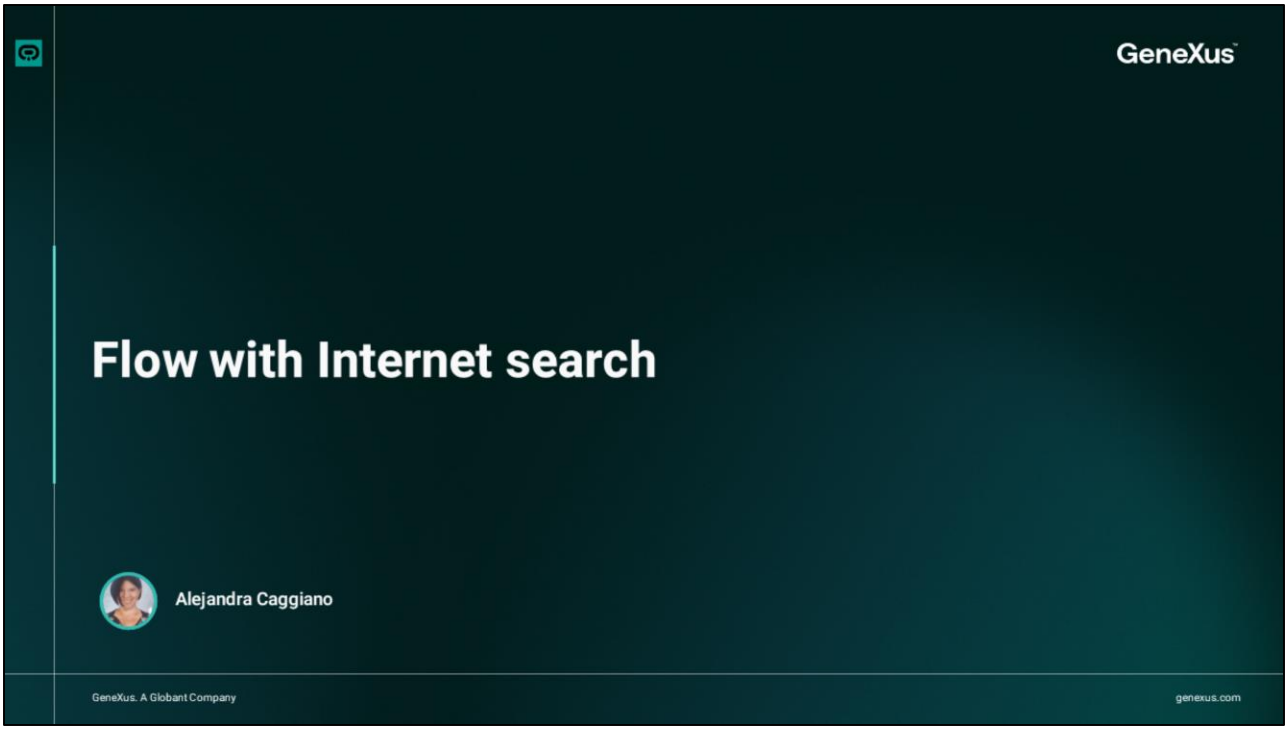
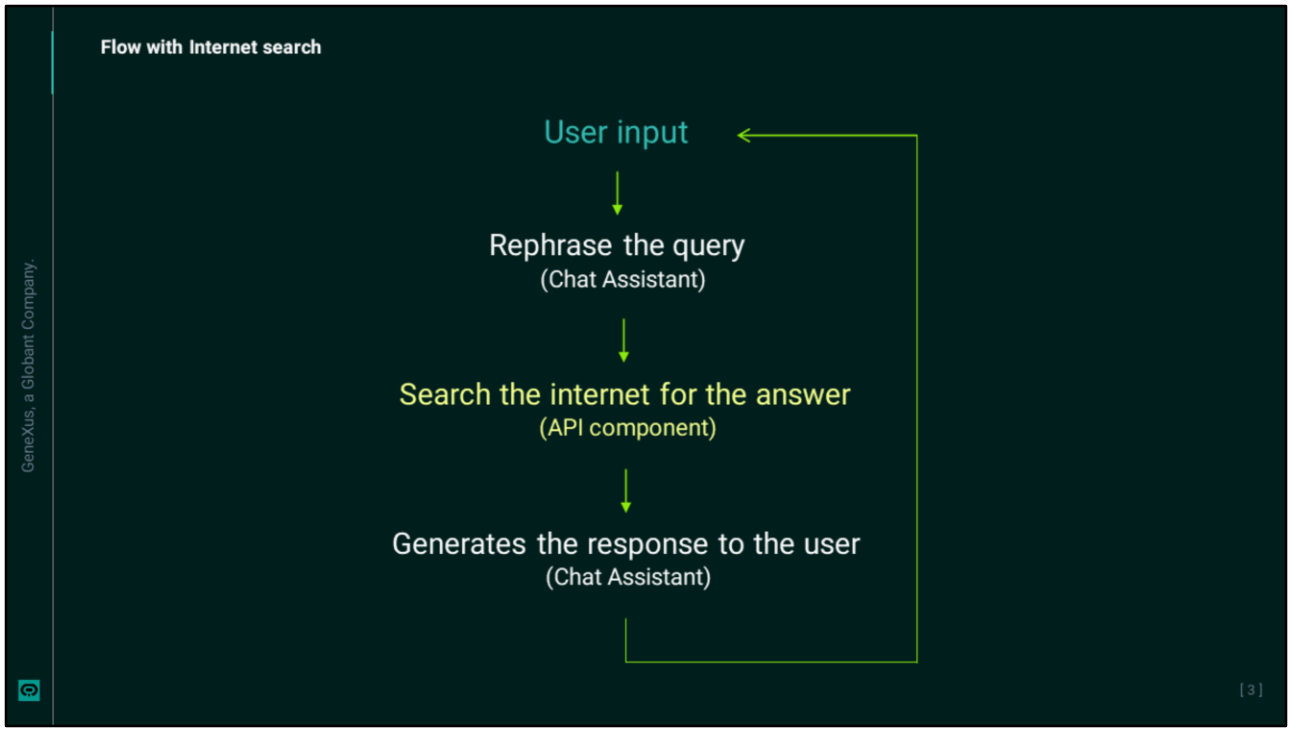

Globant
Enterprise AI

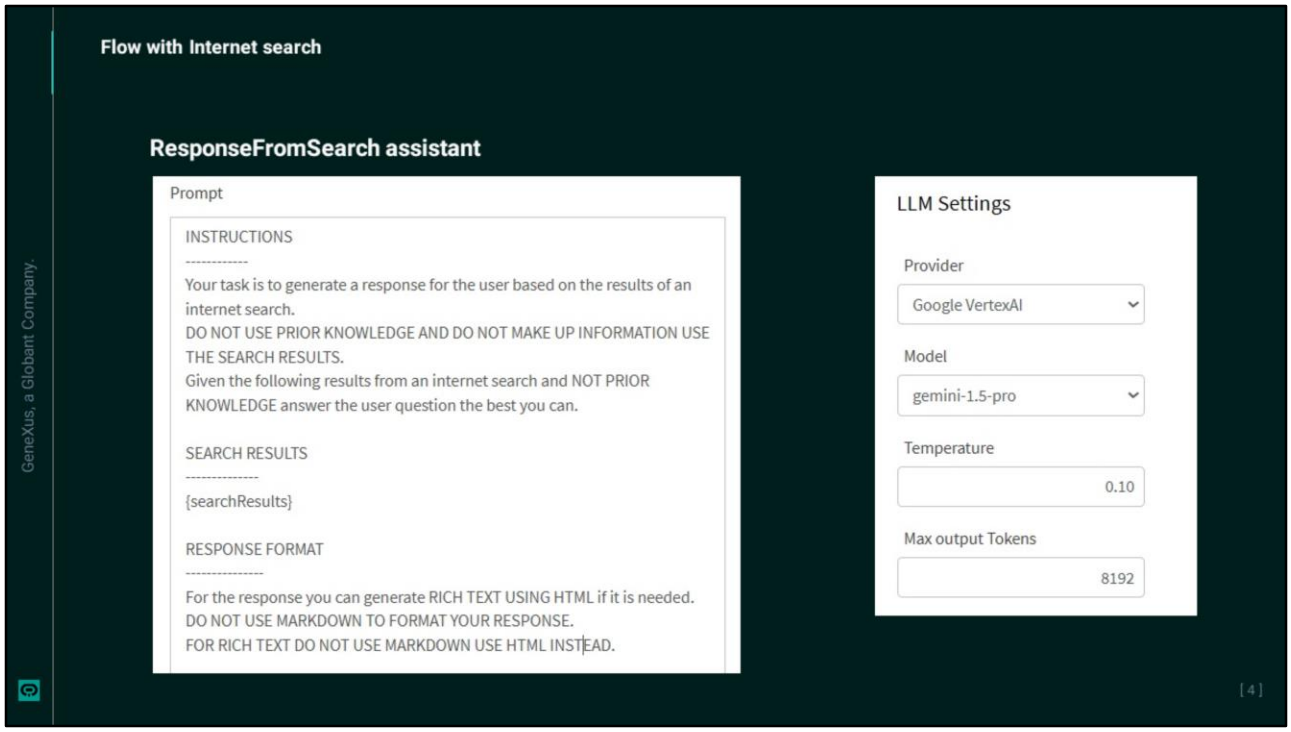


Let's create a new flow that manages two chat assistants, and includes a search API component.



One of the assistants will answer questions through internet search results, and the other will be able to rephrase the questions, based on the conversation history, as if they were independent questions.

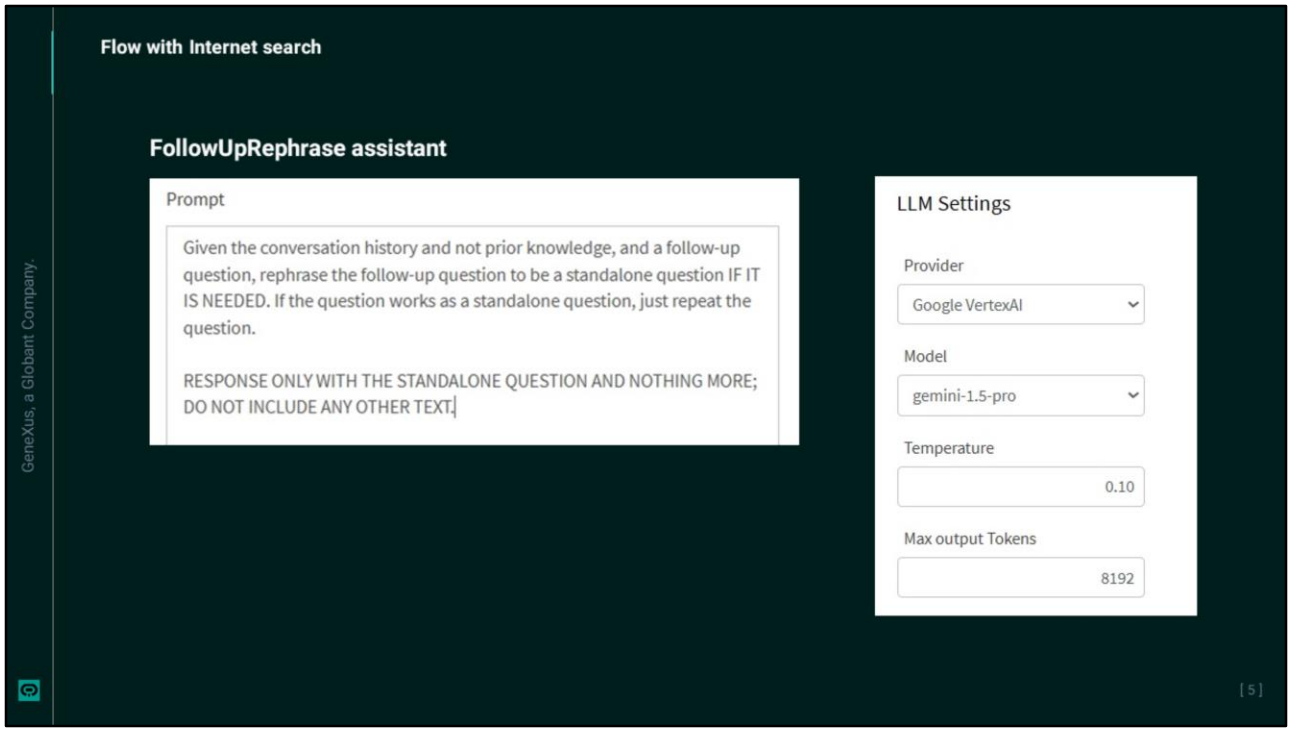
The API component will perform the internet search, using a new search engine and will be the link between the two assistants.



We access the backoffice, and in our project we see the two assistants created:
Let's look at the ResponseFromSearch assistant prompt. It states that this assistant should generate a response for the user based on the results of an internet search.

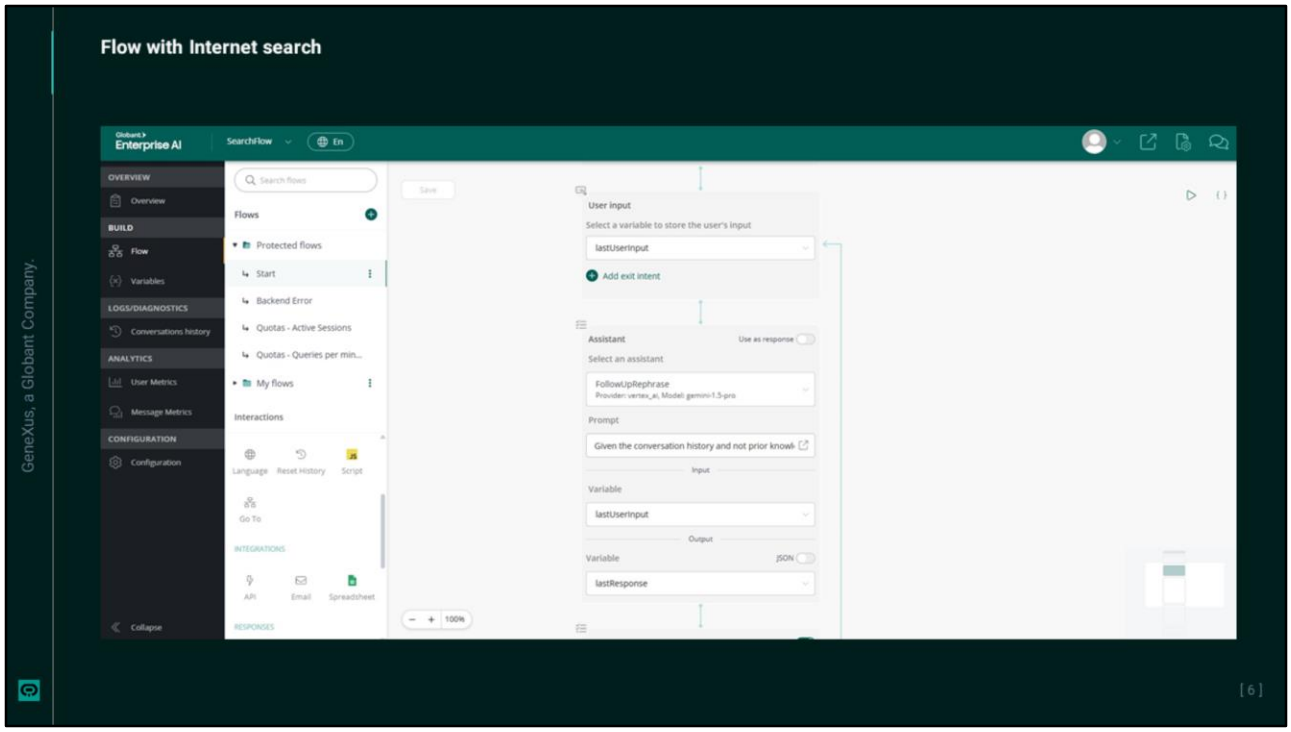
DO NOT USE PRIOR KNOWLEDGE AND DO NOT MAKE UP INFORMATION, USE THE SEARCH RESULTS.

With the results of an internet search and NO PRIOR KNOWLEDGE, it will answer the user's question as best as possible. The result will be stored in the SearchResults variable, and for the answer, it can generate rich text with HTML if necessary.



Let's now look at the definition of the other chat assistant named FollowUpRephrase.

It states that it must rephrase a question based on the history of the conversation and not on prior knowledge. It should respond based on the standalone question only, without including any other text.

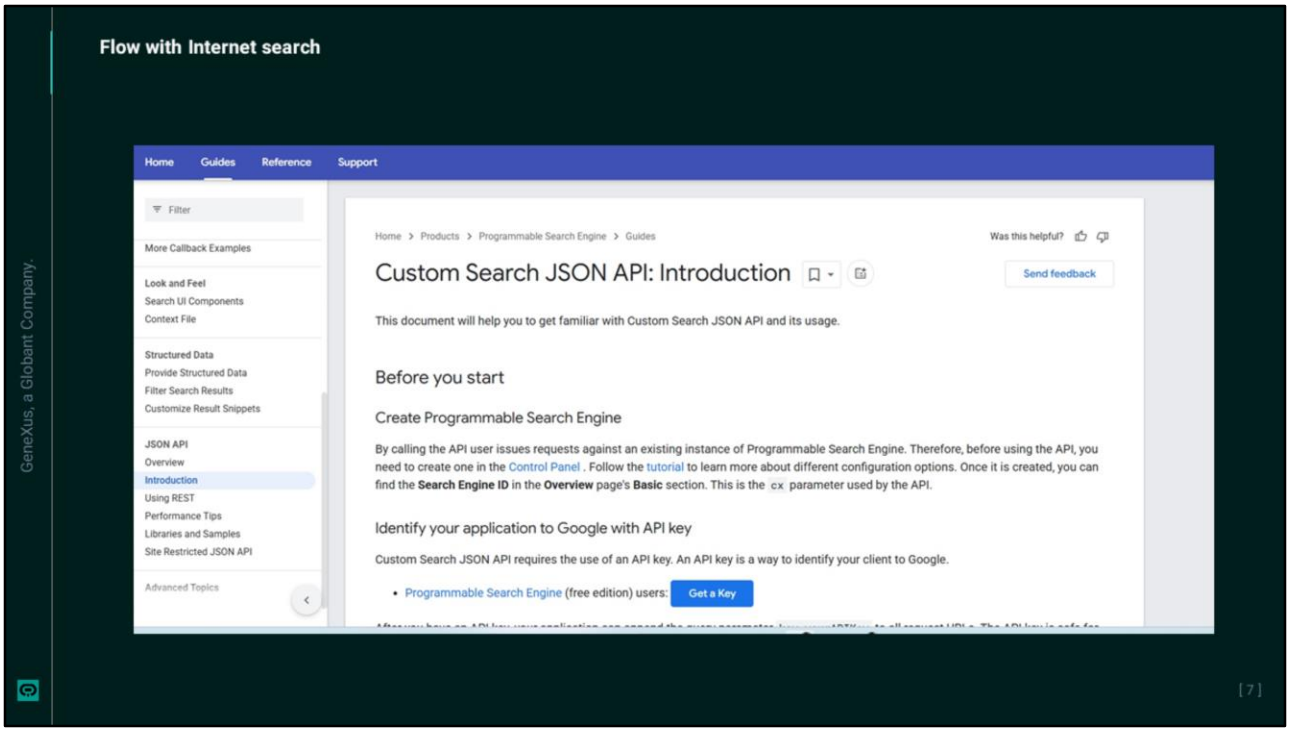


Good. Let's start defining the flow.
We access Flows, and create a new flow named InternetSearchFlow. We add a description and indicate the Spanish and English languages.

Then from the UserInput node, which receives the user's query, we connect to the FollowUpRephrase assistant. This assistant receives the query in the lastUserInput variable, but its response will be stored in a new variable. So we clear this option and indicate the lastResponse variable in the output.

This assistant will return a question whose answer will be searched for using a search engine. To do this, we need an API component to access it and act as a link between both assistants.

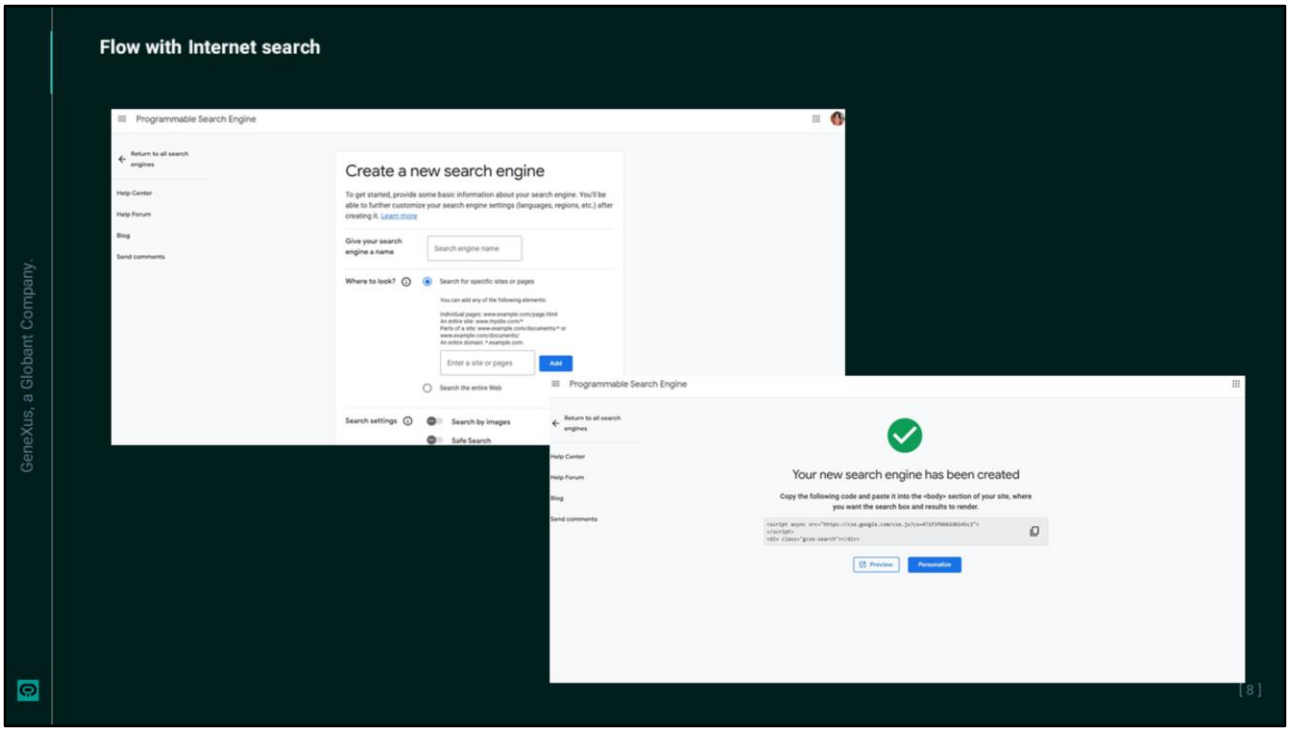
We go to the interactions, select API and drag it to its position in the flow.



To configure the necessary information we must define the search API, and for that we are going to create a new programmable search engine.

The first step is to obtain the API key, so we access the site and click on Get Key.

We select or create a project. The key that is shown must be saved because we will need it for the API component in the flow.



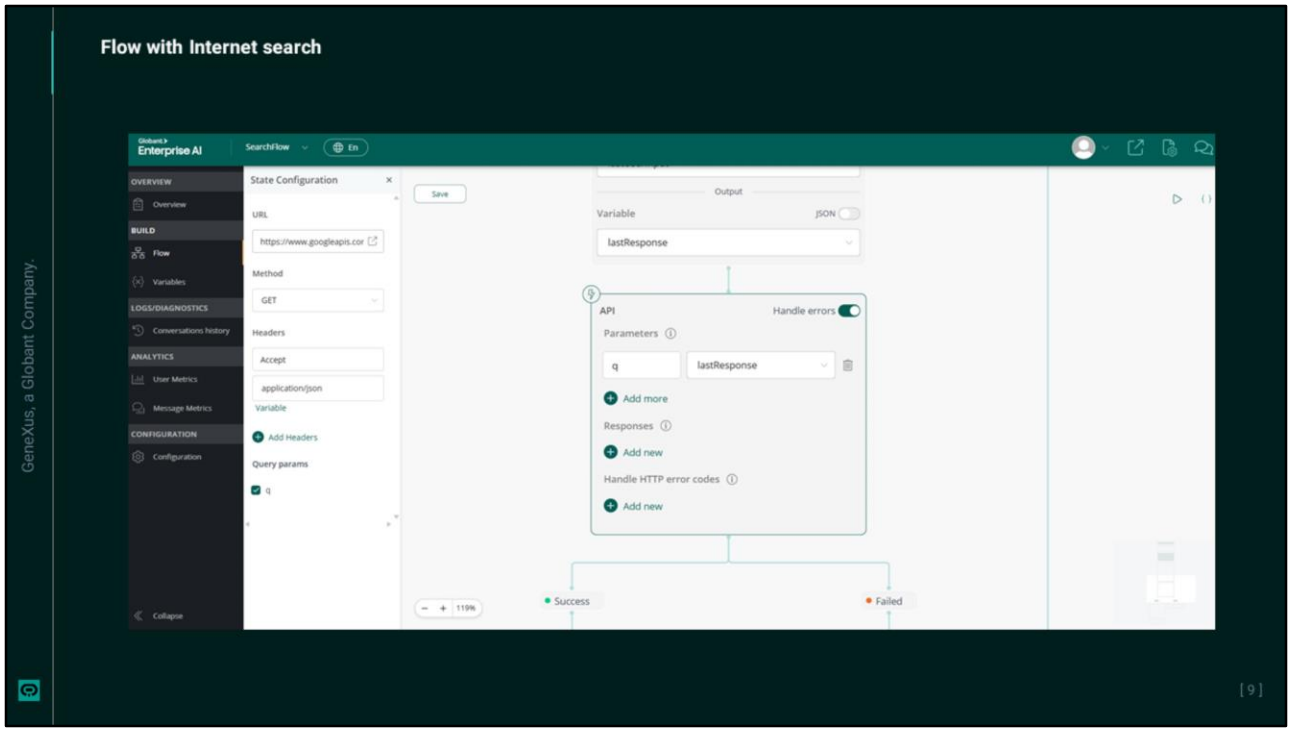
Next, we will create the new search engine.

We access the site, (Programmable Search Engine), click on Add, and fill in the necessary data.

We indicate a name for the new search engine and we also instruct it to search the entire web. We also leave additional settings such as image search or SafeSearch filter. We select the "I am not a robot" checkbox, and finally click on Create.

In the window that appears we select the button to customize.

This brings us to the general description of the search engine we have just created. We must then find and copy the Id, which is a unique value associated with the engine, and which will be necessary data for the configuration of the API node in our flow.



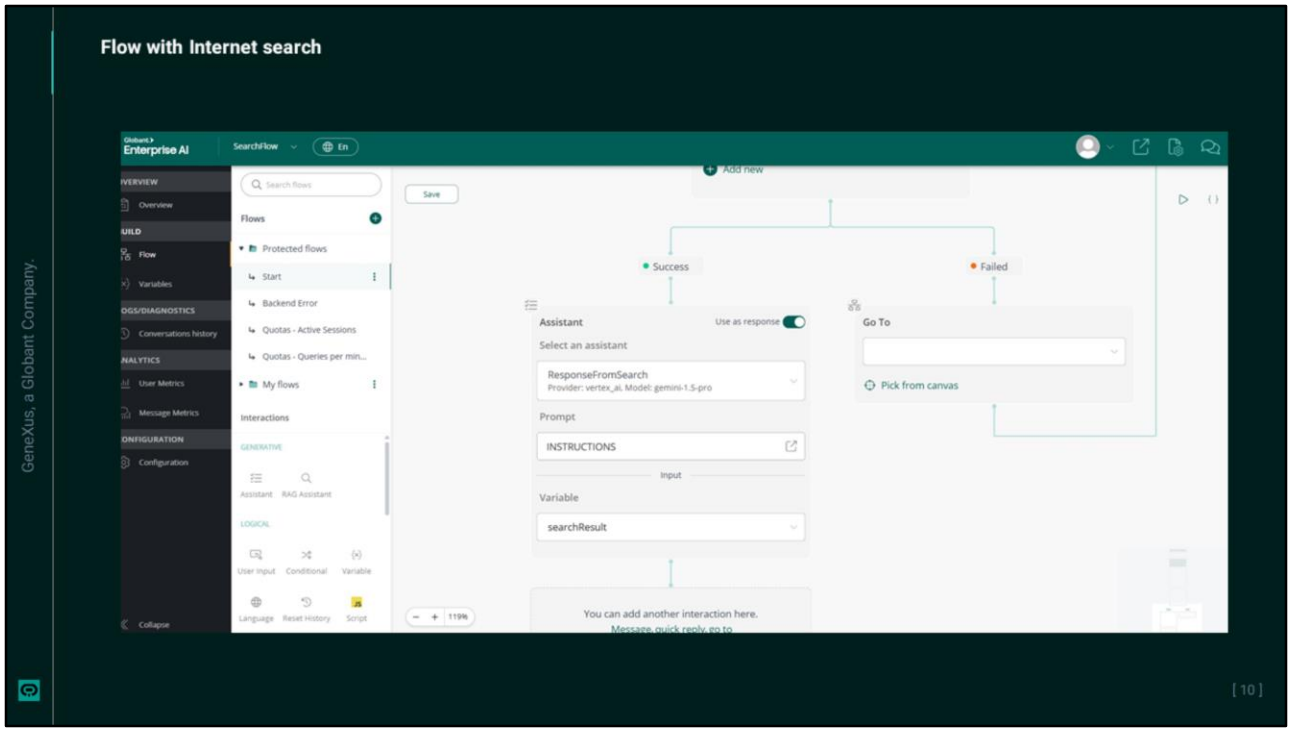
Well, we now have everything we need to continue defining the flow, so we go back and next to the API node, we select Edit.

In the left side menu, we must indicate the URL, making sure to reference the values obtained for the API key and the Id of the search engine.

In the API node, we go to the Parameters section, Add new, and configure the parameter with the following information. As parameter name, we set "q", and indicate lastResponse as the value, since it will be the value obtained from the previous response. Then in the left menu we mark "q" as the query parameter.

In the Response section of the node, we associate the items property with the searchResults variable. If the variable does not exist, we create it.

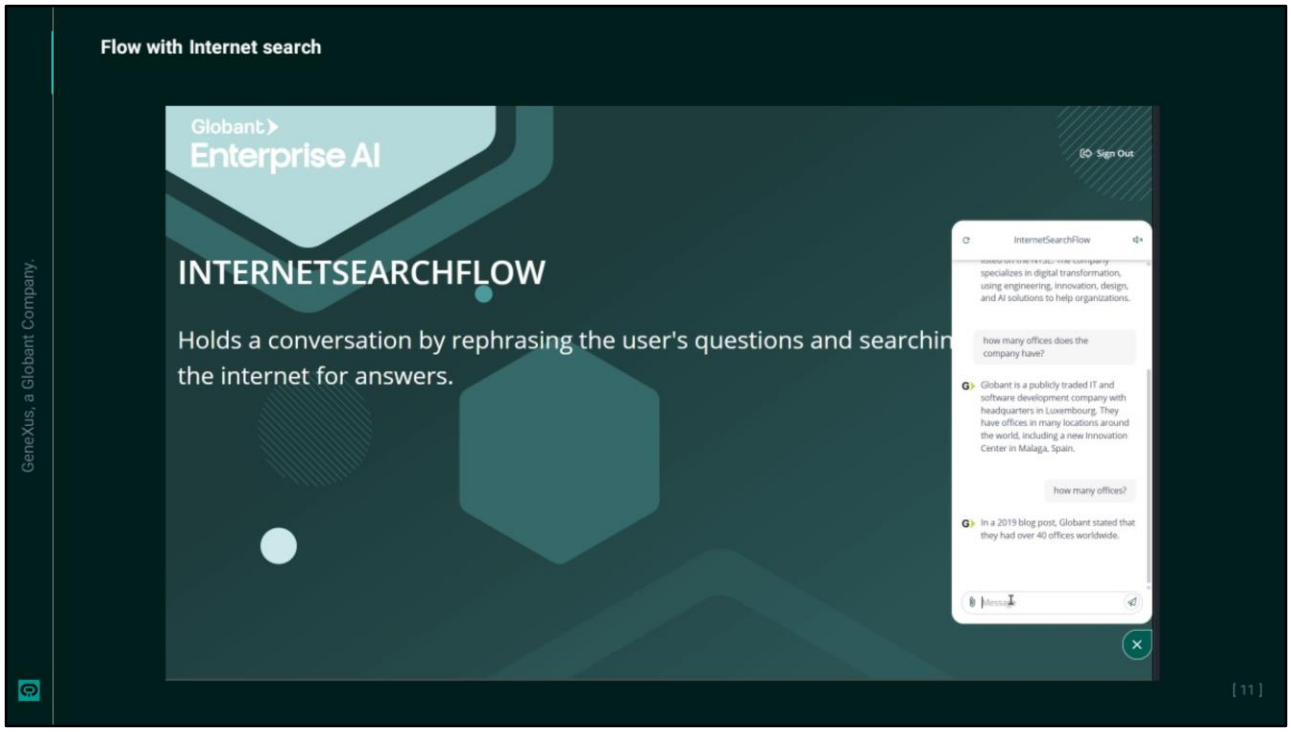
Once the API node is defined, we must make sure that the flow follows its course and that the sequence of variables is correct. The API node returns the result of the search in the searchResults variable specified as the response. So this variable should be the input in the ResponseFromSearch assistant, which will return the response to the user.



Next, from the interactions box, we drag another Assistant node and indicate the ResponseFromSearch assistant. The input of this assistant will be the searchResults variable received through the API node.

Good. We now add the corresponding Go To node to return to the UserInput node.

We have now completed the definition of the flow, so we save the changes, and now we can test its behavior.



Let's go to the test page.

We ask "What is Globant?""How many offices does it have?“

And so we can continue the conversation.


Globant
Enterprise AI