OpenAI Decisions API

Readme

Version 1.0.1
06/11/2026

Table of Contents

1. Introduction . . . 3

1.1 Overview . . . 3

1.2 Use cases . . . 4

2. Requirements & Prerequisites . . . 5

2.1 System Requirements . . . 5

2.2 Prerequisites . . . 5

3. Getting Started . . . 6

3.1 Quick Start . . . 6

3.1.1 Setup . . . 6

3.1.2 Configuration and Use . . . 6

4. Support & FAQs . . . 10

4.1 Support . . . 10

4.2 FAQs . . . 10

Appendix A: Record of Changes . . . 11

Appendix B: References . . . 11

1. Introduction

This document contains all essential information for the user to make full use of this package. It includes a description of the functions and capabilities and step-by-step procedures for setup and configuration of the package.

1.1 Overview

The OpenAI Decisions API package adds four decision actions to Automation 360: **Predicate**, **Choice**, **Score** and **Evaluate**. Each action takes content, such as an email body or an image, together with an instruction and a set of criteria written in plain language. The package sends them to the OpenAI Responses API and returns a structured Dictionary. The Dictionary holds the decision, a confidence value where applicable and a short rationale, so a bot can use the result directly in If conditions, routing rules and reports without parsing free text.

How it works from start to finish

1. The bot supplies the content to review, as text or as a JPG, JPEG or PNG image, and an optional state value that is passed through to the result.
2. The bot supplies an instruction and criteria that describe the decision to make.
3. The package builds the request and calls the OpenAI Responses API with your API key.
4. The package checks the answer against the structure expected for the selected action and returns it as a Dictionary.
5. The bot assigns the Dictionary to a variable and reads the keys it needs, for example `selected_choice` or `confidence`.

Actions and the keys they return

Action	What it does	Keys in the returned Dictionary
Predicate	Answers a yes/no question about the content.	result, confidence, rationale, state
Choice	Selects the best-matching option from a list you define.	selected_choice, confidence, rationale, state
Score	Rates the content on a numeric scale you define.	score, scale_min, scale_max, rationale, state
Evaluate	Runs any combination of Predicate, Choice and Score on the same content in one call.	Optional keys predicate, choice and score. Each key holds the Dictionary of that action.

Sample input and output

The sample below uses a customer email as the input. The values in the output are illustrative. Actual results depend on the model, the instruction and the criteria you configure.

SAMPLE INPUT	
Item	**Value**
Content (State)	I was charged twice this month and need the duplicate refunded today. Your portal also keeps logging me out.
Action	Choice
Instruction	Which team should handle this email?
Criteria	Billing (charges, refunds, invoices); Technical (login problems, errors, outages); Sales (pricing, upgrades, new purchases); General (anything else)

SAMPLE OUTPUT (ILLUSTRATIVE)	
Key	**Example value**
selected_choice	Billing
confidence	0.91
rationale	The customer reports a duplicate charge and asks for a refund.
state	The state value supplied by the bot, if any

Note: Decisions are generated by an AI model. Review the results, and the criteria behind them, before you rely on them in customer-facing processes, external communications or official records. The confidence value is a useful signal for sending uncertain cases to a person for review.

1.2 Use cases

The key use cases include:

- Route emails, tickets and service requests to the right team based on their content.
- Rate the urgency, priority or sentiment of incoming messages, for example how frustrated a customer is.
- Check whether a message or document meets a condition, such as stating a deadline or containing a required clause.
- Approve, flag or reject items against criteria that your team defines.
- Classify content from images, such as a scanned note or a screenshot, with the same criteria used for text.
- Send low-confidence decisions to a person for review before the process continues.
- Run several questions on the same content in one call to reduce the number of actions in a bot.

2. Requirements & Prerequisites

2.1 System Requirements

Enterprise (Cloud deployed) and Community Edition device requirements.

Review the machine hardware specifications, operating system versions, and browser types supported by Automation Anywhere Enterprise for creating and running bots and packages as an Enterprise (Cloud deployed) or Community Edition user on your local machine.

In addition, this package requires:

Requirement	Details
Control Room	Automation 360 Control Room, build 8750 or later
Bot Agent	Version 22.11 or later
Operating system	Windows or macOS on the device that runs the bot
Network	HTTPS access from the bot device to https://api.openai.com/v1/responses

2.2 Prerequisites

Prerequisite	Details
OpenAI account	An OpenAI account with API access and billing enabled. OpenAI charges for API usage to your account.
OpenAI API key	A secret key used to authenticate each request. See section 3.1.1.
OpenAI organization ID	Optional. Needed only when your requests must carry the organization header.
Model	A model that supports structured outputs. The default is gpt-6-luna.
Control Room permissions	Permission to upload packages, create bots and create Credential Vault lockers and credentials.
Package file	OpenAI_Decisions_API-1_0_1 package (.jar) downloaded from the Bot Store.

3. Getting Started

3.1 Quick Start

3.1.1 Setup

Step 1: Create an OpenAI API key

1. Sign in to the OpenAI platform at platform.openai.com.
2. Open the API keys page and select the option to create a new secret key.
3. Give the key a name, create it and copy the value. OpenAI shows the full value only once, so store it in a safe place.
4. If your requests need an organization header, copy your organization ID from the organization settings page.

Menu names in the OpenAI dashboard can change. Refer to the OpenAI documentation for the current steps.

Step 2: Store the API key in the Credential Vault

Keep the API key in the Credential Vault so that it does not appear in bot files.

1. In the Control Room, open the Credential Vault and create a locker, or use an existing one.
2. Create a credential in the locker and add the attributes shown in the table below.
3. Give the bot developers and runners access to the locker.

Locker Name	Credentials Name	Attribute Name	Value
Locker_OpenAI	Credential_OpenAI	API_KEY	Your OpenAI API key (mark the attribute as masked)
Locker_OpenAI	Credential_OpenAI	ORG_ID	Your OpenAI organization ID (optional)

Step 3: Add the package to the Control Room

1. Download the package (.jar) from the Bot Store.
2. In the Control Room, open the Packages page and add the package by uploading the .jar file.
3. Set the uploaded version as the default version, so that it is available in the bot editor.

For full instructions, see Automation 360 documentation.

3.1.2 Configuration and Use

For Packages:

Drag any of the four actions from the **OpenAI Decisions API** package into your bot and complete the fields described below.

Fields shared by all actions

Field	Required	Description
API key	Yes	The OpenAI API key. Select the credential from the Credential Vault.
Organization ID	No	The OpenAI organization ID, sent in the OpenAI-Organization header.
Model	No	The model identifier. It must support structured outputs. Default: gpt-6-luna.
State source	Yes	Select **Text** to enter the content as text, or **Image** to supply a JPG, JPEG or PNG image file.
State	Yes	The content to analyze. The same value is returned in the `state` key of the result.
Response timeout	No	Response timeout in milliseconds. Default: 30000.
Connection timeout	No	Connection timeout in milliseconds. Default: 30000.

Predicate

Answers a yes/no question about the content.

Field	Required	Description
Name	Yes	A name for this action configuration, for example hasDeadline.
Instruction	Yes	The yes/no question for the model, for example: Does the customer ask for action by a specific time?
Criteria	Yes	One or more rows, each with a name and a description, that explain what counts as a yes.

Returns a Dictionary with the keys `result, confidence, rationale, state`.

Choice

Selects the best option from a list that you supply.

Field	Required	Description
Name	Yes	A name for this action configuration, for example team.
Instruction	Yes	How the best option must be selected, for example: Which team should handle this email?
Criteria	Yes	The options, one row each, with a name and a description of when to choose it.

Returns a Dictionary with the keys `selected_choice, confidence, rationale, state`.

Score

Rates the content on a numeric scale that you define.

Field	Required	Description
Name	Yes	A name for this action configuration, for example frustration.
Instruction	Yes	How the content must be scored, for example: How frustrated is the customer?
Criteria	Yes	The levels of the scale, one row each, with a name and a description.

Field	Required	Description
Scale minimum	Yes	The lowest allowed score. Default: 0.
Scale maximum	Yes	The highest allowed score. It must be equal to or greater than the minimum. Default: 100.

Returns a Dictionary with the keys `score, scale_min, scale_max, rationale, state`.

Evaluate

Runs Predicate, Choice and Score questions on the same content in a single call.

Field	Required	Description
Predicate, Choice, Score checkboxes	At least one	Select each decision to run. Each selected decision shows its own fields.
Name, Instruction, Criteria	Yes, for each selected decision	Same meaning as in the individual actions.
Score scale minimum and maximum	For Score	The allowed range for the score.

Returns a Dictionary with the optional keys `predicate, choice, score`. *Each key holds the Dictionary of the selected action.*

Reading the result in a bot

Assign the output of each action to a Dictionary variable. Read a key with the Dictionary key syntax, for example `$dChoiceAnswer{selected_choice}$` or `$dChoiceAnswer{confidence}$`. To view the whole result as text, use the Dictionary to JSON action from the JSON package.

Sample bot

The package ships with a sample bot that shows how to use all four actions on one customer email. Follow these steps to run it.

1. Import the sample bot into your Control Room and open it in the bot editor.
2. In the variables list, set **API_KEY** to your OpenAI API key and **sOrg_ID** to your organization ID. Leave **sOrg_ID** empty if your requests do not need it. For production use, select the API key from the Credential Vault in each action instead of keeping it in a variable.
3. Review **sState**. It holds the sample email that the bot analyzes. Replace it with your own text if you wish.
4. Run the bot. It runs four steps in this order.

Step	Action	What the bot does
1	Choice (team)	Selects Billing, Technical, Sales or General. If confidence is 0.80 or higher, a message box names the team. Otherwise it says the email needs human review.
2	Score (frustration)	Scores customer frustration from 0 (Calm) to 2 (Angry). If the score is 1.0 or higher, a message box says to escalate to a senior agent.
3	Predicate (hasDeadline)	Checks whether the customer asks for action by a specific time. If confidence is 0.80 or higher, a message box says to set the ticket priority to High.

Step	Action	What the bot does
4	Evaluate	Runs all three decisions above on the same email in a single call and shows the combined result.

Each step also displays the full result as JSON in a message box, so you can see every key that the action returned.

Important: Before you share or publish a bot, make sure that no API key or organization ID is stored as a variable value. Remove the default values of API_KEY and sOrg_ID.

For configuring the bot:

INPUT VARIABLES				
Variable Name	**Type**	**Mandatory**	**Purpose**	**Example Input**
API_KEY	Text	Yes	OpenAI API key used to authenticate the requests.	Your OpenAI API key
sOrg_ID	Text	No	OpenAI organization ID sent with the request.	org-xxxxxxxx
sState	Text	Yes	The content that the actions analyze.	I was charged twice this month and need the duplicate refunded today.

OUTPUT VARIABLES				
Variable Name	**Type**	**Mandatory**	**Purpose**	**Example Output**
dChoiceAnswer	Dictionary	Yes	Result of the Choice action.	selected_choice: Billing
nConfidence	Number	No	Confidence read from the Choice or Predicate result.	0.91
dScoreAnswer	Dictionary	Yes	Result of the Score action.	score: 1
nScore	Number	No	Score read from the Score result.	1
dPredicateAnswer	Dictionary	Yes	Result of the Predicate action.	confidence: 0.90
dEvaluateAnswer	Dictionary	Yes	Combined result of the Evaluate action.	keys: predicate, choice, score
sChoiceAnswer, sScoreAnswer, sPredicateAnswer, sEvaluateAnswer	Text	No	The matching Dictionary converted to JSON text for display.	JSON text

The example values are illustrative and change with the content and the model.

4. Support & FAQs

4.1 Support

Free bots are not officially supported. You can get access to Community Support through the following channels:

- You can get access to community support, connecting with other Automation Anywhere customers and developers on the Community in the Developers Forum.
- Automation Anywhere also provides a Product Documentation portal which can be accessed for more information about our products and guidance on Automation 360.

4.2 FAQs

Which OpenAI model does the package use?

The default model is gpt-6-luna. You can enter another model identifier in the Model field. The model must support structured outputs.

Does the package send my data to OpenAI?

Yes. The content, instruction and criteria are sent to the OpenAI Responses API. Check your organization's data handling policies before you send personal or confidential data.

Who pays for the OpenAI usage?

The OpenAI account that owns the API key. The package itself is free.

What does the confidence value mean?

It is a value reported with the decision. The sample bot treats 0.80 or higher as sufficient to act and sends anything lower to a person. Choose the threshold that suits your process, and treat the value as a guide.

Can I analyze images?

Yes. Set State source to Image and select a JPG, JPEG or PNG file.

The action returns an error. What should I check?

Check that the API key is valid and that the account has API access and billing. Then check the organization ID, the model name, network access to api.openai.com and the timeout values. Make sure the instruction and criteria fields are not empty.

Can I run more than one decision on the same content?

Yes. Use the Evaluate action and select the decisions to run.

Do I need to review the results?

Yes. The results are produced by an AI model. Review decisions, and the criteria behind them, before using them in customer-facing processes, external communications or official records.

For questions relating to the Control Room: See the Automation 360 FAQs.

Appendix A: Record of Changes

The table below records the version number, the date of the version, the author of the version and a brief description of the reason for creating the version.

No.	Version Number	Date of Change	Author	Notes
1	1.0.1	06/11/2026	Automation Anywhere	Initial release of the package and readme.

Appendix B: References

No.	Topic	Reference Link
1	Guidance: Building basic Automation 360 bots	Click here
2	Guidance: Building Automation 360 packages	Click here
3	Community Developers Forum	Click here
4	Automation Anywhere University	Click here
5	OpenAI platform and API documentation	Click here