Use the Azure ML Connector in your OutSystems applications

Use the Azure ML Connector to communicate with an Azure Machine Learning Score Model in real time and return prediction results.

Prerequisites

Before using Azure ML Connector in your applications, you need to do the following:

1. Create Training and predictive experiments in Azure ML Studio.
2. Deploy the predictive experiment as New.

Use the connector in your OutSystems application

To use the Azure ML connector in your OutSystems applications, you need to perform the following steps (detailed below):

I. Get the API Key and Endpoint for the created predictive experiments.

II. Create data structures to hold Request and Response JSON strings.

III. Add a reference to the connector in your application.

IV. Use the connector in your logic flow.
Step I. Get the API Key and Endpoint for the created predictive experiments

The API Key and Endpoint values act as input parameters for the action of the connector in your logic flow. To obtain these values, do the following:

1. Go to the [list of Web Services](https://success.outsystems.com/Documentation/11/Extensibility_and_Integration/Artificial_Intelligence/Use_the_Azure_ML_C...) and select the required experiment. Alternatively, select the **Web Services** tab from the menu of the [Azure Machine Learning Web Services homepage](https://success.outsystems.com/Documentation/11/Extensibility_and_Integration/Artificial_Intelligence/Use_the_Azure_ML_C...) and select the required experiment.

2. Click on **Use Web Service** to go to the **Consume** tab in the Menu.
3. Copy the values of the **Primary Key** and the **Request-Response** fields and save them in a document.
The **Primary Key** value is the API Key and the **Request-Response** value is the Endpoint.

### Step II. Create data structures to hold Request and Response JSON strings

To create data structures from the Request and Response JSON strings, do the following:

1. Switch to the **Swagger API** tab in the menu from the web service detail page on the Azure ML Web Service website.

2. Click on **Submit Request** to expand the section displaying the JSON strings for **Execution request** and **ExecutionResults**.
3. In Service Studio, go to the Data tab, right-click the Structures folder and select Add Structure from JSON.

4. Name the structure and enter the JSON string for Execution request in the input box. Delete the empty “Global Parameters” object and click the Add Structure button to create the structure.
5. Repeat steps 2 and 3 to obtain a structure for the **ExecutionResults** JSON string.
Step III. Add a reference to the connector in your application

To add a reference to the connector in your application, you need to do the following:

1. In Service Studio, click the Manage Dependencies toolbar button.
2. Select AzureMLConnector from the list of Producers and check all the elements.
3. Click on OK to add a reference to the elements of the connector in your application.

Step IV. Use the connector in your logic flow

To use the connector in your logic flow, do the following:
1. Create two local variables TestValues and Request_Example. Set the data type of TestValues as Input1Item and of Request_Example as the **Execution request** structure.
   - TestValues holds the values to be set.
   - Request_Example is the **Execution request** structure with the values of Var1 appended to it. This acts as an input parameter of the connector.

2. Drag TestValues to the flow and fill in the required fields.

3. Drag a **ListAppend** action to the flow and select the values of the required input parameters from the dropdown.

4. Add the **JSON Serialize** action to the flow to serialize the **Execution request** structure to a JSON string.

5. In the Logic tab, expand the Server Actions folder and open the **AzureMLConnector** dropdown. Drag the **PredictResults** action to the desired flow.
   - In the RequestBody_JSON field of the PredictResults action, select the output of the JSON Serialize.
Go back to the document where you had saved the values of the API Key and the Endpoint. Enter the Endpoint value in the EndpointURL field and the API Key value in the AuthToken field.

6. To deserialize the JSON string output from the connector and map it to the `ExecutionResults` data structure, drag a JSON Deserialization action to the flow and enter `PredictResults.ResponseBody_JSON` in the JSON String field.

7. Publish and test.