Score** node is now the sum of 5 rules: revenue score, size score, defaulted score, age score and services score.



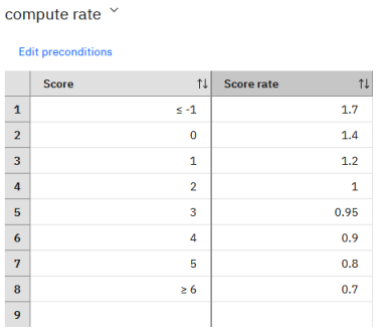
6. Return to the Decision model by clicking on Back to the diagram → above the rule.

2.2.3.4 Updating a decision table

You update the decision table for the score rate because the values change following the update of the rule to compute the score.

1. Click on the **Score rate** node. Then, in the **Logic** tab, click on the **compute rate** decision table.

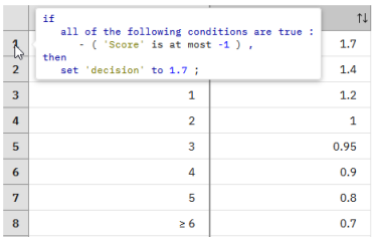
   It displays the decision table that contains the decisions to determine the score rate.



	Score	Score rate
1	≤ -1	1.7
2	0	1.4
3	1	1.2
4	2	1
5	3	0.95
6	4	0.9
7	5	0.8
8	≥ 6	0.7
9		

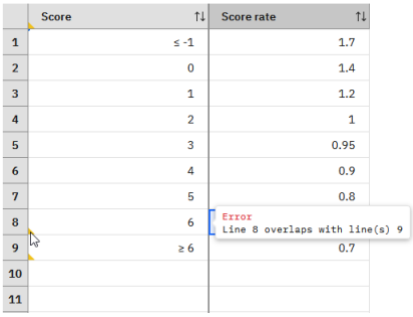
   In this table, each row represents a single decision. The Score represents a condition parameter and the Score rate the value for the action.

2. Hover over the **1** in the first row to display the text as shown below:



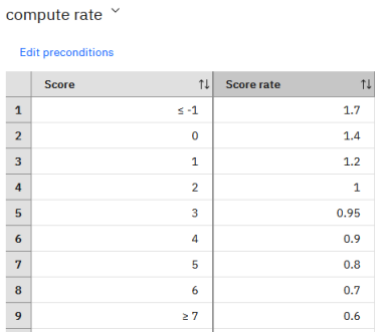
1	≤ -1	1.7
2	0	1.4
3	1	1.2
4	2	1
5	3	0.95
6	4	0.9
7	5	0.8
8	≥ 6	0.7

3. Select **row 8** and then right-click to open the contextual menu. Select **Insert row -> Above**. Enter 6 in Score and 0.7 in Score rate.



The warning icon appears in lines 8 and 9 to outline an error. The error indicates an overlap of row 8 and 9. You must change the values in row 9.

4. Change to 7 for **Score** and 0.6 for **Score rate**.

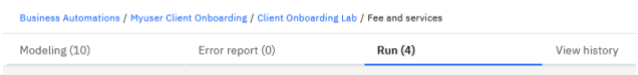


5. Click on **Back to the diagram** to return to your decision service.

2.2.4 Validating the decision service

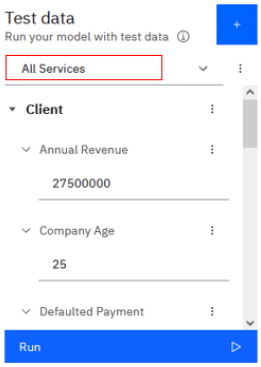
Now that you have modified the decision model and updated the decision logic, the decision service should be tested before deploying to production. Automation Decision Services integrates a test environment allowing you to fine tune your rules and verify their behavior.

1. Select the **Run** tab.

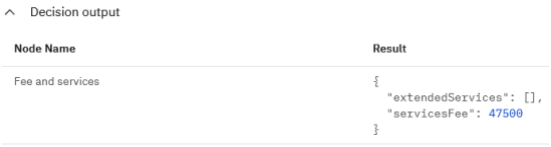


On the left side of the screen, the **Test data** pane allows you to select the data to submit to execute the rules.

2. To see the behavior after the changes, select the same data set as you did before the changes: **All Services** and click on **Run**.



The decision output for servicesFee is now 47500.

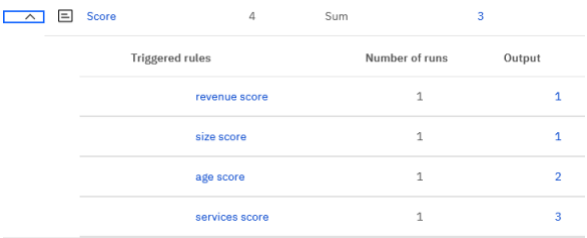


3. Expand **Run history**, and then expand Score.

   It shows that the rule you added **services score** has been triggered.

4. The output displays the result for the decision node **Score**, it is a consolidated result. It is the sum of the rules that triggered the result, applied in sequence.

   Each displayed output is the sum, including the previous output.



   In this example, the output of the **Score** decision is 3:
   - revenue score output is 1
   - then added to size score (size score is 0) it remains 1,
   - then added to age score result is 2 (age score is 1)
   - then added to services score it is 3 (services score is 1).

5. Click on **Client Onboarding Lab** in the breadcrumbs to navigate back to the Models tab.

For more information, see IBM Documentation
[Building decision models]
[Creating decision model]

2.3 Summary

You have completed the Exercise 1 - Modeling Decisions. You created a project and a decision service to define the fee and services for a client being onboarded to a set of services.

- You explored the decision model diagram and its elements.
- You updated the business logic.
- You created a rule.
- You added a link in the decision diagram.
- You edited a decision table.
- You tested and validated the decision service.

3 Exercise 2: Adding machine learning in the decision model

3.1 Introduction

In this exercise you leverage a machine learning model to improve the quality of the decision that this model can evaluate. This model can evaluate a risk based on information provided during the request. Adding a predictive model allows you to benefit from the previous customer's experience to estimate the risk level.
This model provides a prediction for the client risk based on the client information (annual revenue, company age, number of employees, and industry). If the client onboarding is risky, the prediction returns 1. If it is not, it returns a value of 0. It also provides the prediction probability.

3.2 Exercise instructions

In this exercise, to use the machine learning model deployment into Automation Decision Services you:

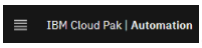
- Define a **provider** in the project to access the machine learning deployment
- Create a **predictive model** connected to the machine learning deployment
- Use this **predictive model** into a **decision service**
- Validate a decision service

3.2.1 Log in to your project

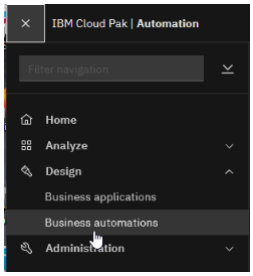
If you are already connected to Automation Decision Services and have your project open you can skip this section and pursue in next section Testing the model before changes.

If you begin the Lab start with this exercise, follow the steps below.

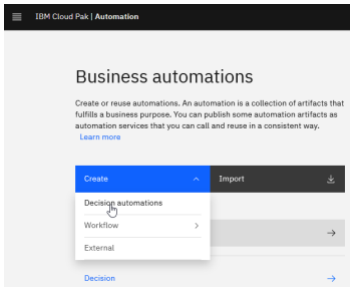
3.2.2 Creating a project and importing a Decision service

1. Log in to Business Automation Studio.
2. Click on the Navigation Menu on the top left corner 

.
3. Expand **Design**, and then click Business automations.

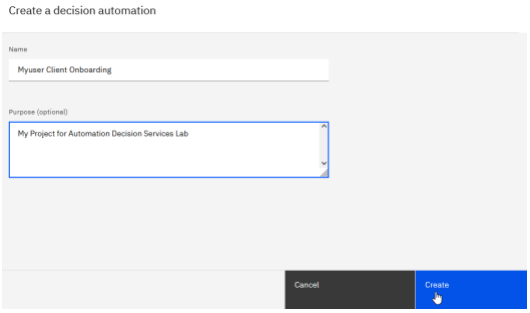


4. Click on Create and select Decision automations.

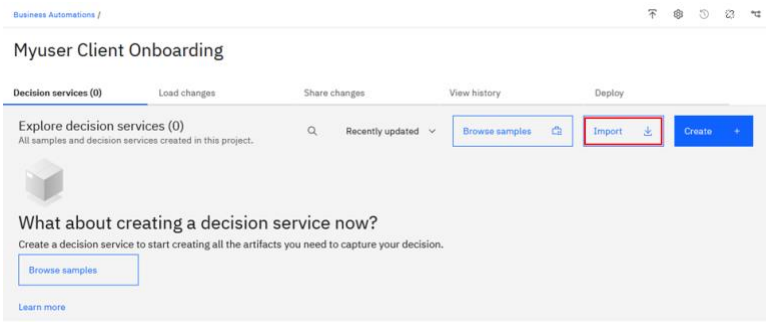


5. Enter a name for your project. Enter **UsrNNN Client Onboarding Decision** where *UserNNN* is your assigned username (a name starting by your username to avoid conflicts with other projects).

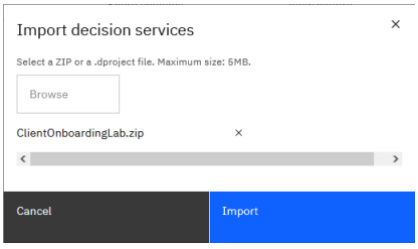
6. Click on **Create**



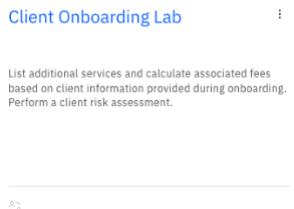
7. Your new project opens in Decision Designer. Click on **Import** to import the Decision Service provided by the Lab team.



8. Browse to select the project prepared for the Lab **ClientOnboardingLab.zip** and click on **Import**.



9. A tile of a decision service named **Client Onboarding Lab** appears on the project.



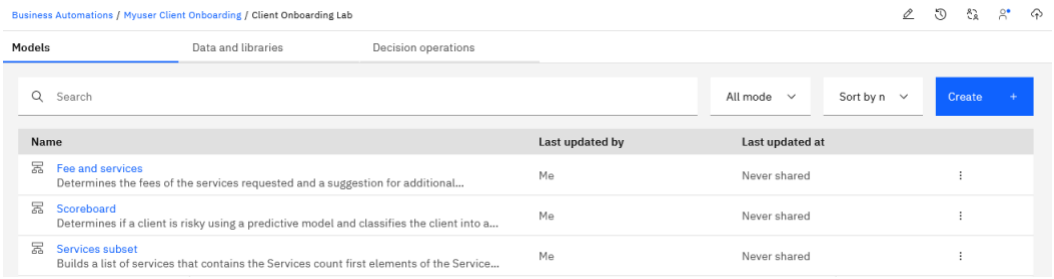
10. Click on **Client Onboarding Lab** to open your decision service.

3.2.3 Testing the model before changes

The Client Onboarding Lab decision service contains several decision models. In this exercise, you work on the Scoreboard decision model. It uses one data model: Data.

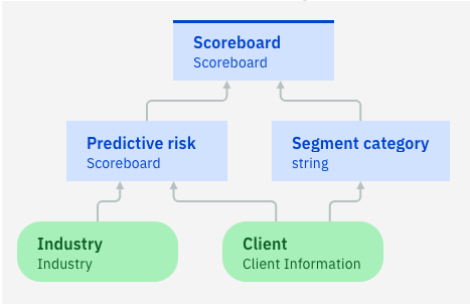
If you are not familiar with the data model, you can check Exercise 1 Exploring the decision service: data model and decision model.

1. In the Models tab, click on Scoreboard.

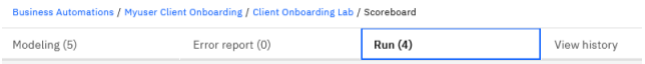


2. It opens the Decision model **Scoreboard**.

   The decision model diagram is a dependency diagram that shows the steps to make the final decision. The top decision **Scoreboard** is the final decision. It calculates a client risk and categorizes the client in a business segment. The segment is computed by the decision node **Segment category** from the input node **Client**. The risk is computed from the **Predictive risk** node with the input nodes **Industry** and **Client**. This model is not complete because the Predictive risk node has no business logic defined. You will complete it in this exercise.

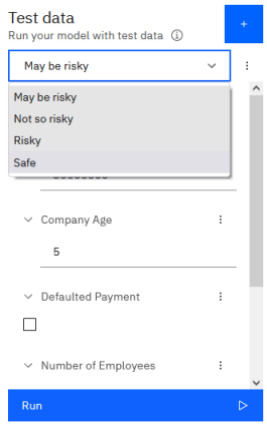


3. Select the **Run** tab to test the model



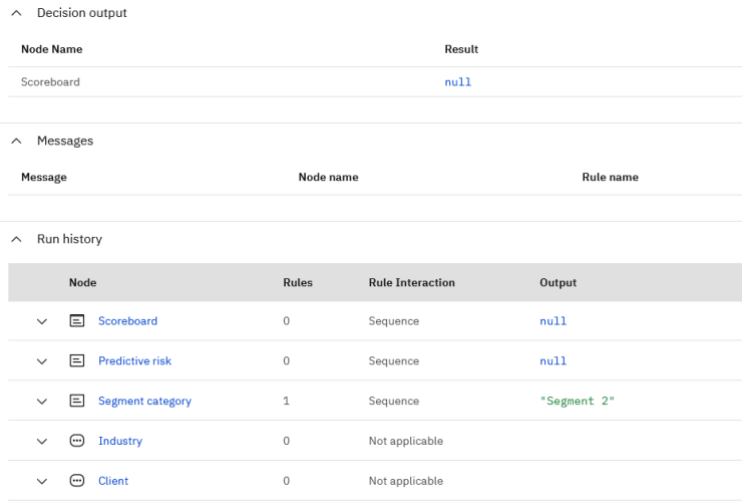
   On the left side of the screen, the **Test data** pane allows you to select which set of data to submit to execute the rules. In this Lab, 4 sets of data are pre-defined: May be risky, Not so risky, Risky, Safe.

4. Select the **May be risky** data set and click on **Run** to test the model.



A report displays the input and output of each node in the decision model.

5. Expand Decision output and Run history.



The final decision is displayed in the top of the report: for Scoreboard the result is "null".
The decision result provides the segment 2 category based on the Annual revenue specified in the data set. If the revenue is below 50000000 it categorizes the client in segment 2. If the revenue is greater, the client is set in segment 1. Here the risk is not yet computed. You need to add the logic of the decision node Predictive risk.

3.2.4 Defining a Machine learning provider

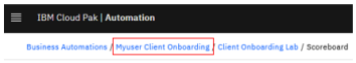
A machine learning provider gives access to Machine learning deployment. Two types of providers are currently supported in Automation Decision Services: IBM Watson® Machine Learning and IBM Open Prediction Service.

In this exercise, the Machine learning model has been deployed on an instance of IBM Open Prediction Services (OPS).
This machine learning model has been defined using a Jupyter notebook and saved into a Predictive Model Markup Language file (PMML). This PMML file has been used to deploy the machine learning model into IBM Open Prediction Services (OPS).

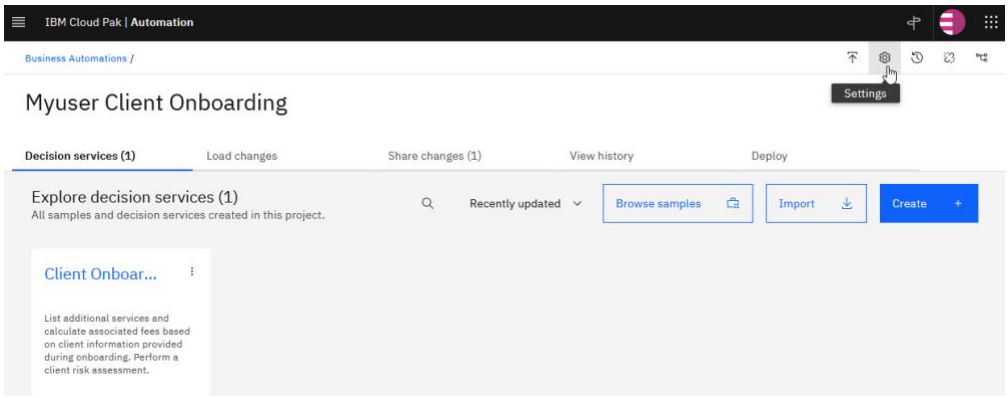
For more information, see IBM Documentation Managing machine learning providers

For more information about ADS ML Service, see the Open Prediction Service Hub repository [available on GitHub](#).

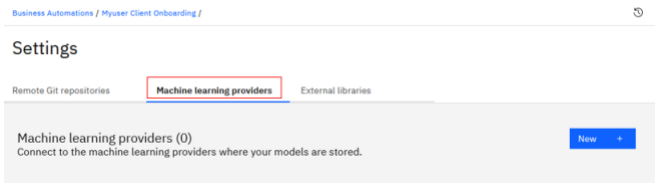
1. Click on your project Myuser Client Onboarding in the breadcrumbs to navigate back to your project.



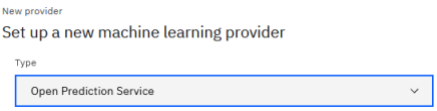
2. Click on **Settings** in the top-right corner.



3. Open the Machine learning providers tab and click **New**.

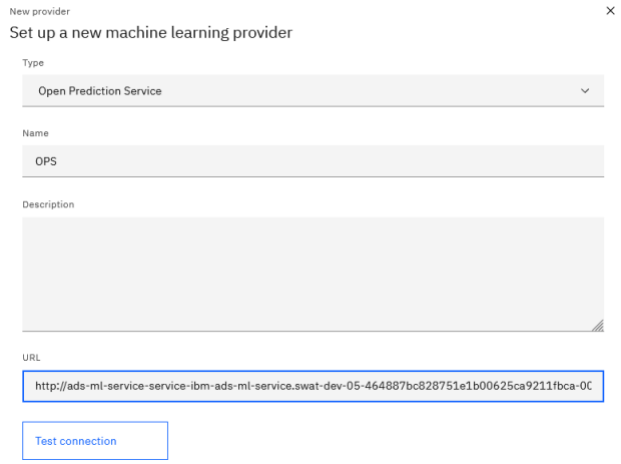


4. Select the provider type **Open Prediction Service**

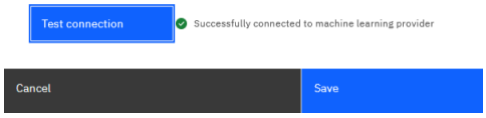


5. Enter **OPS** in the **name** field, and optionally a description
6. Enter the URL of the Open Prediction Service Instance. The URL you need is the URL of ADS ML Service (see [Prerequisites](#) if you need the URL)

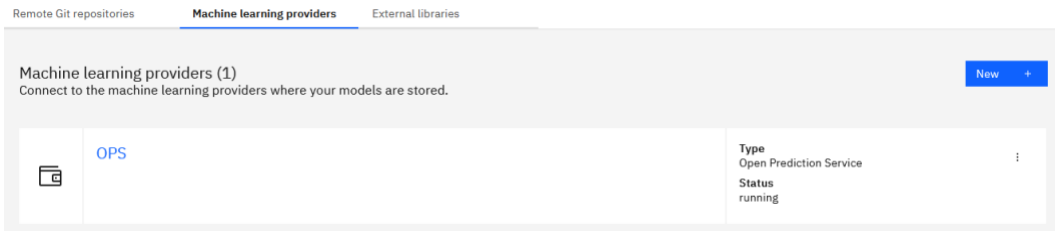
   Make sure that you remove ***/docs*** from the end of the url. For example: http://ads-ml-service-service-ads-ml-service.cp4ba-jam-americas-464887bc828751e1b00625ca9211fbca-0000.us-south.containers.appdomain.cloud



7. Click on the **Test connection** button to verify your OPS provider access. Upon successful connection click on **Save**

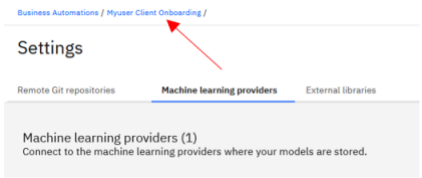


8. In the **Machine learning providers** tab, you have now the OPS provider you defined.



For more information, see IBM Documentation
Integrating machine learning
Managing machine learning providers

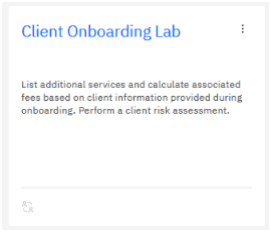
9. Go back to your decision service by clicking on your project name in the breadcrumbs.



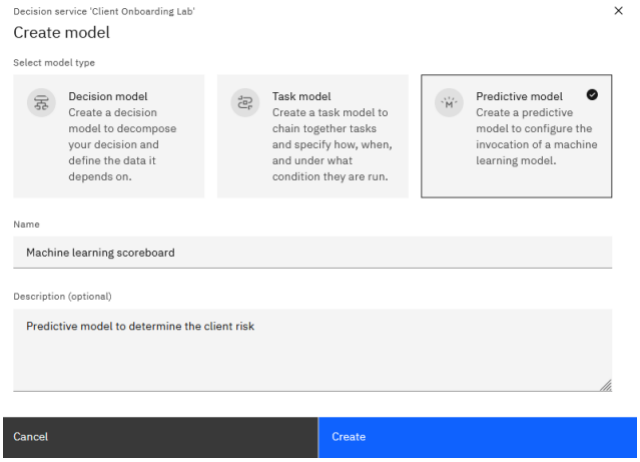
3.2.5 Creating and connecting the Predictive Model

Now that the provider is defined, you can create a predictive model to encapsulate the machine learning model deployment. You connect it to the appropriate machine learning model deployment and define the input and the output of this predictive model.

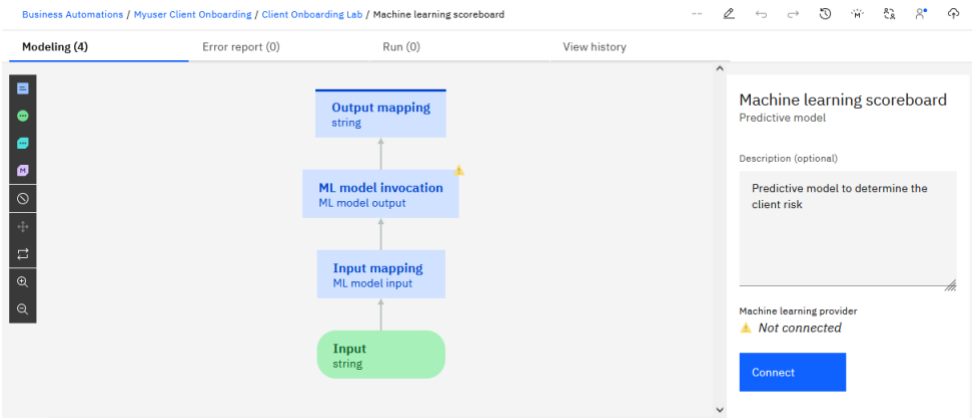
1. Click on the tile **Client Onboarding Lab** to open the decision service



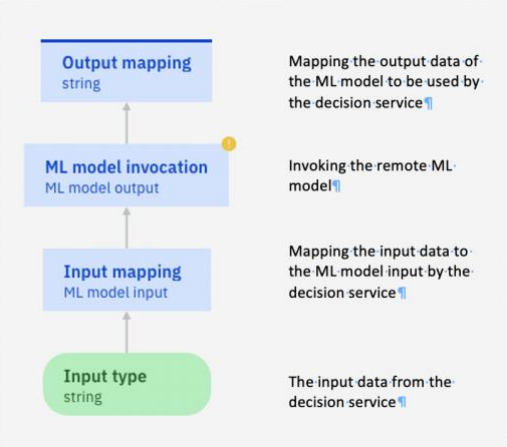
2. On the **Models** tab, click on **Create** and select **Predictive model**.
3. Enter a name for your model (Machine learning scoreboard), a description (optional), and click on **Create**.



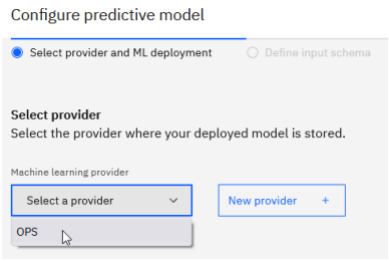
4. The Predictive model **Scoreboard** is opened.



A predictive model is a simple Decision Model diagram. It has one or multiple input nodes, two mapping nodes, and one node that invokes the Machine Learning model.



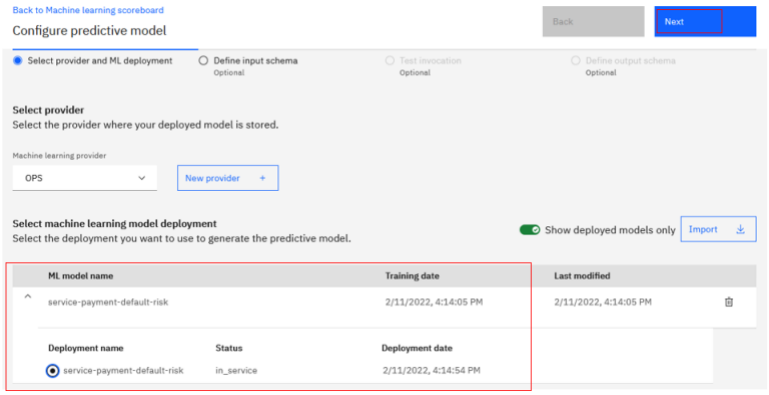
5. In the right-hand pane click on **Connect**.
6. Expand **Select a provider**, and select OPS.



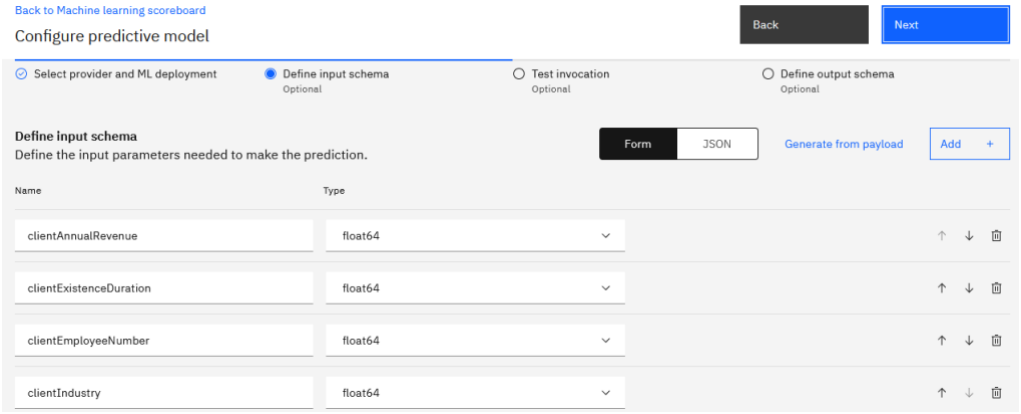
7. In **ML model name** list, expand service-payment-default-risk.

   The list displays all the machine learning models deployed in the environment. Expand the model with the most recent Last modified date.

8. Choose service-payment-default-risk, click on **Next**.

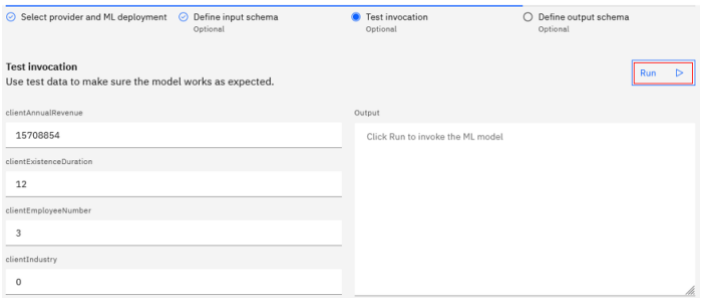


9. Look at the **input schema**: you must provide the values to the machine learning model from the input nodes defined in your predictive model. To do so, click on **Next**.

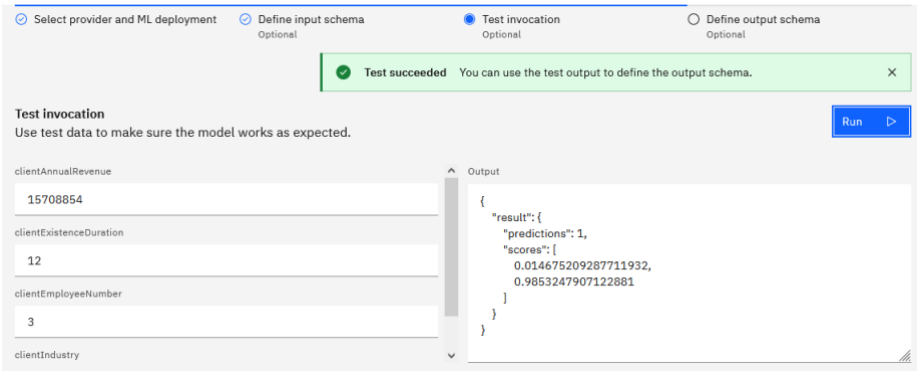


10. Enter some values to test in the **Test invocation**.

clientAnnualRevenue: 15708854
clientExistenceExistenceDuration: 12
clientEmployeeNumber: 3
clientIndustry: 0



11. Click on **Run**

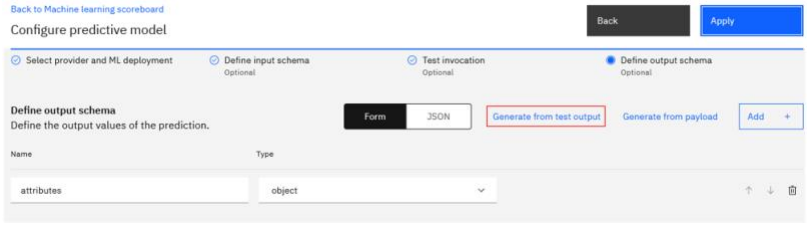


The output returns: it is a risky client as the prediction is 1

```
{
   "result": {
      "predictions": 1,
      "scores": [
         0.014675209287711932,
         0.9853247907122881
      ]
   }
}
```

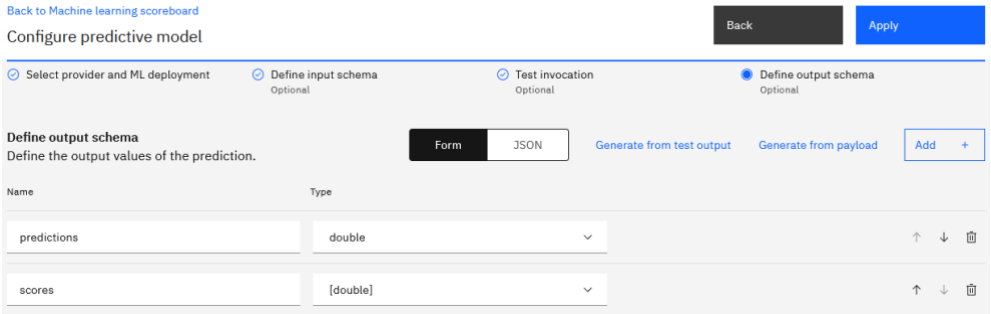
12. Click on **Next**. Look at the **output schema**: you will now map it to the scoreboard type in your predictive model.

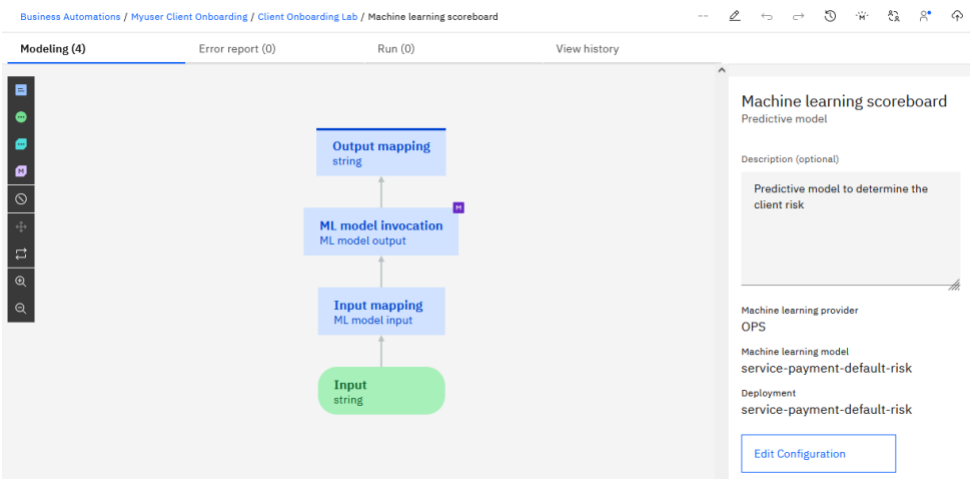
13. Select Generate from test output.



14. Click **OK**

15. Finally click on **Apply** to finish the configuration of your predictive model.





3.2.6 Defining the mapping

You map the data types of the input and output of the machine learning model to the data types of the input and output of the Decision Service data model.

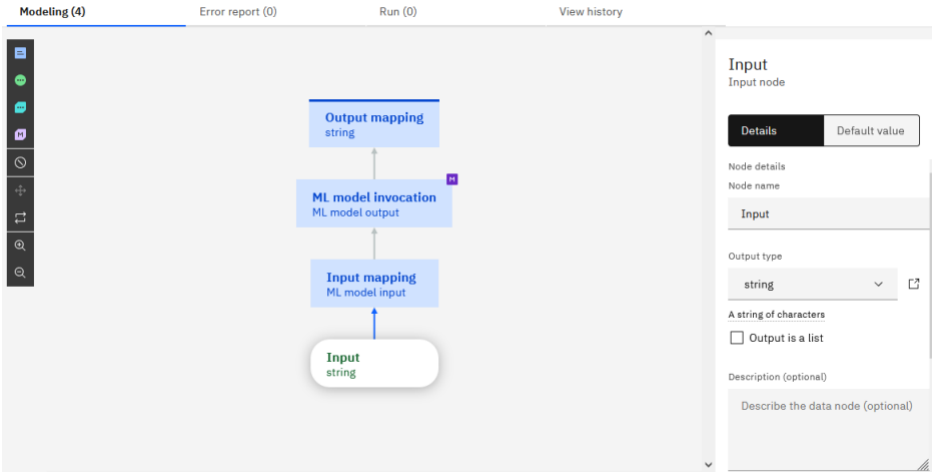
3.2.6.1 Defining the input mapping

In the previous section you saw the input mapping of the ML model, which requires:
- clientAnnualRevenue as a number
- clientExistenceDuration as a number
- clientEmployeeNumber as a number
- clientIndustry as a number

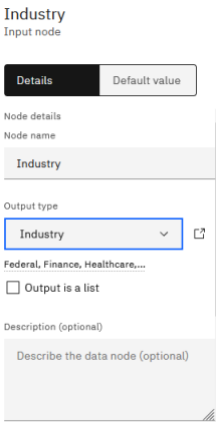
You get the three first values form the Client information type. The last one is taken from Industry enumeration type. You map each value to a number. You must now:
- Define an input node for **Industry**
- Add an input node for **Client Information**
- Define the rules to map these input nodes to the values expected by the ML model. For Industry, you need a decision table to map the enumeration type to an integer.

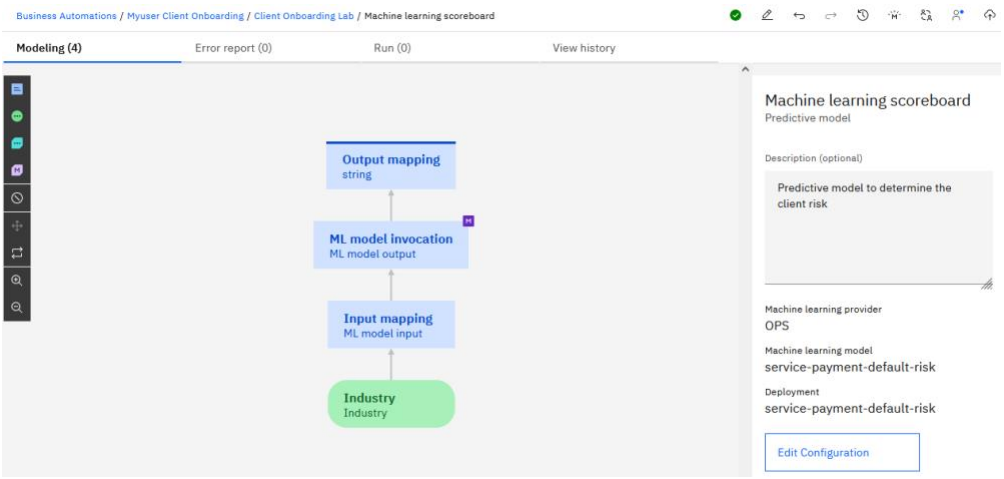
The predictive model Machine learning model is based on the industry and the client information as input. You update the model to map your data to the required input:

1. Select the **Input** node in the diagram.



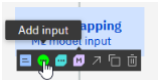
2. In the right pane **Details,** enter Industry as the **Node name** and expand **Output type** to select Industry as the type.



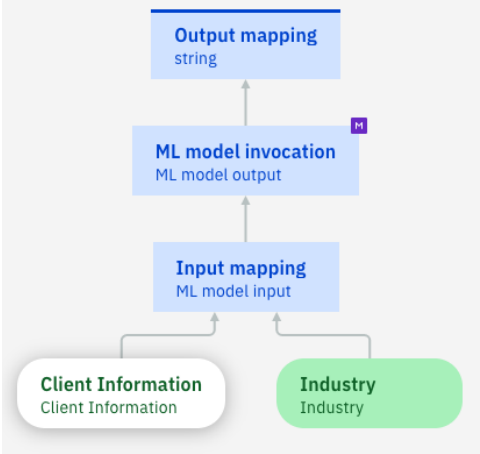


You add a node in the model for Client information.

3. Hover over the **Input mapping** node in the diagram and click on the **Add input** icon.

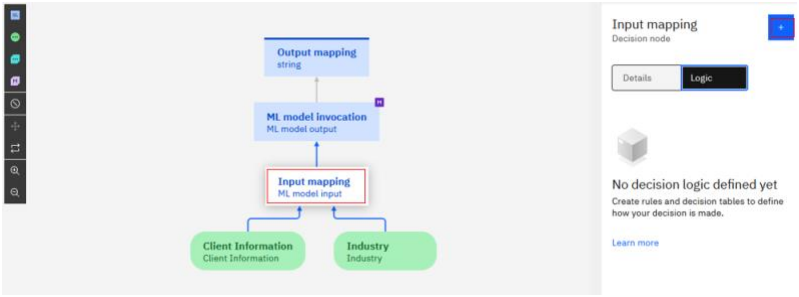


4. Select the new **Input 1** node and update the Node name to **Client Information** and Output type to **Client Information**.



You have added the Input nodes. Now, you map the input with the Decision Service data model.

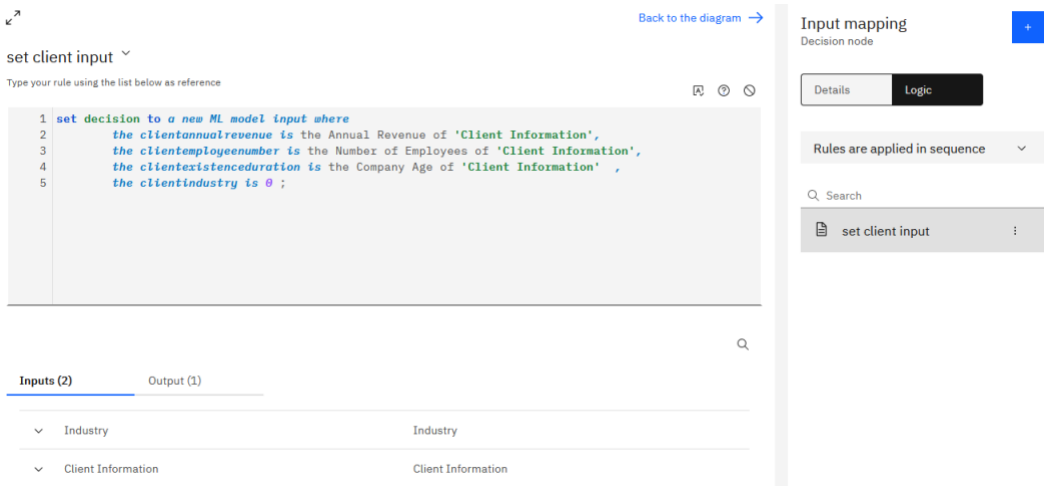
5. In the Predictive model diagram, select the **Input mapping** node. Then, click on the **Logic** tab, select the + sign, and add a **Business rule**.



6. In the wizard, notice the **preview** of the generated rule. It matches the input of the Machine Learning Service (Industry, Client Information). Enter the rule name **set client input** and click on **Create**.

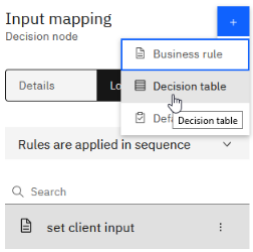
7. Edit the rule with the following statement:

```
set decision to a new ML model input where
       the clientannualrevenue is the Annual Revenue of 'Client Information',
       the clientemployeenumber is the Number of Employees of 'Client Information',
       the clientexistenceduration is the Company Age of 'Client Information'  ,
       the clientindustry is 0 ;
```

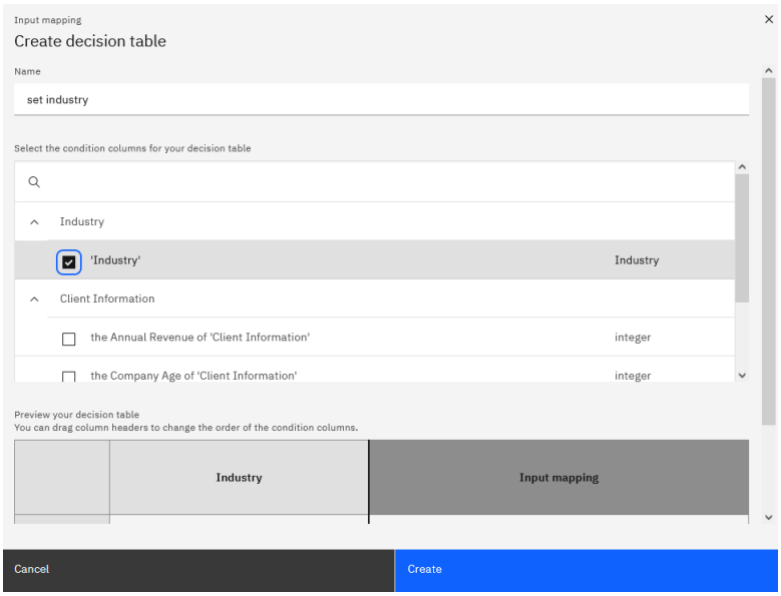


This rule always sets the client industry to 0. Now, you add a new decision table to update the business logic by defining the industry from the industry provided as the input.

8. In the **Logic** tab, select the **+** sign and select **Decision table**.

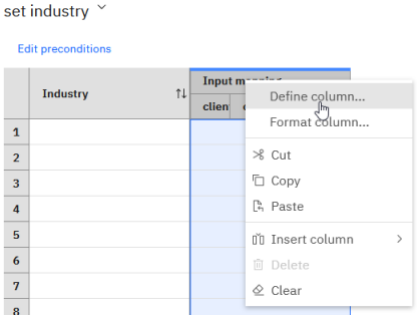


9. Enter **set industry** for the Name. Select Industry in the **condition columns** list. Click on **Create**.



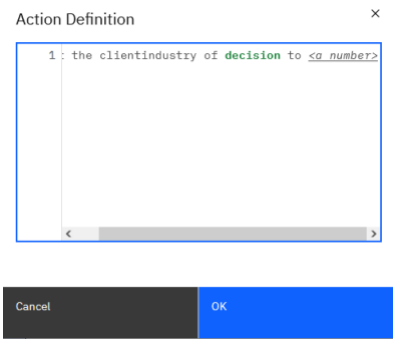
10. Right-click on the **Input mapping** column and select **Define column**.

    You need to update the column to set the industry only. The other attributes were set in an earlier step by the rule set client input.

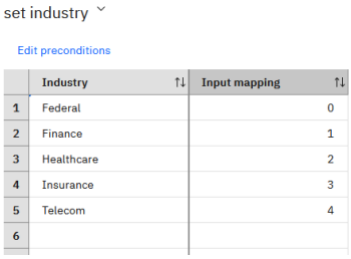


11. Update the rule in the editor with the following rule statement. Then click **OK** in the Action Definition dialog.

```
set the clientindustry of decision to <a number>
```



12. Double click in **Row 1** and select **Federal** in the drop-down list for the Industry. Add **0** for Input mapping.
13. Pursue for rows 2 to 5. Enter the following values:



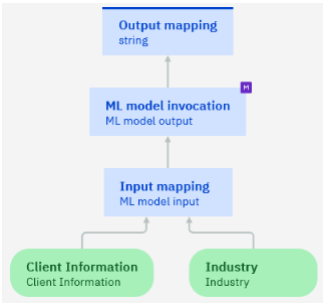
	Industry	Input mapping
1	Federal	0
2	Finance	1
3	Healthcare	2
4	Insurance	3
5	Telecom	4
6		

3.2.6.2 Defining the output mapping

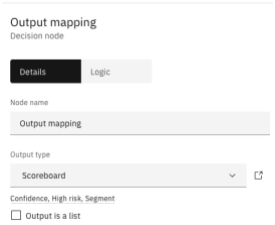
In the previous section you saw the output of the machine learning model. You must map it to the output of your predictive model. Your predictive model will return a scoreboard in which it predicts the score and gives the probability. To do this you:

- Change the output type of the Output mapping node of the predictive model to scoreboard.
- Define the rule to build this scoreboard:
  - Write a default rule for the case of the machine learning model in error. This rule must be the only one to be executed in case of an error. You use the rule policy First rule applies and ensure that this rule is on top.
  - Write a rule to map the output of the machine learning model to the scoreboard. The segment remains Unknown since it is not computed by the machine learning model.

1. Click on Back to the diagram → to return to the Predictive model diagram.

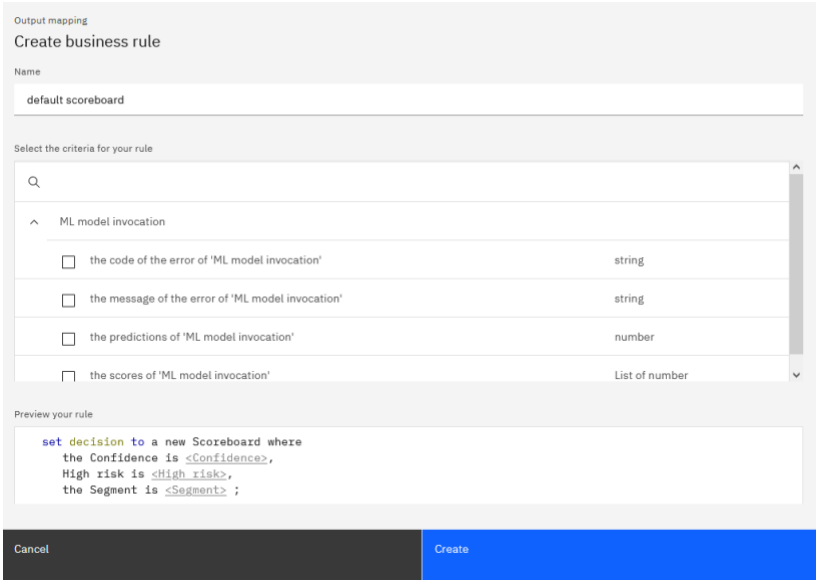


2. In the Predictive model diagram, select the **Output mapping** node. Then, in the **Details** tab, expand **Output type** and select **Scoreboard** in the list.

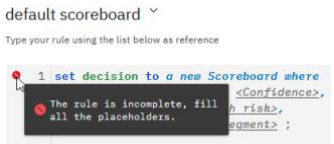


You define the business logic by adding two rules to feed the output type from the machine learning model invocation. First, a rule to define an output when the machine learning model is in error (**default scoreboard**). Second, a rule to map the machine learning model output (**get machine learning output**). The two rules should not be applied in sequence. Only the first rule applies if there is an error in the machine learning model.

3. Select the **Logic** tab, then select the + sign and add a **Business rule**.
4. Enter a name for the rule: **default scoreboard**. Click on **Create** to edit your rule.

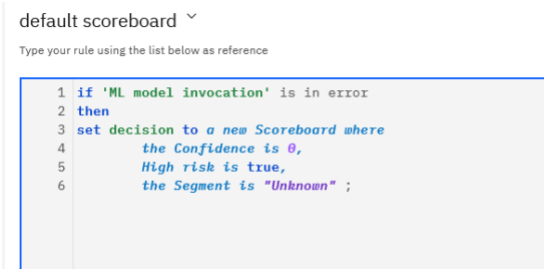


5. In the editor, a rule is predefined. An icon indicates an error because the rule is incomplete. Hover over the error icon in line 1 to see the details:

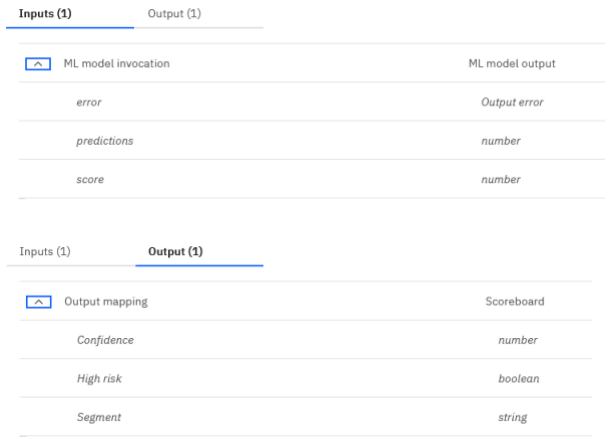


6. Edit the **rule** statement as follows

```
if 'ML model invocation' is in error
then
set decision to a new Scoreboard where
       the Confidence is 0,
       High risk is true,
        the Segment is "Unknown" ;
```

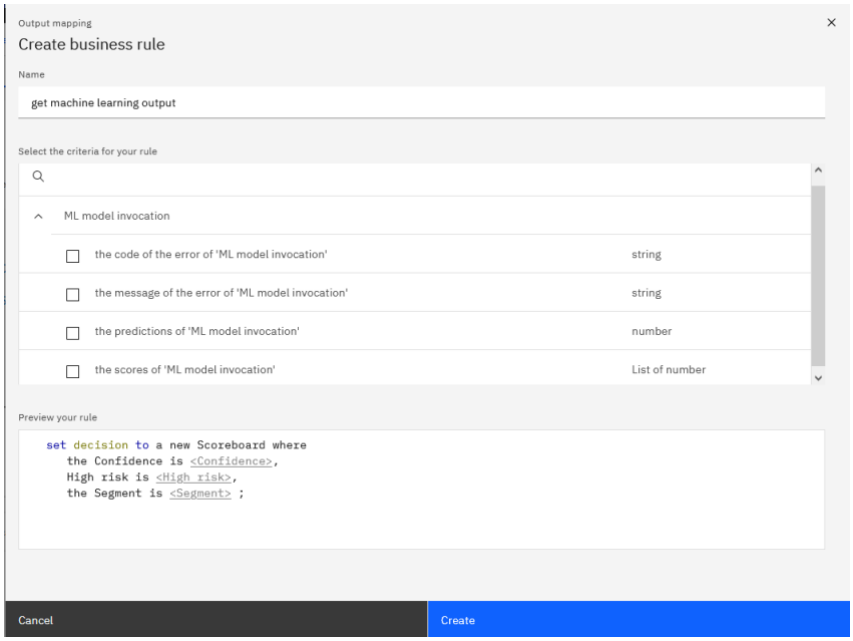


7. Click on **Inputs** and **Output** below the rule to see the data types of the input and output.



Now you add the second rule.

8. Click on the **Logic** tab. Then select the **+** sign and add a **Business rule**.
9. Enter the name **get machine learning output**. Click on **Create** to edit your rule.



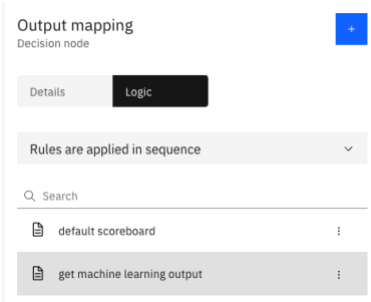
The rule is incomplete as you can see from the icon in line 1.

10. Edit the rule with the following statement:

```
definitions
set percent to the maximum score of 'ML model invocation';
if the predictions of 'ML model invocation' is 1
then
set decision to a new Scoreboard where
    High risk is true ,
```

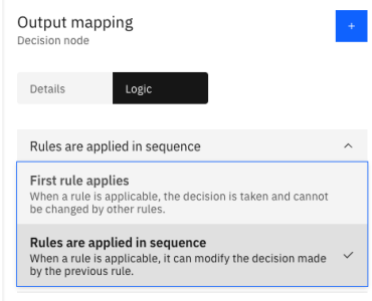
```
  the Confidence is percent,
    the Segment is "Unknown";
else
set decision to a new Scoreboard where
    High risk is false ,
  the Confidence is percent,
    the Segment is "Unknown";
```

11. In the **Logic** tab in the right pane you can see **Rules are applied in sequence**, first default scoreboard, then get machine learning output.

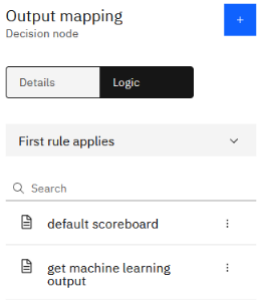


How the rules interact with each other is defined by a **rule interaction policy**.

12. Expand **Rules are applied in sequence**. You see two policies **First rule applies** and **Rules are applied in sequence**. The choices for the rule interaction policy depend on the Decision type. Here, the decision node **Output mapping** is a custom type.



13. Change the rule policy to **First rule applies**.

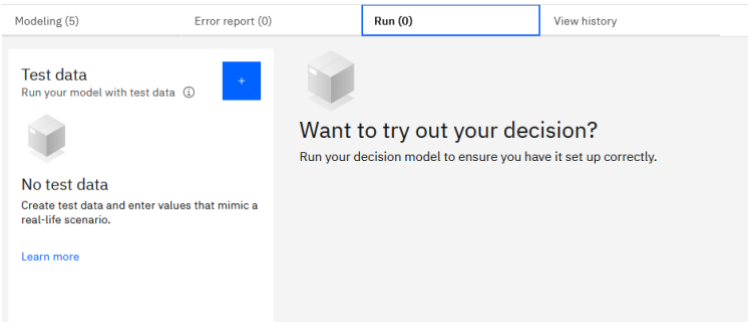


For more information, see IBM Documentation Choosing an interaction policy

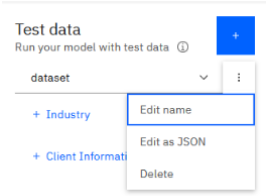
3.2.7 Validating the predictive model

Now that you have created your Predictive model and defined the predictive decision logic, you test it to verify. IBM Automation Decision Services integrates a test environment allowing you to fine-tune your rules and verify their behavior online against the test data.
You add datasets to define the data that you want to test.

1. Select the **Run** tab.



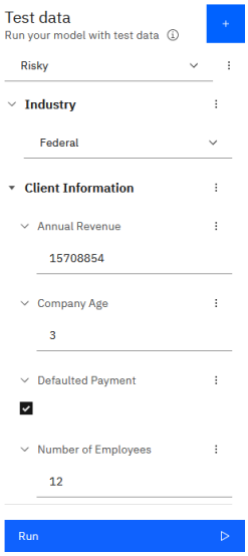
2. In the **Test data** pane, click on **+** sign. Then, select the 3 dots to edit and add test data.



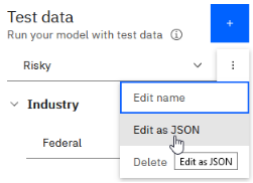
Now you add four data sets: Risky, May be risky, Not so Risky, Safe.
To add test data, you enter a name and define the values.

3. Select Edit name, enter **Risky**. Expand **Industry** and **Client Information** to enter the test data

Name: **Risky**
Industry: Federal
Annual Revenue: 15708854
Company Age: 3
Defaulted Payment: true (checked)
Number of Employees: 12



4. You can see the JSON content by clicking on the 3 dots near the name and selecting Edit as JSON.




```
{
 "industry": "Federal",
 "clientInformation": {
  "annualRevenue": 15708854,
  "companyAge": 3,
  "defaultedPayment": true,
  "numberOfEmployees": 12,
 }
}
```


You can edit datasets as FORM by entering the data in the form as you did in this step or as JSON. To enter a new data set you can paste a JSON content.

5. (Optional) Repeat the steps to add 3 additional datasets. Copy/Paste the JSON content

Name: **Not so risky**

```
{
 "clientInformation": {
  "numberOfEmployees": 10,
  "annualRevenue": 61399457,
  "companyAge": 4
 },
 "industry": "Healthcare"
}
```

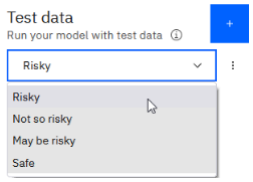
Name: **May be risky**

```
 {
  "industry": "Federal",
  "clientInformation": {
   "annualRevenue": 15708854,
   "companyAge": 3,
   "numberOfEmployees": 12
  }
 }
```

Name: **Safe**

```
 {
  "industry": "Healthcare",
  "clientInformation": {
   "annualRevenue": 75314927,
   "numberOfEmployees": 67,
   "companyAge": 26
  }
 }
```

You now have 4 data sets (or less depending on the number of data sets you added previously).



6. Select a data set of your choice and click on **Run** in the left pane.
7. See the Decision output for the data set Risky.

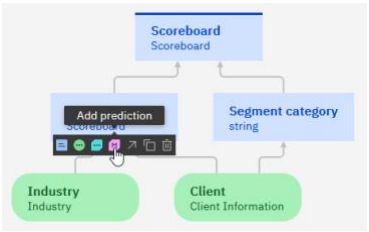
8. You can explore the output in JSON format, by clicking on **Show JSON output**

3.2.8 Use the predictive model in the scoreboard decision model

You add a prediction node to the scoreboard decision model connected to this predictive model. You connect this prediction node to the predictive risk input node and create a rule to define the risk.

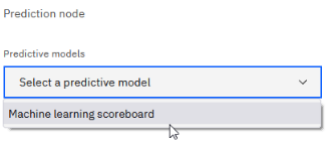
3.2.8.1 Add a prediction node

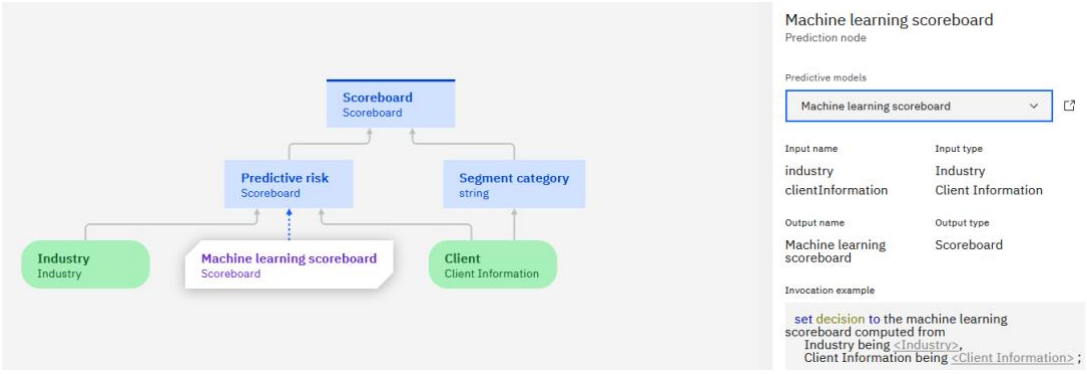
1. Go back to your **Scoreboard** decision model by clicking Client Onboarding Lab in the breadcrumbs and then on Scoreboard.
2. It shows the decision model Scoreboard. Hover over the **Predictive risk** node and click on **Add prediction** icon.



The prediction node shows an error because it needs to be connected to a predictive model.

3. Click on your new **Prediction node**, and in the right-side pane select the predictive model you created. Expand the list of Predictive models and select **Machine Learning scoreboard**.





3.2.8.2 Editing the logic definition of the predictive risk

You update the decision logic by adding a predictive rule. You add a rule calling the predictive model with the appropriate input, to define the risk and the confidence of the Scoreboard decision node.

1. Select the decision node **Predictive risk**. In the **Logic** tab, click the **+** sign and create a Business rule.
2. Enter a name for the rule**: predictive risk**.
3. Click on **Create**.
4. Edit the rule statement as follows:

```
set decision to the machine learning scoreboard computed from
     Client Information being Client ,
     Industry being Industry ;
```

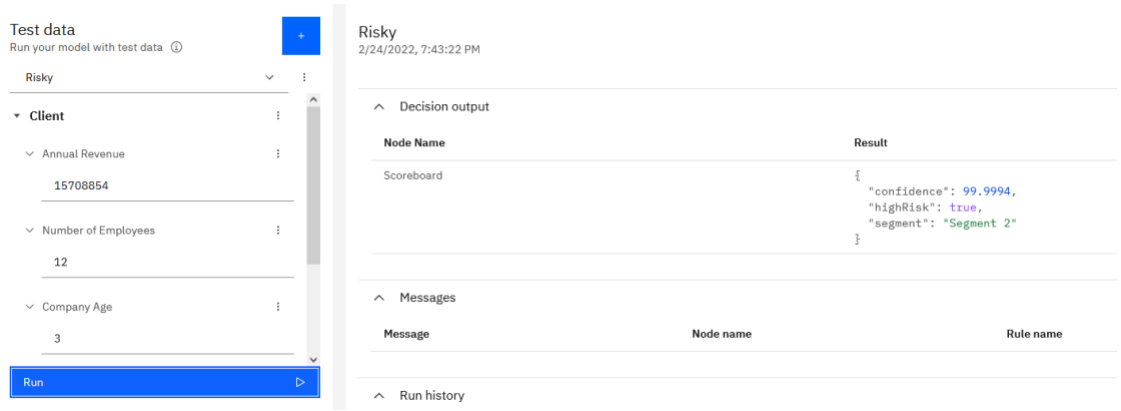
3.2.9 Validating the final decision service

Now that you created the predictive model and updated the decision model Scoreboard to integrate the prediction to compute the risk for the client onboarding, you can validate the changes. You use the four datasets previously created to validate:
Risky, May be risky, Not so Risky, Safe

1. Select the **Run** tab.

2. In the **Test data** tab, select the Risky dataset.

3. Click on **Run** and check the results.



The run history shows that the rule **Predictive risk** ran and computed the **confidence** level 99,99% and **high risk** value (true) for the **Scoreboard** decision node.

4. Click on **Myuser Client Onboarding** in the breadcrumbs to move on to the next exercise.

3.3 Summary

You have completed the Exercise 2 - Adding machine learning in the decision model.

- You added a predictive service provider and connected your project to it.
- You modified the scoreboard for the client onboarding by combining descriptive rule and predictive rule. By adding a predictive model to your project decision, you infused machine learning to evaluate the client risk based on a trained model.
- You added data sets to verify that your model is running correctly against the business rule policies defined for the Client Onboarding scenario.

4 Exercise 3: Sharing and publishing decision services

4.1 Introduction

In this exercise, you learn how to collaborate on your decision services and how to make it ready to be executed by the other components of the platform. You connect your project to a Git repository to be able to build and deploy a decision service as an archive. You can directly collaborate in Automation Decision Services by sharing your decision service and giving appropriate access. You publish the decision service archive as an automated service in Business Automation Studio.
This exercise is dedicated to integrators and anyone who wants to understand how to execute a decision service.

Integrating the automated services in other applications is not covered in this Lab. Look at the Workflow and Business Automation Application Labs for this.

For this exercise, you can either work with the project you created or with the decision services prepared by the Lab team.

4.2 Exercise instructions

In this exercise you prepare a decision service for collaboration:

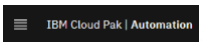
- **Connect** a decision service to a GitHub repository
- **Share changes** in a decision service to make them visible to collaborators
- **Deploy** the decision service as **an automated service**
- **Execute** the decision service

4.2.1 Log in to your project

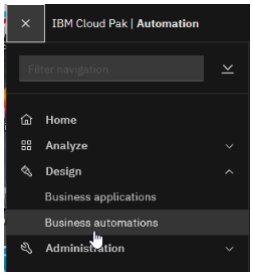
If you did the previous exercises and have your decision service you can pursue with your project. In Decision designer, open your project. Then move to the step Exploring a decision operation.

If you are starting the Lab with this exercise, follow the steps below.

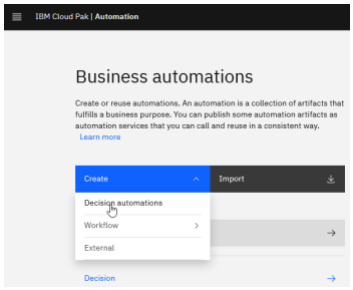
4.2.2 Creating a project and importing a Decision service

1. Log in to Business Automation Studio.
2. Click on the Navigation Menu on the top left corner 

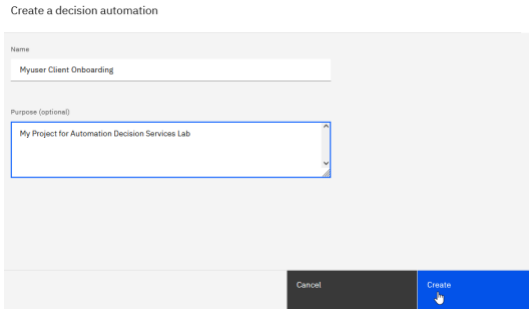
.
3. Expand Design, and click on Business automations.



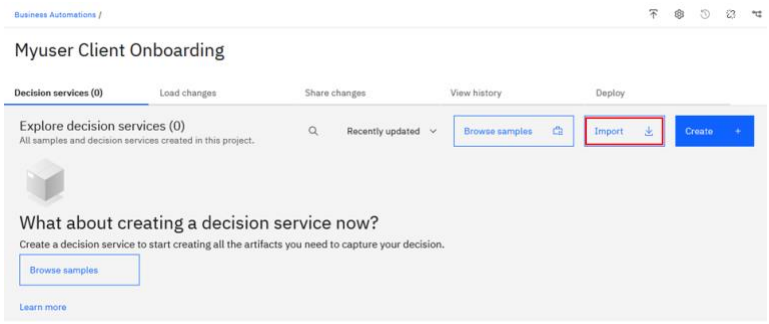
4. Click on Create and select Decision automations.



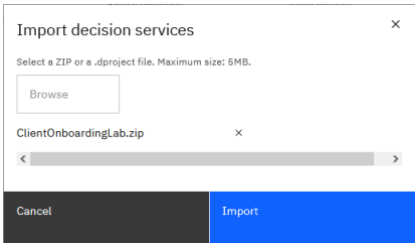
5. Enter a name for your project. Enter **UsrNNN Client Onboarding Decision** where *UserNNN* is your assigned username (a name starting by your username to avoid conflicts with other projects).
6. Click on **Create**.



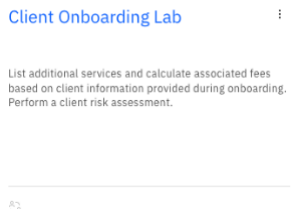
7. Your new project opens in Decision Designer. Click on **Import** to import the Decision Service provided by the Lab team.



8. Browse to select the project prepared for the Lab **ClientOnboardingLab.zip** and click on **Import**.



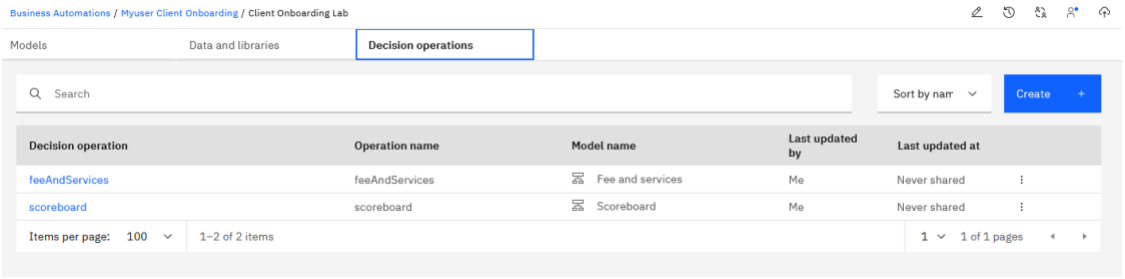
9. A tile of a decision Service named **Client Onboarding Lab** appears on the project.



4.2.3 Exploring a decision operation

To deploy your decision service, you need to define an operation that is used to call the service. The operation includes the name and a reference to a decision model. A decision service must contain at least one decision operation to be deployed and executed. The decision service prepared for the Lab already contains two decision operations **feeAndServices** and **scoreboard**.

1. Click on the tile of the decision service Client Onboarding Lab.
2. Open the **Decision operations** tab to explore the Decision operations:



3. Click **Myuser Client Onboarding** in the breadcrumbs to return to the project.

For more information, see the IBM Documentation Creating decision operations

4.2.4 Creating a Git repository

As a prerequisite you must have a GitHub account to do this step.

1. Open GitHub in your browser, and sign-in with your GitHub credentials.
2. Click the + button at the upper left part of the page and select **New repository** to create an empty repository.
3. Give the repository a unique name, and add the following description:

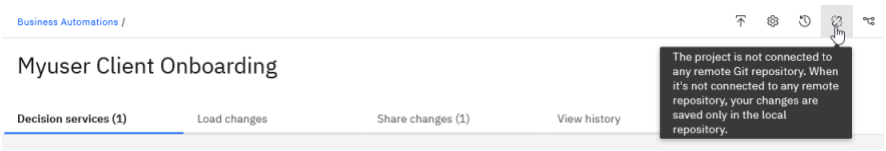
   Git repository for the Automation Decision Services Lab

4. Select Private and Click on **Create repository**. (The repository must **not** contain a readme, .gitignore, or license file.)
5. Click on the Copy button to copy the HTTPS URI and paste it in a safe location for use in the next step. The URI has the following format: https://github.com/<yourAccountName>/<yourRepoName>.git
6. Open the drop-down list for your GitHub account in the upper right corner of the page.
7. Click Settings and then Developer settings > Personal access tokens > Generate new token.
8. Put **ads synchronization** as the Note, and select the role **repo** to give full control of the repository to Automation Decision Services.
9. Click **Generate token** at the bottom of the page. Copy the generated access token before closing this page. You use it in the next step.

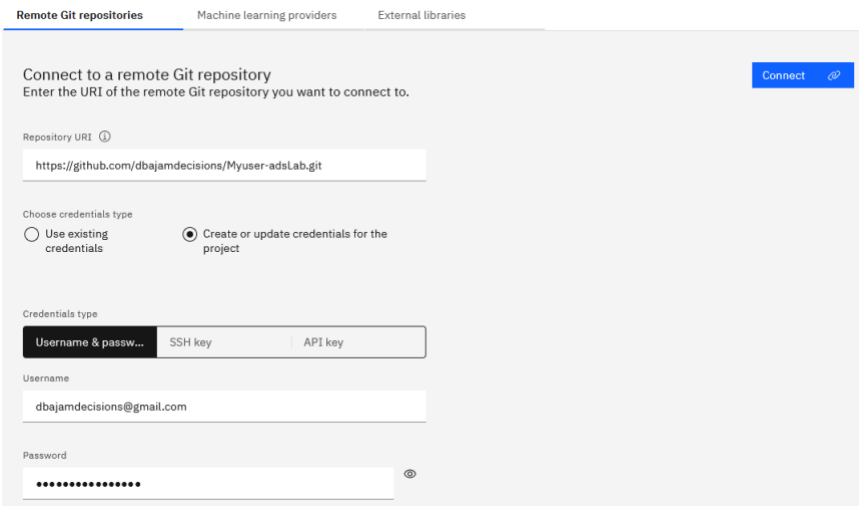
4.2.5 Connecting your project to the Git repository

You can manage your project on a Git repository where you will get the history of all changes from Automation Decision Services. Connecting to a Git repository is required to be able to build and deploy archives.

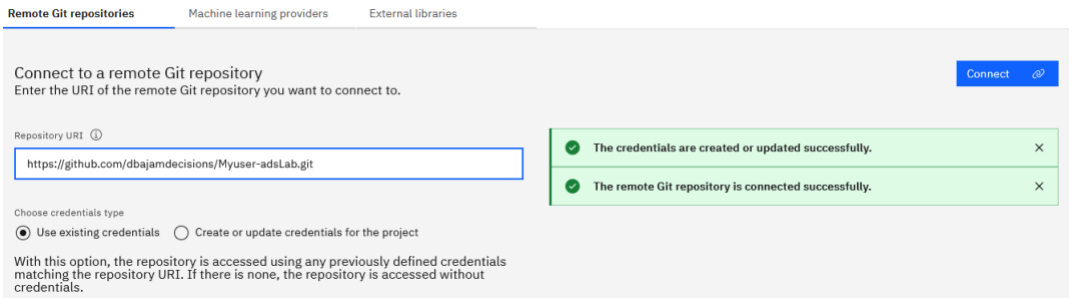
1. Check the status of **Remote Git repository** in the upper right corner of Decision Designer. It shows that the project is not connected.



2. Click on this button to make a connection.
3. Enter the **Git HTTPS URI**, the **username** used to create the repository, the **personal access token** as the password and then click on **Connect**.



Upon successful connection, Decision Designer displays the following messages and updates the status of the Remote Git repository.

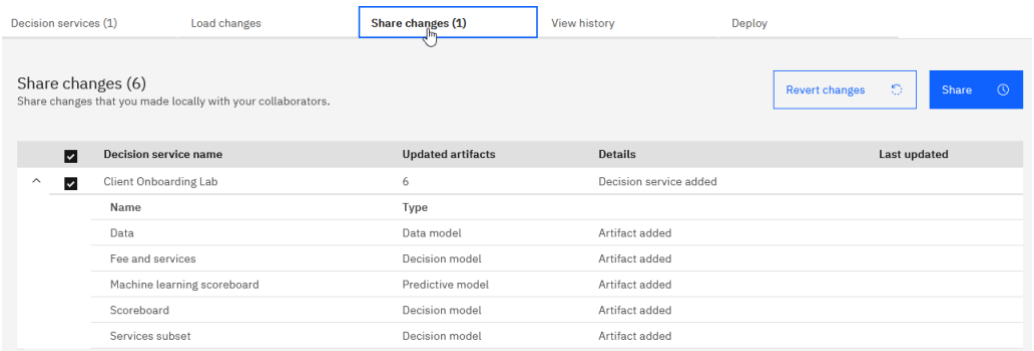


4.2.6 Sharing your decision service

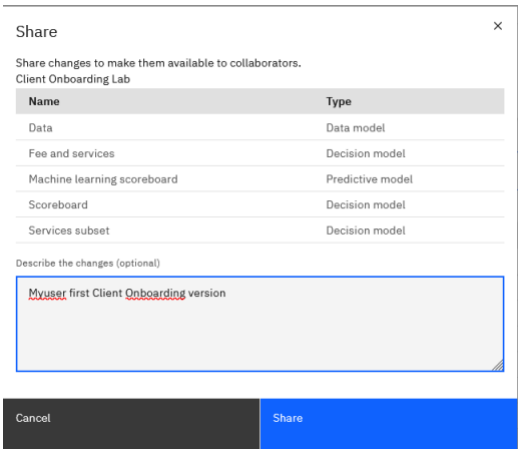
The changes made on a decision service become visible when you share them. You give the access rights in Business Automation Studio to your collaborators.

1. Return to MyUser Client Onboarding and click on the **Share changes** tab

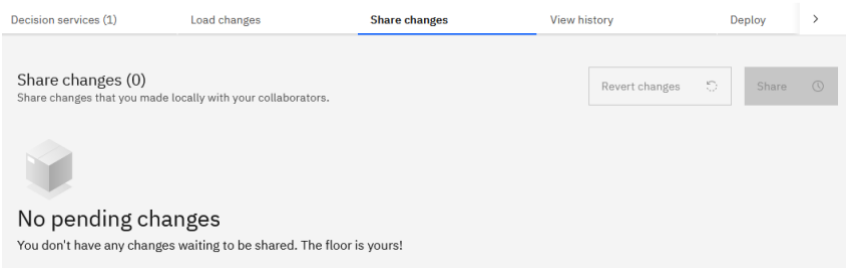
It displays the changes you made to the Decision service. It shows the number of changes.
You see artifacts of your decision service. In your project the number of changes may be different.



2. Click on **Share**.
3. Enter a comment to describe the changes (`Myuser first Client Onboarding version`) and then click on **Share**.



4. As you see there are no pending changes now.

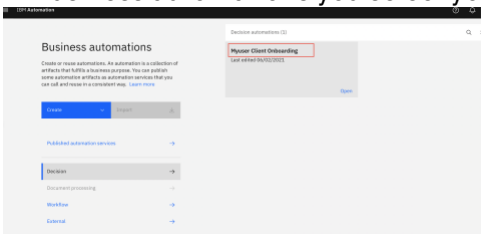


Sharing changes means that updates done locally are published and visible to other users provided that you give them access.

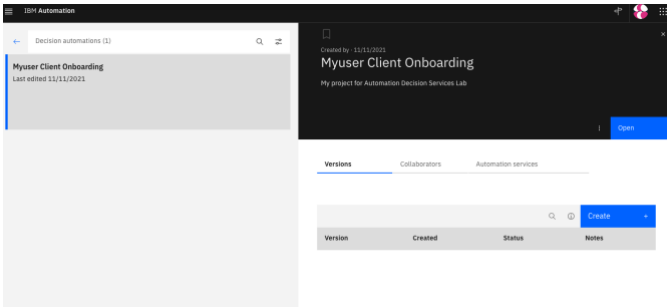
In this Lab, adding other users is not covered, however below is the procedure you would follow.

To share a project with other users, click **Business Automations** in the breadcrumbs or go to **Business automations** from the upper left menu (IBM Automation, Expand Design, Select Business automations, then select Decision.

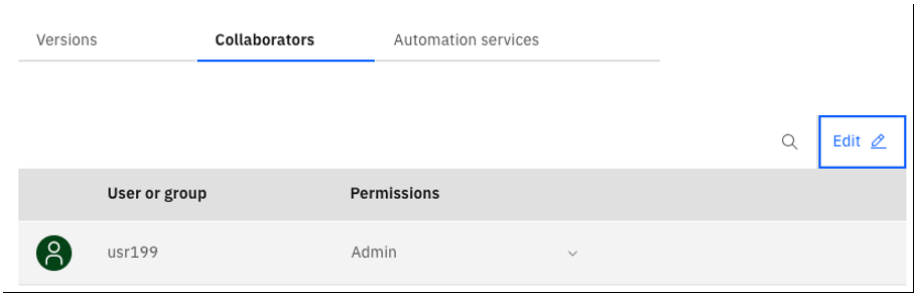
1. In Business automations you select your project by clicking on your project name.



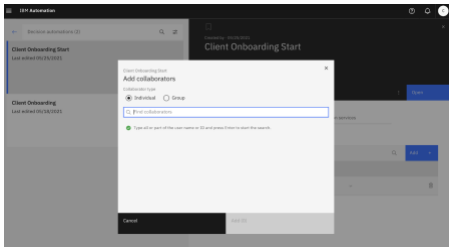
2. It opens.



3. Then, you select the **Collaborators** tab. You see that you have Admin permissions on your project. To add collaborators, click **Edit** and the name of the users you want to add and select the permissions (Admin, Edit or Read).



4. Add the name of the users you want to add and select the permissions (Admin, Edit or Read). Click Cancel (adding users is not covered in this lab).

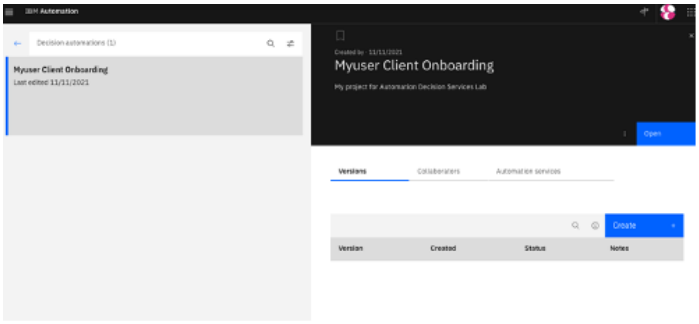


4.2.7 Creating a version

To deploy your decision service, you need to create a version of the current content. Versions correspond to tags in Git. A version is a snapshot of the project and records a point-in-time of the decision services within the project.
Creating versions requires following permission types for the project:
- **Admin** - Administrative privileges
- **Edit** - Write permission

For more information, see IBM Documentation [Managing access to projects]

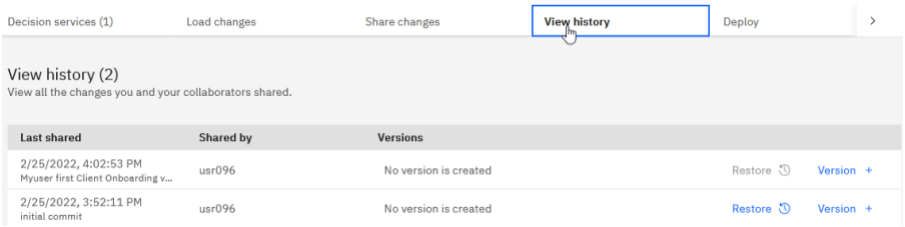
1. Return to your project by clicking on **Open** if you see the screen below



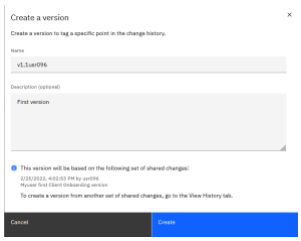
2. or navigate from the **navigation menu** in the upper-left corner to show (IBM Automation, Expand Design, Select Business automations, Select Decision, Select your project and click Open)
3. The project opens as below

4. Open the **View History** tab.

   It shows all the changes made to the project.

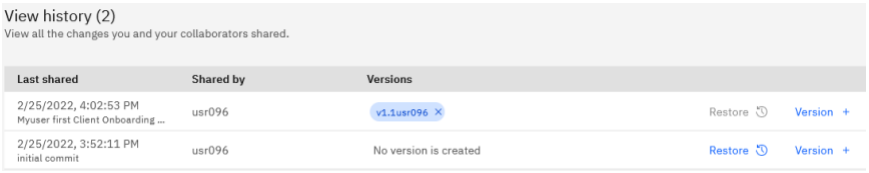


5. Select the most recent version and Click on **Version +** button to create a version of your project
6. Enter a name for this version (add your user number in the name i.e v1.1usr090) and a description



7. Click on **Create**

   A new version is created



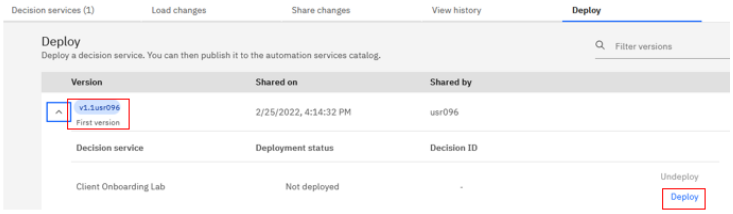
   You are now ready to deploy your version.

4.2.8 Deploying your project

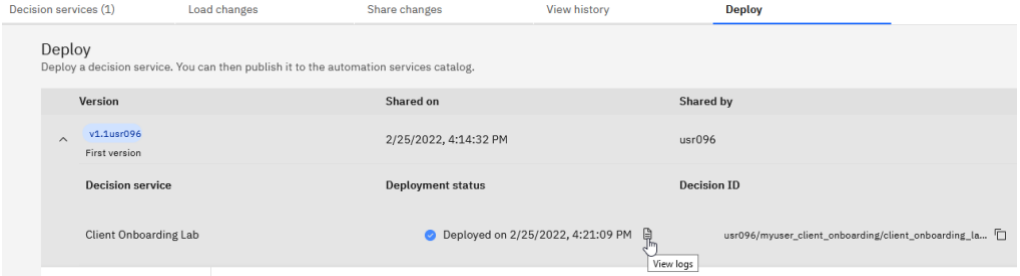
You build and deploy your decision service directly in Automation Decision Services. A decision service archive is built from your decision service and is deployed to the decision runtime in the deployment space ID named ‘embedded’. It is ready to be published as an automation service.

1. Click on the **Deploy** tab and expand the **Version** your created in the previous step.

   Note: in this example the name of the version is 1.1usr090, in your exercise add your user number as the name of the version.



2. Click on **Deploy** and **Deploy** again in the confirmation window. This triggers a build and deployment through the embedded repository for runtime archives. Wait for the deployment to finish.
3. Once completed, you can check the logs by clicking the **View logs button** in the Deployment status.



Note that a **DecisionId** has been added. The decisionID parameter is required to call the decision service in the runtime service. It provides the decision path to the generated decision service archive.

4.2.9 Executing your decision through the ADS runtime

To directly execute your decision in the ADS runtime:

1. Click on {..} next to the decision id. This opens the Swagger UI tool dedicated to the REST Api generated for your decision service archive.

2. Expand Post /feeAndServices/execute, click on Try it out, and enter the following values:

```
{
 "client": {
  "annualRevenue": 27500000,
  "companyAge": 25,
  "defaultedPayment": false,
  "numberOfEmployees": 350
 },
 "services": {
  "servicesRequested": [
   "Employee Benefits Plan",
   "Mental Health Care",
   "Onsite Medical Testing",
   "Virtual Medical Assistance"
  ],
  "Industry": "Healthcare"
  }
}
```

The response code is 200 and you get the following body:

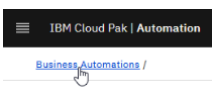
```
{
 "extendedServices": [],
 "servicesFee": 47500
}
```

Note: You may get a different result if you did not do Exercise 1.

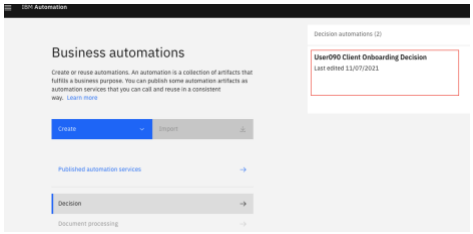
4.2.10 Publishing your decision service through Decision Designer

To publish the version of your decision deployed into the catalog of **Automation Services**, this is the procedure.

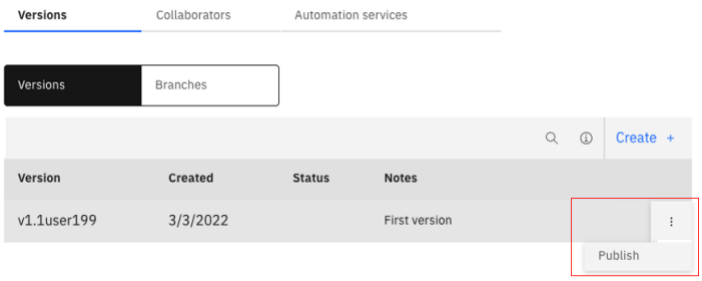
1. Click on **Business Automation** in the breadcrumbs.



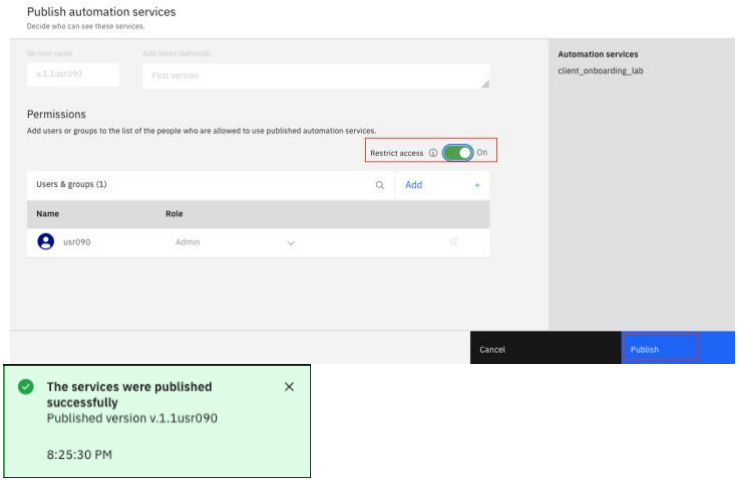
2. In the tile of the Decision automation, **click** on the name of your Decision.



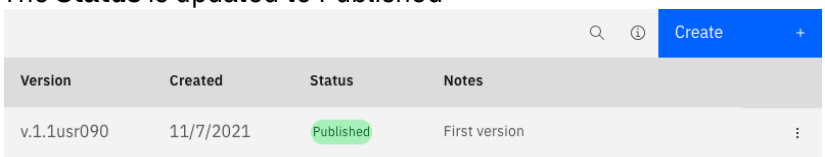
3. Click on the 3 dots menu to select **Publish.**



4. The next dialog lets you set users or group permissions to use the automation services published. **Set the restrict access to Yes** to prevent mixing with other pre-defined automation services (this Lab is shared environment), and then click on **Publish.**

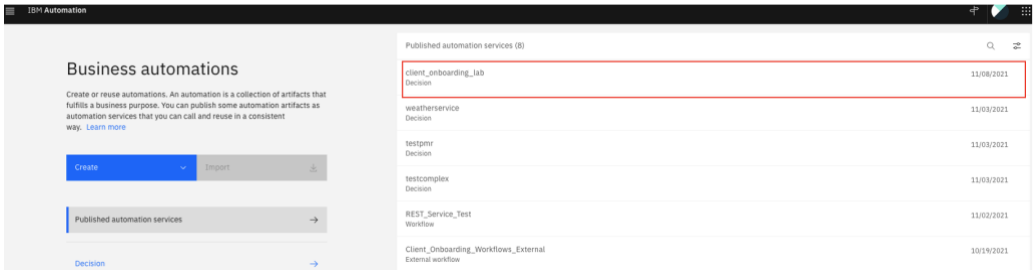


The **Status** is updated to Published



5. Navigate back to IBM Automation page and check Published automation services.

   Your decision service is listed



   Once the archive is published as an automation service, you can execute it in other Cloud Pak for Business Automation capabilities. Look at the Workflow and Business Automation Application labs to learn how to work with pre-published decisions.

4.3 Summary

You have completed the Exercise 3 - Sharing and publishing decision service.

- You made updates to your decision services visible for other collaborators by sharing the changes.
- You connected your decision project to a Git repository.
- You created a version and explored the procedure to deploy and publish a decision archive.

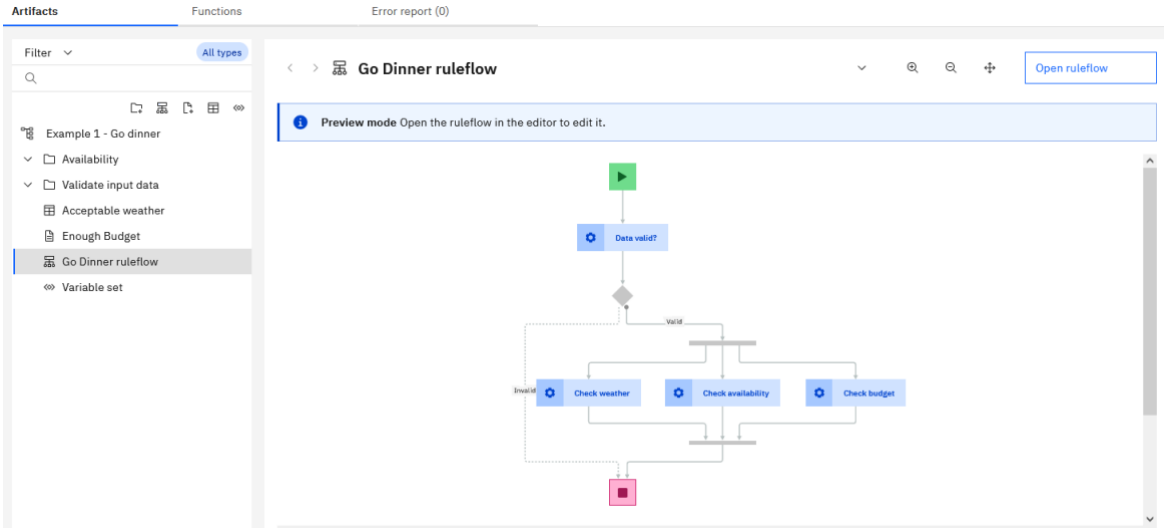
Congratulations on completing the lab!

5 Additional information

5.1 Explore the Samples

Samples and tutorials for Automation Decision Services
Samples and tutorials in GitHub

The repository of decision services includes the Training sample, which has several decision services to introduce the main features in Automation Decision Services: diagrams, business rules, decision tables, rule policies, and data and task models. The sample includes a series of decision models that gradually increase in complexity in defining a decision service. The following image shows a task model from the Training sample:



5.2 Consult Documentation and Communities

IBM Documentation
IBM Automation Decision Services
Decisions
Glossary

IBM Business Automation Community
Decision Management

DMN
Decision Model and Notation