

Multiagent HR Application

To design and implement an intelligent LLM application that can analyze user intent and dynamically choose the most suitable reasoning or response framework (e.g., Agent-based or RAG-based) to deliver accurate and context-aware answers

What is Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation (RAG) is an advanced technique that enhances Large Language Models (LLMs) by combining **information retrieval** with **text generation**.

Instead of relying solely on the model's internal knowledge, RAG retrieves relevant external documents or data from a **knowledge base or vector database** and uses them to generate more accurate, up-to-date, and context-specific responses.

Key Components

Retriever:

Searches external data sources (e.g., documents, databases, or embeddings) to find the most relevant information for the user query.

Generator (LLM):

Uses the retrieved information to produce a coherent, factual, and context-aware response.

Benefits

Reduces hallucinations by grounding responses in real data

Provides access to domain-specific or private knowledge

Improves factual accuracy and contextual relevance

Example

When a user asks a question about company policies, a RAG system retrieves related documents from the company's internal database and then generates an answer based on that content — ensuring the response is accurate and source-based.

Retrieval Augmented Generation

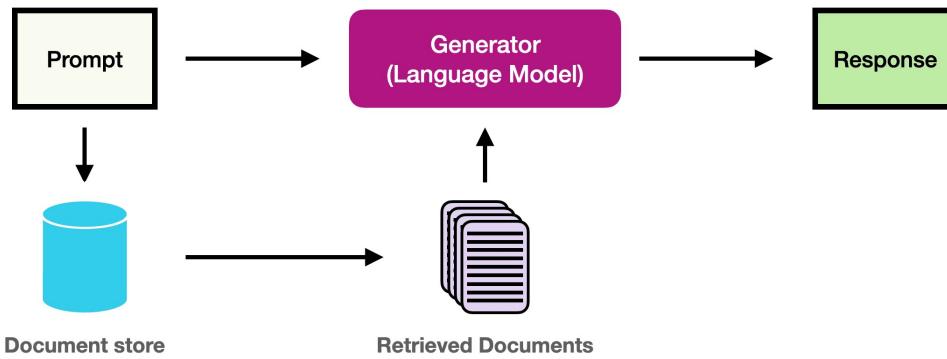


Image source: promptingguide.ai

What is Agentic AI?

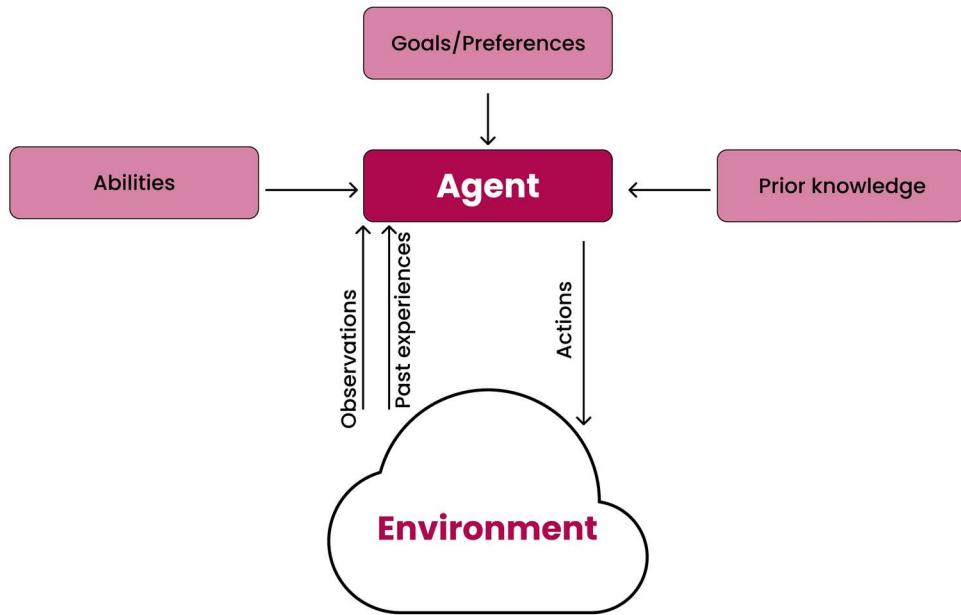
Agentic AI represents the next evolution of intelligent systems — moving beyond passive models that only generate responses, toward autonomous agents capable of reasoning, planning, and acting to achieve specific goals.

These systems use a continuous sense–think–act loop:

- Sense – Observe or gather information from the environment (e.g., data, APIs, user input).
- Think – Use reasoning or planning to decide what to do next.
- Act – Execute actions such as retrieving data, running code, or interacting with external tools.

Agentic AI integrates key components such as memory (to retain context across steps), tool use (to extend capabilities), and goal-oriented reasoning (to plan multi-step tasks).

What is an AI agent?



codiste.com

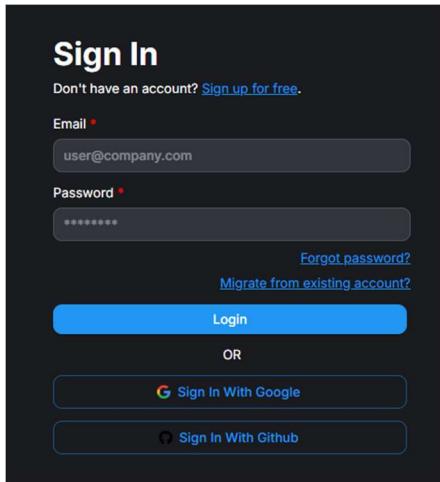
Image source: codeiste.com

1. Flowise Website

Goto this Website : <https://cloud.flowiseai.com/signin>

Sign up for free->complete signup of a free Account

Sign In with the new account created



2. Flowise Development Interface

A screenshot of the Flowise development interface. The top navigation bar shows the project name "nyprojectwork's Organization / Default Workspace" and includes "Star 42,696", "Upgrade", and settings icons. The main area is titled "Chatflows" with the sub-instruction "Build single-agent systems, chatbots and simple LLM flows". On the right, there is a search bar "Search Name or Category [Ctr]", a "No Chatflows Yet" message, and a "+ Add New" button. The left sidebar contains a navigation menu with the following items: Chatflows (selected), Agentflows, Executions, Assistants, Marketplaces, Tools, Credentials, Variables, API Keys, Document Stores, Evaluations, Datasets, Evaluators, and Evaluations. The "Chatflows" item has a small icon of two stylized characters.

In Flowise v2 (and above), there are two main paradigms for developing LLM applications: **Chatflow** and **Agentflow**. Both are built on top of LangChain and provide drag-and-drop “low-code” design for AI apps, but they differ in scope and complexity.

a). Flowise: Chatflow

Definition:

A **Chatflow** is a structured conversation pipeline where an LLM is connected to a chain of predefined components (memory, tools, RAG, prompts, etc.) to answer user queries in a controlled way.

Key Features:

- Designed mainly for **conversational AI** (chatbots, FAQ bots, RAG Q&A systems).
- Typically **linear or branching flows** of operations.
- Uses **Conversation Chains** (like ConversationChain, ConversationalRetrievalQA, etc.).
- Good for **Q&A, summarization, translation, document-based chat, customer support bots**.
- Supports **memory** (short-term or long-term) so the chatbot can remember past messages.
- Less autonomous: the flow is deterministic—user input passes through a fixed pipeline.

Example Use Cases:

- Customer support chatbot for FAQs.
- Internal knowledge base Q&A system (with RAG).
- Chat assistant with memory.

b). Flowise: Agentflow

Definition:

An **Agentflow** is a more advanced setup where an LLM is given the ability to **reason and act autonomously** by deciding which tools, APIs, or data sources to call, based on the user's query.

Key Features:

- Built around **Agents** (like LangChain ReAct agent, OpenAI function calling, etc.).
- The LLM can **choose dynamically** which tool to use at each step.
- Tools can include:
 - Web search
 - Database queries
 - APIs (weather, stocks, CRMs, etc.)
 - Custom functions
- More flexible but requires more configuration (prompting, tool definitions, safety rules).
- Suitable for **multi-step reasoning tasks** or where the LLM must decide the workflow dynamically.

Example Use Cases:

- AI assistant that can query multiple APIs (calendar, weather, finance).
 - Customer service agent that chooses whether to fetch FAQ docs, escalate to CRM, or log a ticket.
 - Research assistant that can do web search + summarization + analysis.
-

Summary Difference

Feature	Chatflow	Agentflow
Structure	Predefined pipeline (deterministic)	Dynamic reasoning (autonomous)
Complexity	Simpler, controlled	More complex, flexible
Memory	Yes	Yes
Tools	Limited, chained in flow	Multiple, LLM chooses dynamically
Best for	Q&A, chatbots, RAG systems	Multi-tool assistants, autonomous reasoning
Example	FAQ bot for HR policies	AI assistant for scheduling + search + reports

Summary:

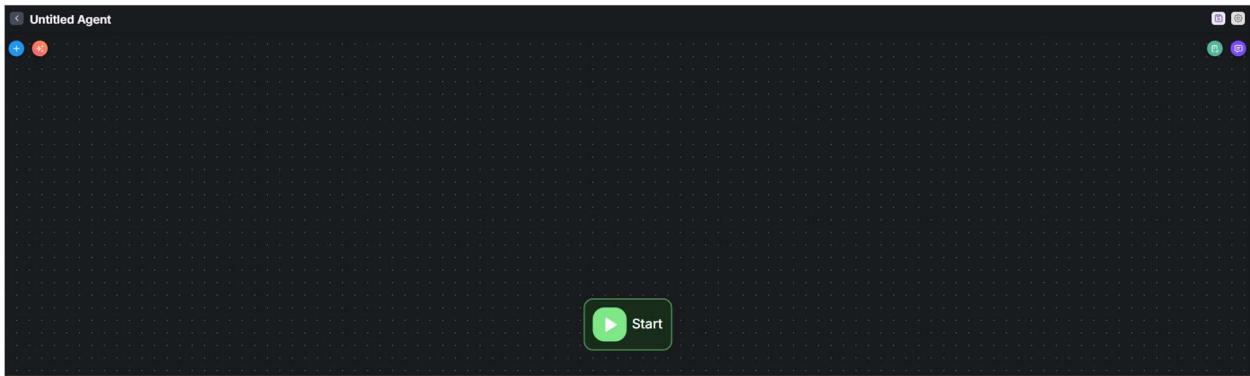
- **Use Chatflow** when you want a **guided chatbot with predictable behavior**.
- **Use Agentflow** when you want an **AI assistant that decides how to solve problems using multiple tools and reasoning**.

Ex – Multiagent HR System

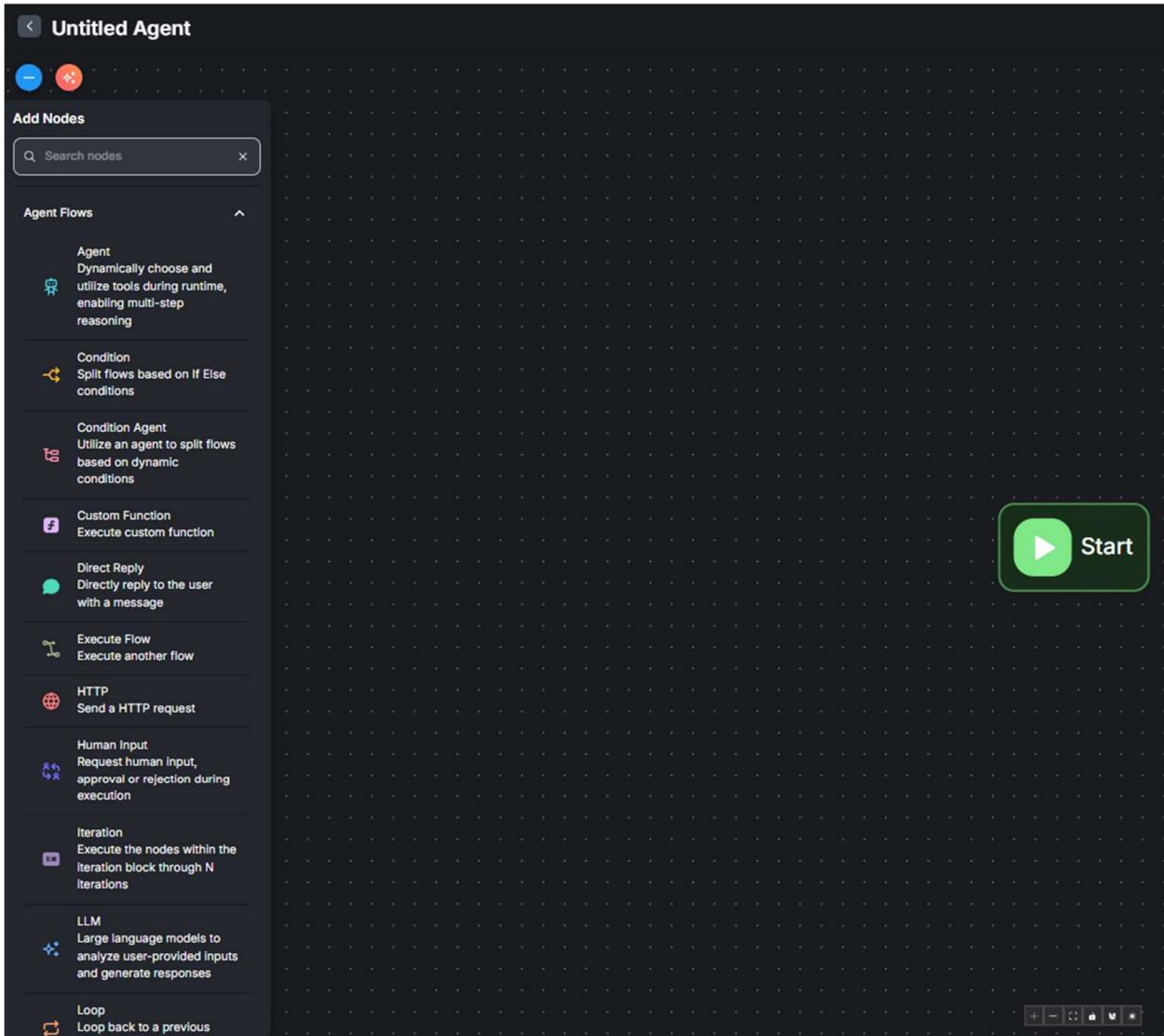
1. Agentflows

Select Agentflows - > **+Add New**

You will see an empty canvas with Untitled Chatflow



Navigate to the Blue + sign and Click on it



There is some Agent Flows provided

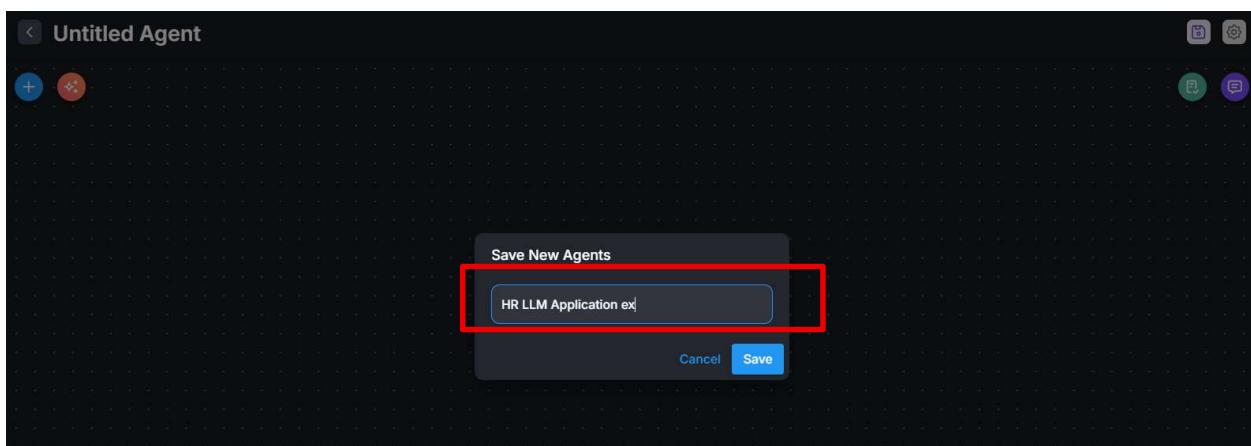
AgentFlow V2 is the architectural model inside Flowise for building **agentic workflows** — i.e., agents that operate in a structured workflow, sharing a runtime state, branching, looping, human-in-the-loop, etc.

Key features include:

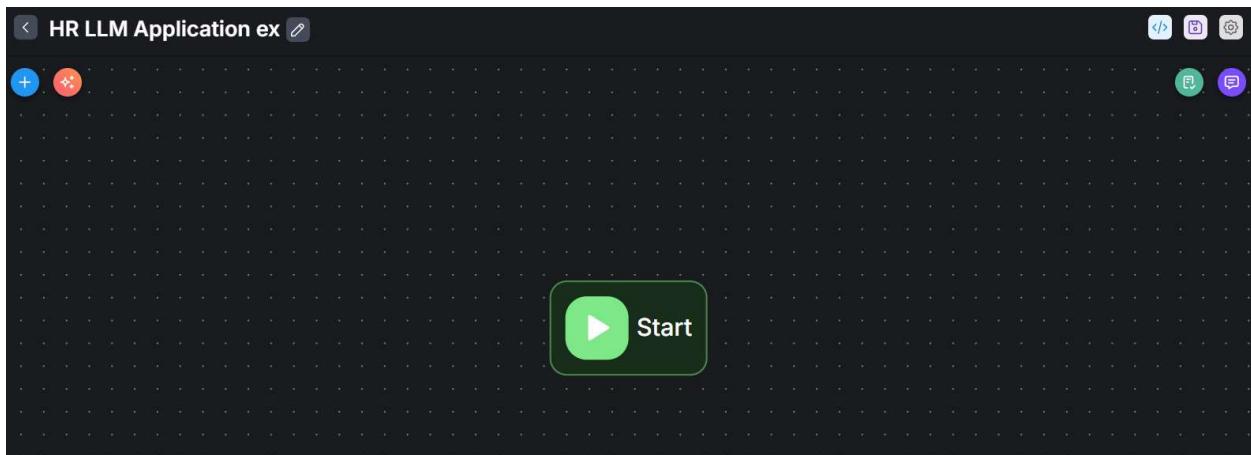
- Nodes are treated as **stand-alone components** each performing a discrete operation.
- A **Flow State** (`$flow.state`) is shared across nodes within a run, enabling data passing and context across non-adjacent nodes.
- Support for loops, conditionals, human approvals, multi-agent orchestration, and tool use.

2. Build an HR Multiagent application

Select Save Agent(at right upper corner). Save as **HR LLM Application ex.**



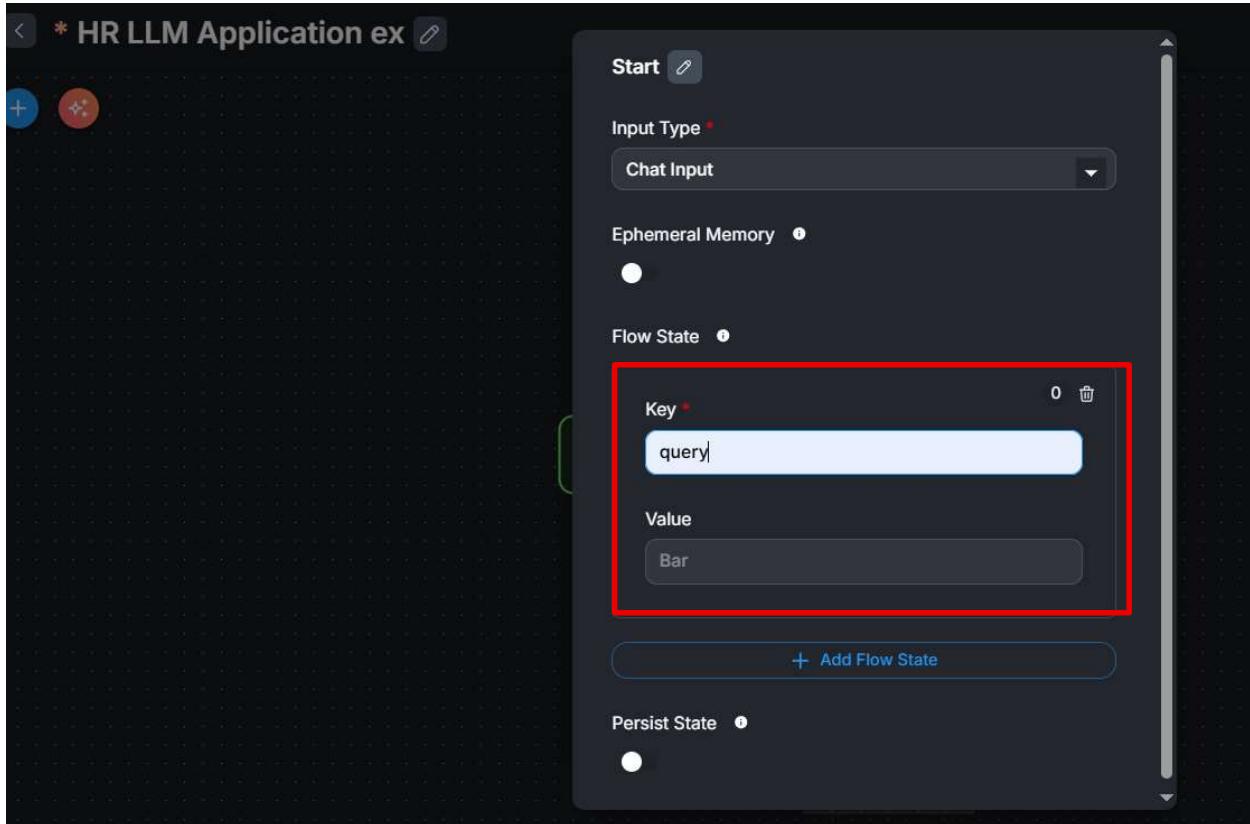
The file name is updated.



At Start Node, double click to access the dialogbox for entry.

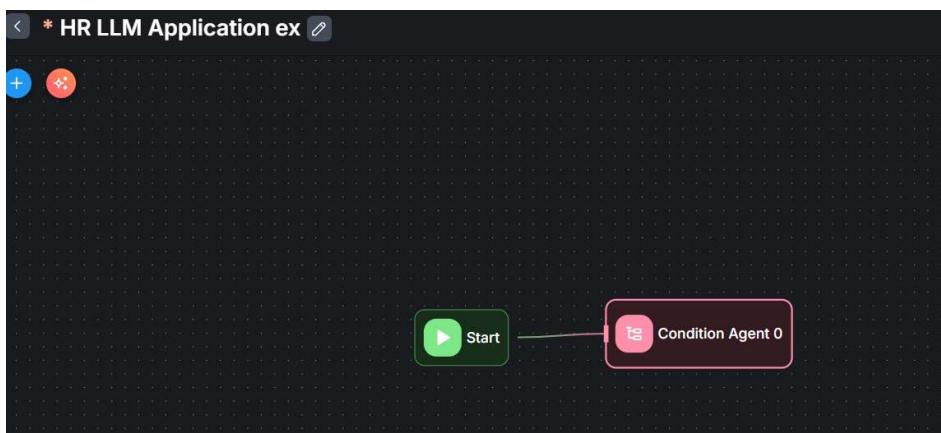
At flow state, Add a Key : query with Value: empty.

The query will store the user question to be pass down the flow.

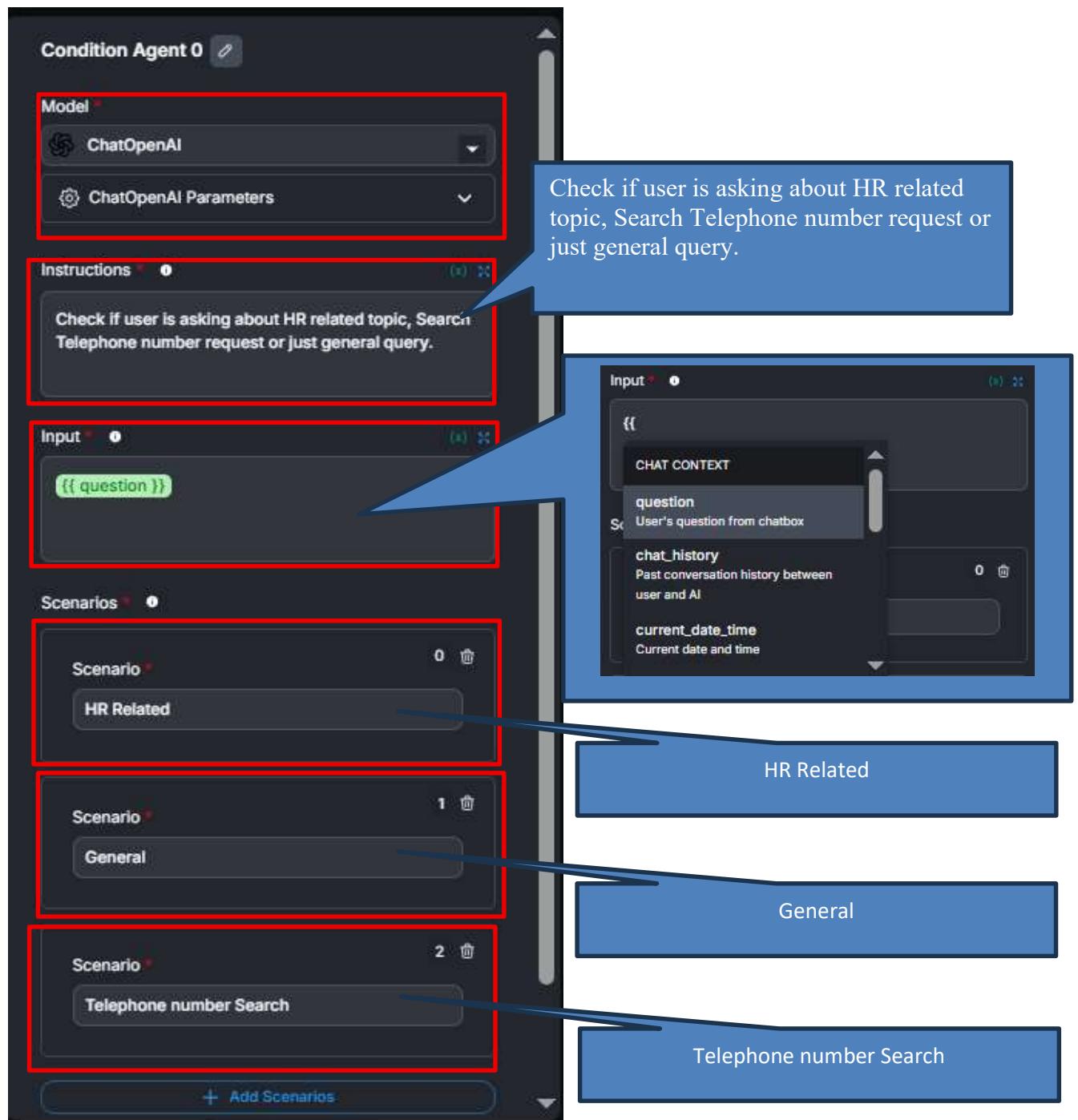


3. Decide on user query intent

Add Node Condition Agent, connect it to start node.



Double click on the condition Agent 0 node, enter the following information to configure it.



There are 3 intents where the LLM application can branch to agent.

- Agent RAG for all query regarding HR policy
- Agent for all query regarding searching for staff telephone numbers(Use tool to interface with mockup telephone number search system developed using Javascripts(js)).
- General for all query regarding information beside the above 2. Will use the LLM as the responding engine.

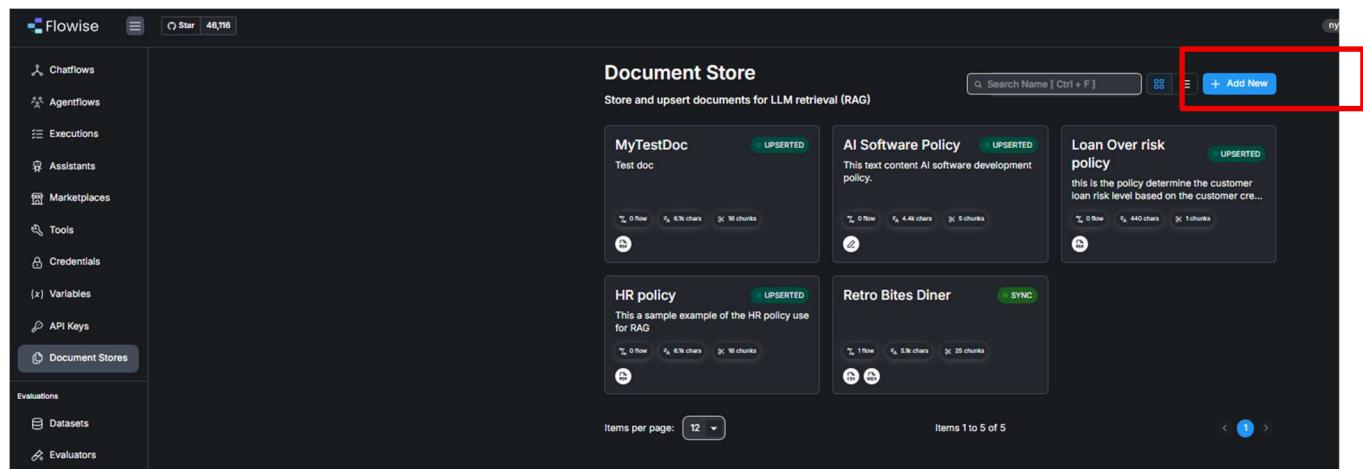
Ensure you save your work before you proceed.

4. HR Policy query with Agent RAG

4.1 Prepare the Document Store for the HR policy.

Ensure you have the API keys for Groq, Mistral and Pinecone.(refer to Appendix to setup the account)

Go Flowise Menu-> Document Stores. Click button **+ Add New**



Enter a name and description for the Document Store. Click Add.

Document Store

Store and upsert documents for LLM retrieval (RAG)

MyTestDoc UPSIDED	AI Software Policy UPSIDED	Loan Over risk policy UPSIDED
Test doc	This text content AI software development policy.	this is the policy determine the customer loan risk level based on the customer cre...
<small>0 flow 6.1k chars 16 chunks</small>	<small>0 flow 4.4k chars 5 chunks</small>	<small>0 flow 440 chars 1 chunks</small>

HR policy UPSIDED

This a sample example for RAG

0 flow 6.1k chars

Add New Document Store

Name *

Description

Items per page: 12

VirtuWorks HR Policy

This is the document stored the Company VirtuWorks HR policy

Add

You will see a VirtuWorks HR Policy Document store creates with Status EMPTY. Click on it to start configure the content.

Document Store

Store and upsert documents for LLM retrieval (RAG)

VirtuWorks HR Policy EMPTY	MyTestDoc UPSIDED	AI Software Policy UPSIDED
This is the document stored the Company VirtuWorks HR policy.	Test doc	This text content AI software development policy.
<small>0 flow 0 chars 0 chunks</small>	<small>0 flow 6.1k chars 16 chunks</small>	<small>0 flow 4.4k chars 5 chunks</small>
Loan Over risk policy UPSIDED	HR policy UPSIDED	Retro Bites Diner SYNC
this is the policy determine the customer loan risk level based on the customer cre...	This a sample example of the HR policy use for RAG	
<small>0 flow 440 chars 1 chunks</small>	<small>0 flow 6.1k chars 16 chunks</small>	<small>1 flow 5.3k chars 25 chunks</small>

Items per page: 12

Items 1 to 6 of 6

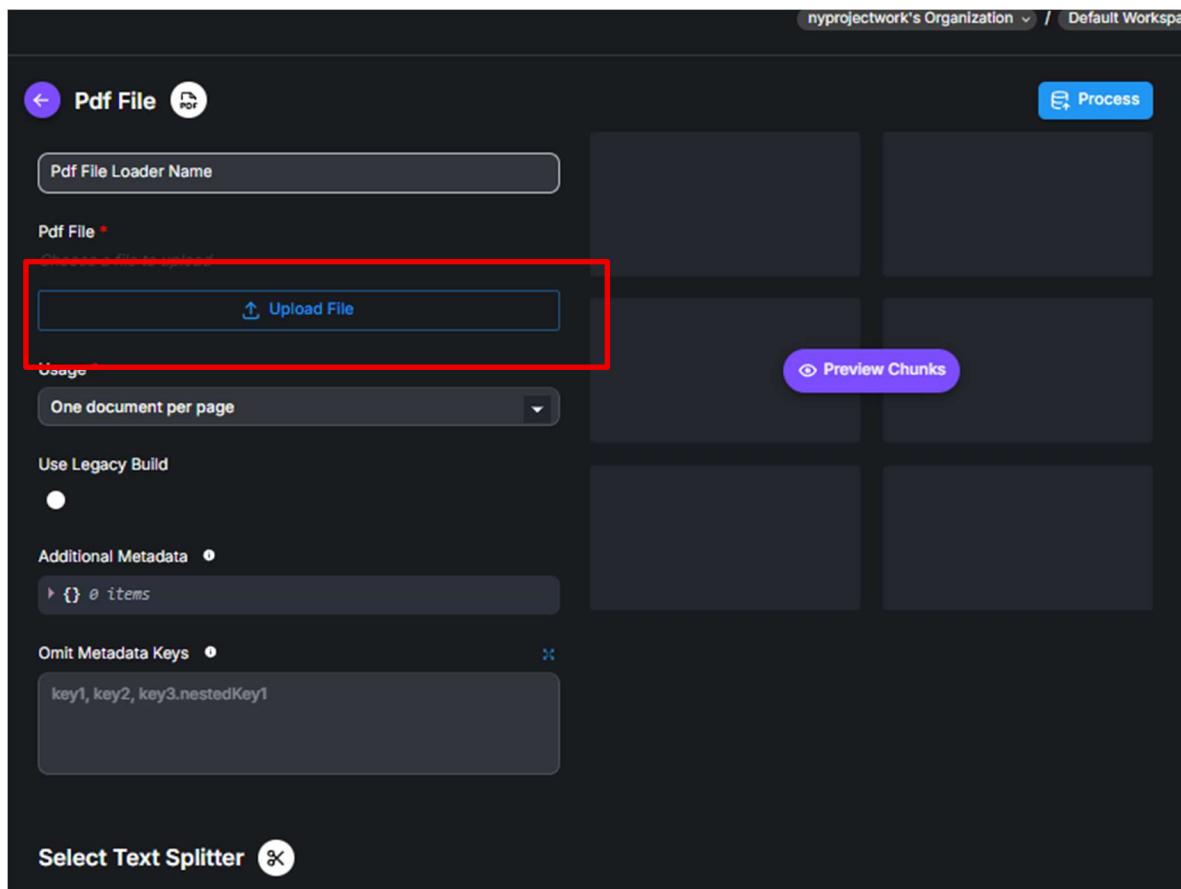
Select Add Document Loader.

The screenshot shows a dark-themed interface for managing a document titled "VirtuWorks HR Policy". At the top right, there are buttons for "+ Add Document Loader" and "More Actions". Below the title, a message states "This is the document stored the Company VirtuWorks HR policy." A small icon indicates the document is "EMPTY". In the center, there is a placeholder image of a person interacting with documents, with the text "No Document Added Yet" below it. A blue button labeled "+ Add Document Loader" is located at the bottom of this section, which is highlighted with a red rectangle.

Select Pdf File

The screenshot shows a modal window titled "Select Document Loader". It contains a search bar and a grid of 18 document loader options. The "Pdf File" option, which has a PDF icon and the text "Pdf File", is highlighted with a red rectangle. Other options include "Json File", "Json Lines File", "Microsoft Excel", "Microsoft PowerPoint", "Microsoft Word", "Notion Database", "Notion Page", "Oxylabs", "Plain Text", "Playwright Web Scraper", "Puppeteer Web Scraper", "S3 Directory", "S3", "SearchApi For Web Search", "SerpApi For Web Search", "Spider Document Loaders", "Text File", and "Unstructured File Loader".

Click **Upload Files**. Select **VirtuWorks_Expanded_HR_Policy_SG.pdf** provide to you.



Enter the information as in the following . Then click **Preview Chunks**. Lastly click **Process**.

Pdf File 7% PDF

Pdf File Loader Name: VirtuWork HR Policy PDF

Pdf File *: VirtuWorks_Expanded_HR_Policy_SG.pdf

Upload File

Usage *: One document per page

Use Legacy Build

Additional Metadata *: (0) 0 items

Omit Metadata Keys *: key1, key2, key3.nestedKey1

Character Text Splitter 1%

Splitter: Character Text Splitter

Chunk Size *: 500

Chunk Overlap *: 200

Custom Separator *: --

Check the green button status for complete processing.

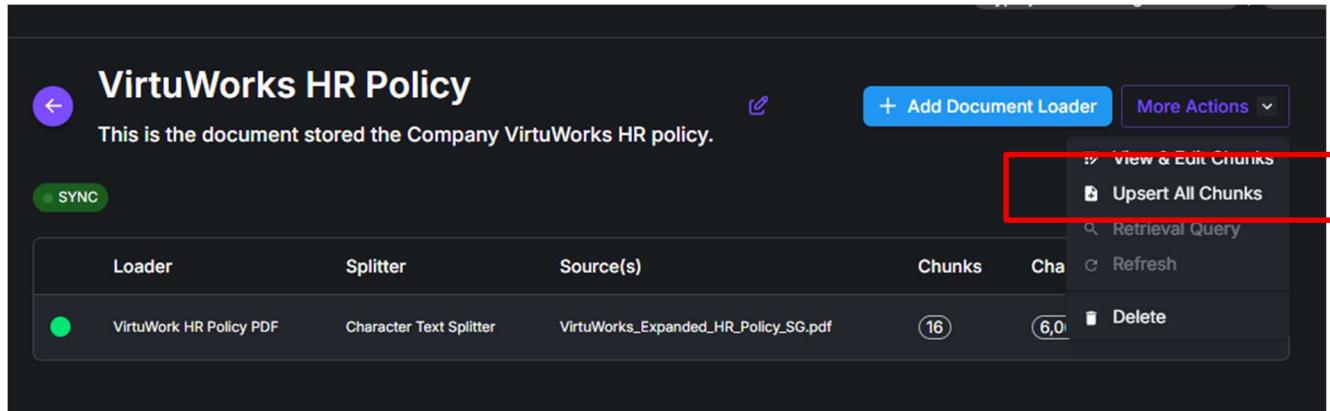
VirtuWorks HR Policy

This is the document stored the Company VirtuWorks HR policy.

+ Add Document Loader More Actions

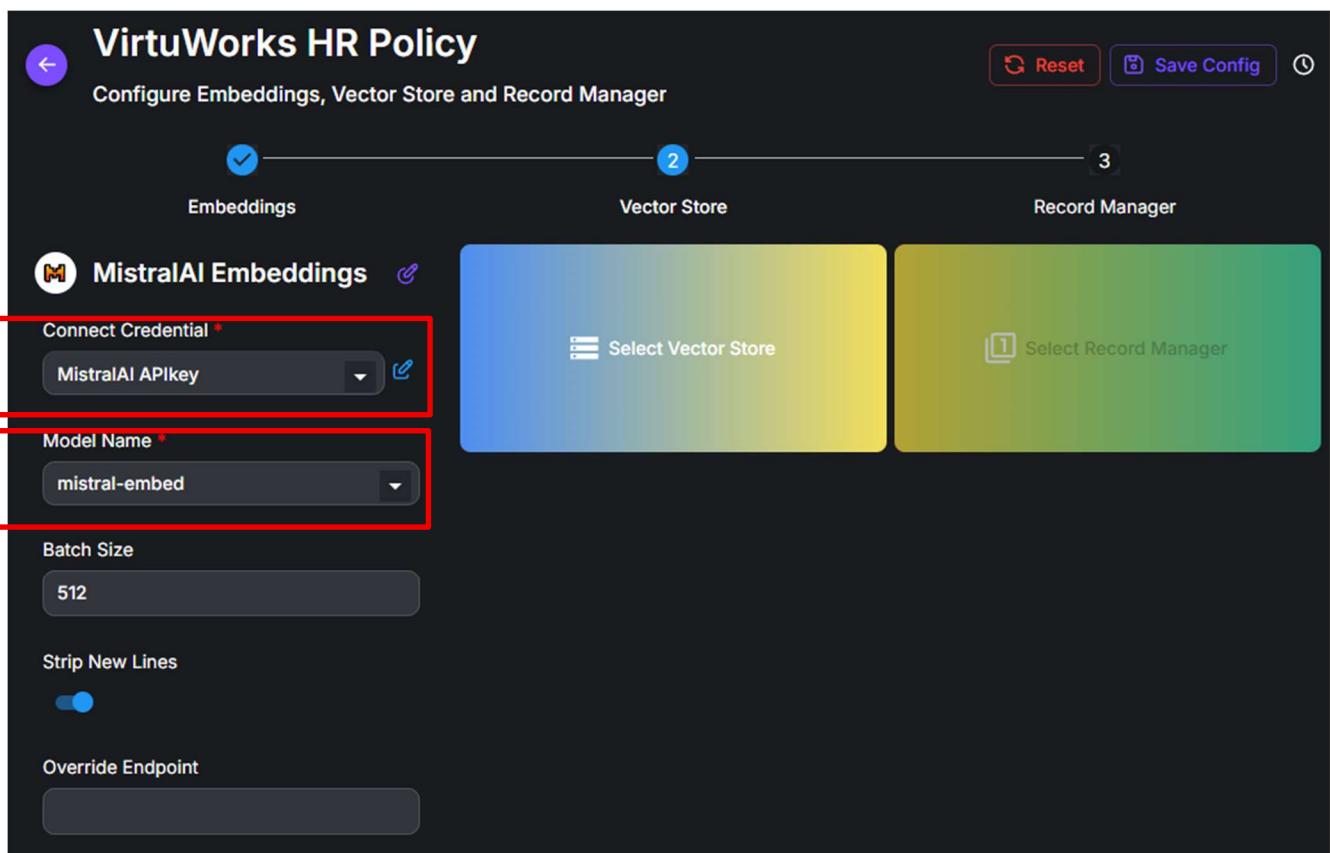
Loader	Splitter	Source(s)	Chunks	Chars	Actions
VirtuWork HR Policy PDF	Character Text Splitter	VirtuWorks_Expanded_HR_Policy_SG.pdf	16	6,064	Options

Select More Actions->Upsert All Chunks



The screenshot shows the VirtuWorks HR Policy document interface. At the top right, there is a 'More Actions' dropdown menu. One of the options, 'Upsert All Chunks', is highlighted with a red box.

Select Embeddings->MistralAI Embeddings. Complete the following configuration.



The screenshot shows the MistralAI Embeddings configuration page. It has three main sections: Embeddings, Vector Store, and Record Manager. The Embeddings section is active, showing fields for 'Connect Credential' (set to 'MistralAI APIkey') and 'Model Name' (set to 'mistral-embed'). Both of these fields are highlighted with red boxes. The Vector Store and Record Manager sections are shown as cards.

Select Vector Store -> Pinecone. Before you can setup the configuration.

You need to login to your Pinecone to setup a Index(database).

At Pinecone. Click **Create index**

The screenshot shows the Pinecone web interface. On the left, there's a sidebar with links: Get started, Database (selected), Indexes (1) (highlighted in blue), Backups, Assistant, Inference, API keys, and Manage. The main area is titled 'Indexes' and shows a single index named 'mytestdb'. Below the index name, it says 'Host: https://mytestdb-0c0a9c6.svc.aped-4627-b74a.pinecone.io' and 'Cloud: AWS'. There are buttons for 'Connect' and '...'. At the top right, there are links for 'Docs', 'Settings', 'Get help', and a user icon. A red box highlights the 'Create index' button at the top right of the main area.

Create a new index

Search application / virtuworks-hr-policy

Configuration
 Custom settings


llama-text-embed-v2
NVIDIA - HOSTED
 State-of-the-art text embedding model.


multilingual-e5-large
MICROSOFT - HOSTED
 Top performing text embedding model from Microsoft research.


pinecone-sparse-eng...
PINECONE - HOSTED
 Sparse vector model for keyword-style search.

Index configured for llama-text-embed-v2:

Modality	Text
Vector type	Dense
Max input	2,048 tokens
Starter limits	5M tokens
Dimension	1024 <input type="button" value="▼"/>
Metric	cosine

Integrated embedding Now
 Let Pinecone manage the conversion to vector embeddings when you upsert and search. [Learn more](#)

Identify the field that contains the data to embed:
 Field map

Capacity mode

Serverless
 Charges based on data storage, reads, and writes.

Cloud provider







* Upgrade to Standard or Enterprise Plans for full access to all cloud providers

Region


Virginia
 us-east-1


Oregon
 us-west-2


Ireland
 eu-west-1

Request a region

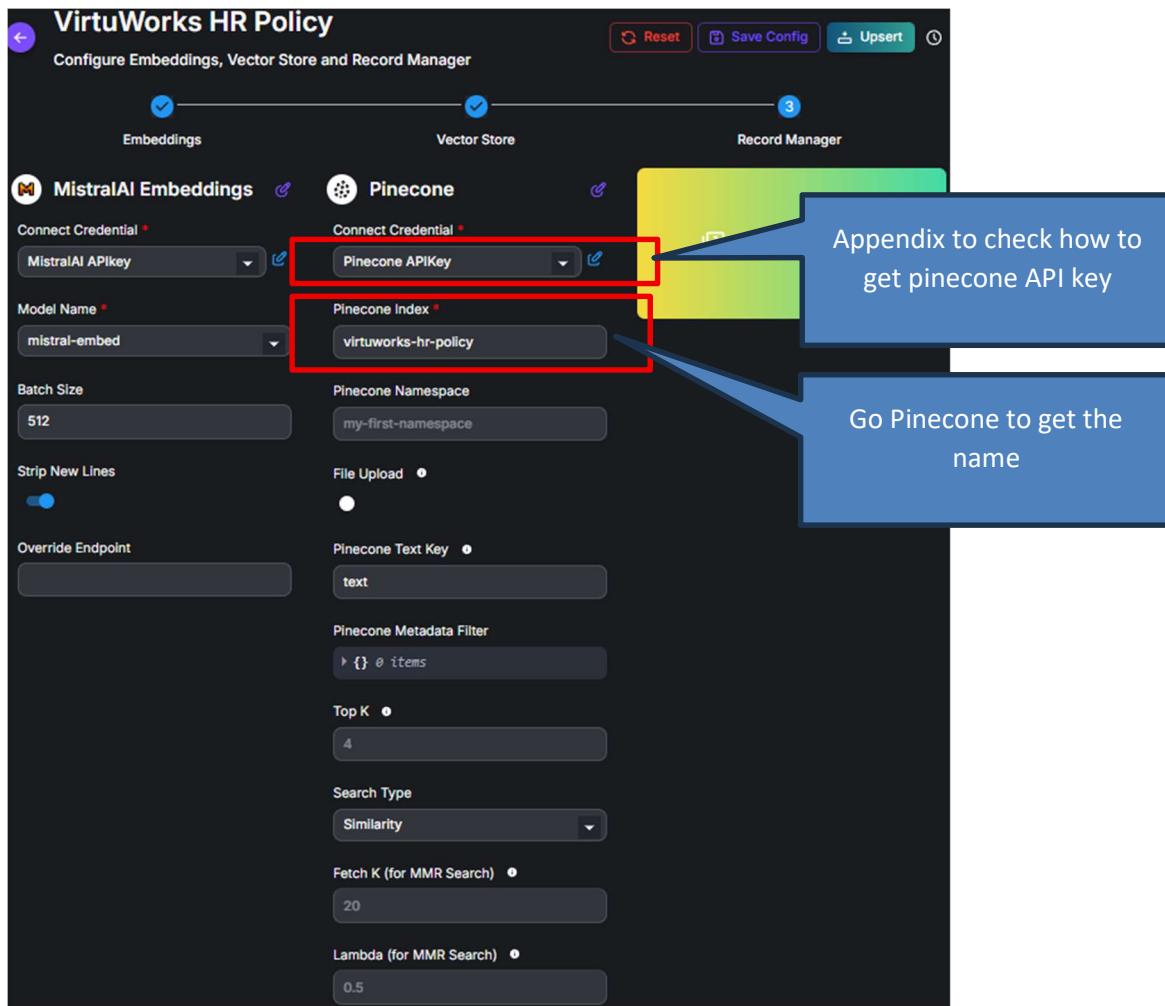
Deletion protection

When enabled, prevents any user from accidentally deleting this index. Termination protection must be disabled before deletion. [Learn more](#)

Add [tags](#) to categorize your index. You can also add tags later.

Setup the configuration as follows. Click Create index

Back to the Flowise ->Document Store-VirtuWorks HR Policy. Set the Pinecone Configuration. Click **Save Config**. Then click **Upset**.



VirtuWorks HR Policy

Configure Embeddings, Vector Store and Record Manager

Embeddings Vector Store Record Manager

MistralAI Embeddings **Pinecone**

Connect Credential * Connect Credential *

MistralAI APIkey Pinecone APIkey

Model Name * Pinecone Index *

mistral-embed virtuworks-hr-policy

Batch Size Pinecone Namespace

512 my-first-namespace

Strip New Lines File Upload

Override Endpoint

Pinecone Text Key

text

Pinecone Metadata Filter

{ } 0 items

Top K

4

Search Type

Similarity

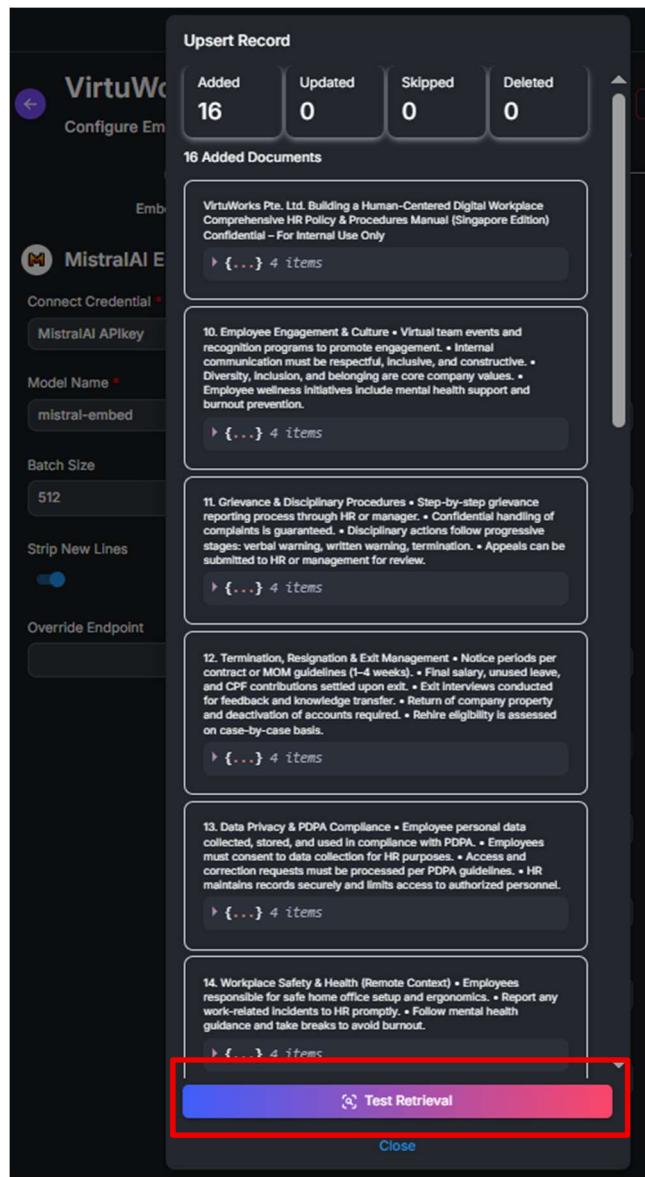
Fetch K (for MMR Search)

20

Lambda (for MMR Search)

0.5

You will see this if the data are uploaded to Pinecone successfully. Proceed to test the retrieval.



Enter this question : What is my working hours?.

The screenshot shows the VirtuWorks HR Policy retrieval playground interface. A red box highlights the search bar where the query "What is my working hours?" has been entered. Another red box highlights the "Retrieved Documents" section, which displays four results:

- #5. Characters: 428
4. Working Hours & Flexibility
 - Flexible work schedules with core collaboration hours 10:00 AM to 3:00 PM SGT.
 - Employees are expected to manage schedules responsibly and attend
- #7. Characters: 622
6. Leave & Time-Off Policy (MOM Compliant)
 - Annual leave: Minimum 7 days after first year, increasing with service length.
 - Sick leave: 14 days per year; hospitalisation leave up to 60 days with
- #15. Characters: 231
14. Workplace Safety & Health (Remote Context)
 - Employees responsible for safe home office setup and ergonomics.
 - Report any work-related incidents to HR promptly
- #3. Characters: 375
2. Employment Structure
 - Employment types: permanent, part-time, contractual, freelance.
 - Probation period: typically 3–6 months with confirmation upon satisfactory performance

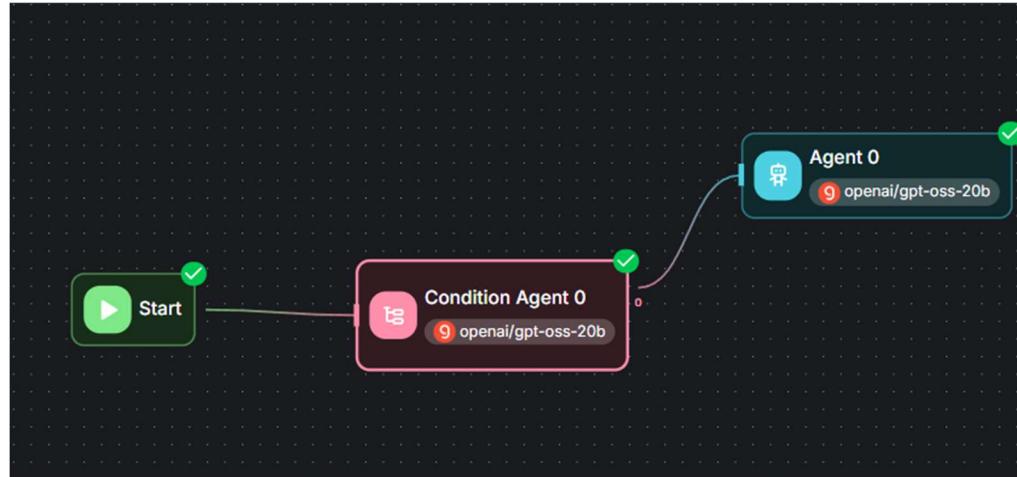
Back to document store. You will see VirtuWorks HR Policy status UPSERTED

The screenshot shows a dark-themed 'Document Store' interface. At the top right are search, filter, and 'Add New' buttons. Below is a list of documents:

- VirtuWorks HR Policy** (Status: UPSERTED): This is the document stored the Company VirtuWorks HR policy. It has 0 flow, 6.1k chars, and 16 chunks.
- MyTestDoc** (Status: UPSERTED): Test doc. It has 0 flow, 6.1k chars, and 16 chunks.
- AI Software Policy** (Status: UPSERTED): This text content AI software development policy. It has 0 flow, 4.4k chars, and 5 chunks.
- Loan Over risk policy** (Status: UPSERTED): This is the policy determine the customer loan risk level based on the customer cre... It has 0 flow, 440 chars, and 1 chunks.
- HR policy** (Status: UPSERTED): This a sample example of the HR policy use for RAG. It has 0 flow, 6.1k chars, and 16 chunks.
- Retro Bites Diner** (Status: SYNC): It has 1 flow, 5.1k chars, and 25 chunks.

4.2 Create the RAG Agent.

At the Node drag the Agent into canvas to connect to Node Condition Agent 0 at index 0. Then configure the RAG Agent as follows



Provide clear and concise answers to HR policy queries based on a knowledge database. The aim is to assist employees by providing accurate and helpful information regarding HR policies. You should ensure that responses are precise, relevant, and catered to the specific questions asked.

Steps

1. Understand the employee's query regarding HR policies.
2. Retrieve the relevant information from the knowledge database that corresponds to the query.
3. Present the information in a straightforward manner, ensuring clarity and accuracy.

Output Format

The output should be a short paragraph that directly addresses the query, ideally between 2-5 sentences. Ensure that it is easy to understand and free from jargon.

Examples

Example 1:

- **Input:** What is the process for requesting leave?
- **Output:** Employees must submit a leave request form to their supervisor at least two weeks prior to the intended leave date. The supervisor will review the request and inform the employee of the decision within five business days.

Example 2:

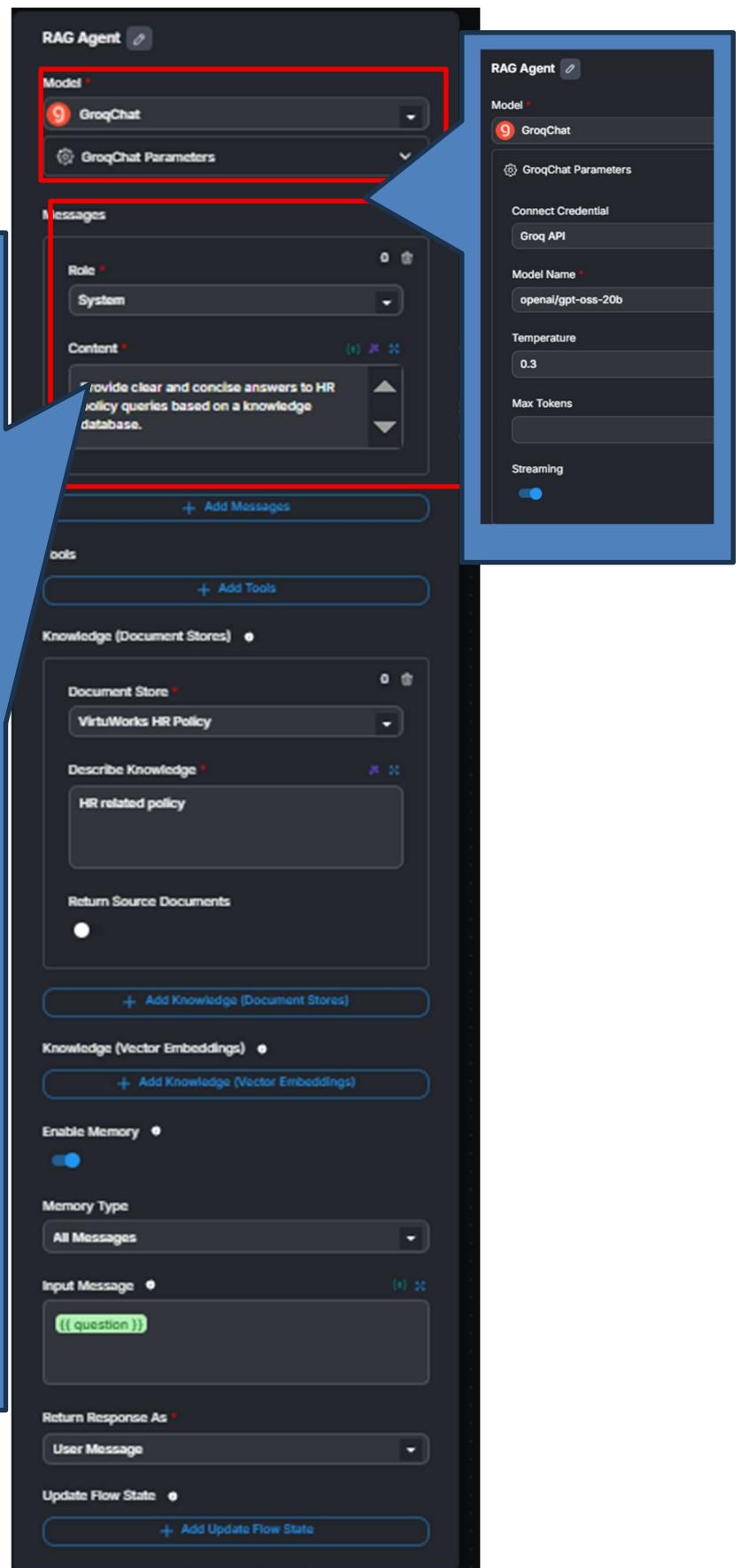
- **Input:** What are the benefits provided to part-time employees?
- **Output:** Part-time employees are eligible for a prorated amount of health insurance benefits, as well as access to the employee assistance program. Additional benefits may vary based on hours worked.

Example 3:

- **Input:** How do I report workplace harassment?
- **Output:** To report workplace harassment, employees should contact HR directly via email or phone. A confidential meeting will be scheduled to discuss the issue and initiate an investigation.

Notes

- Ensure to prioritize the employee's specific concerns when providing responses.
- If the query is unclear, ask follow-up questions to clarify before providing an answer.



At the Messages->Role: System-> Content:

Provide clear and concise answers to HR policy queries based on a knowledge database.

The aim is to assist employees by providing accurate and helpful information regarding HR policies. You should ensure that responses are precise, relevant, and catered to the specific questions asked.

Steps

1. Understand the employee's query regarding HR policies.
2. Retrieve the relevant information from the knowledge database that corresponds to the query.
3. Present the information in a straightforward manner, ensuring clarity and accuracy.

Output Format

The output should be a short paragraph that directly addresses the query, ideally between 2-5 sentences. Ensure that it is easy to understand and free from jargon.

Examples

Example 1:

- **Input:** What is the process for requesting leave?
- **Output:** Employees must submit a leave request form to their supervisor at least two weeks prior to the intended leave date. The supervisor will review the request and inform the employee of the decision within five business days.

Example 2:

- **Input:** What are the benefits provided to part-time employees?
- **Output:** Part-time employees are eligible for a prorated amount of health insurance benefits, as well as access to the employee assistance program. Additional benefits may vary based on hours worked.

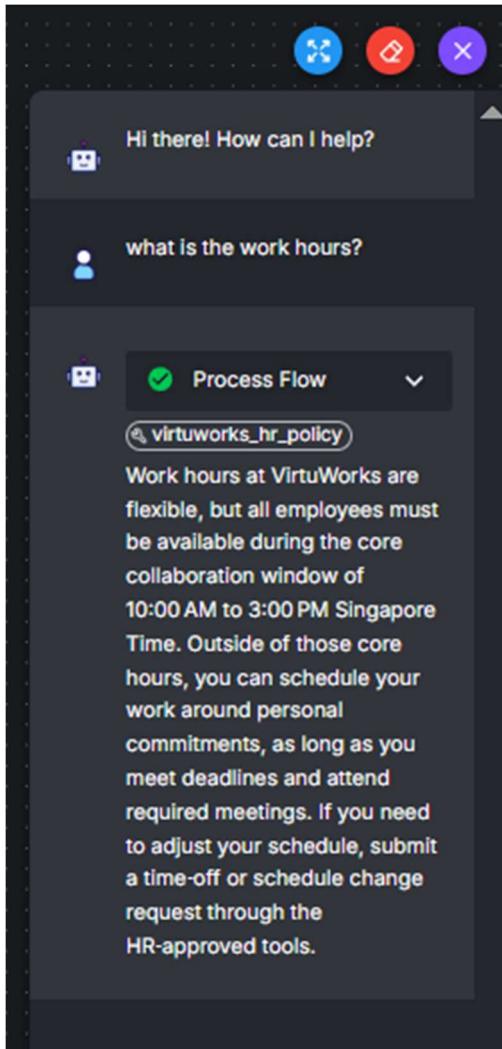
Example 3:

- **Input:** How do I report workplace harassment?
- **Output:** To report workplace harassment, employees should contact HR directly via email or phone. A confidential meeting will be scheduled to discuss the issue and initiate an investigation.

Notes

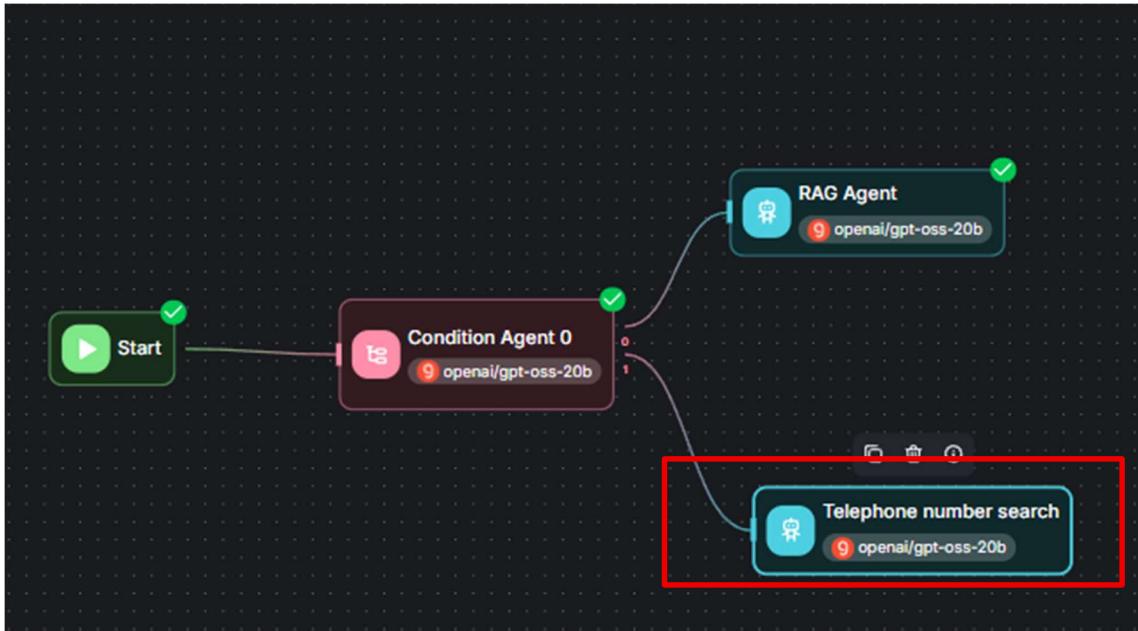
- Ensure to prioritize the employee's specific concerns when providing responses.
- If the query is unclear, ask follow-up questions to clarify before providing an answer.

Save the work. At the Web interface, test the application by entering the query.



5. HR Telephone number search Agent

At Node select Agent. Connect the RAG Agent to the Condition Agent 0 index 1.



Setup the Telephone search Tool for the agent. At the Tools, click Create

The screenshot shows the 'Tools' section of the Flowise interface. On the left is a sidebar with various navigation options like Chatflows, Agentflows, Executions, Assistants, Marketplaces, Tools (which is selected), Credentials, Variables, API Keys, Document Stores, Evaluations, and Datasets. The main area displays a list of tools:

- TelephoneSearch**: This function is to search for the staff's telephone number based on the staff's name....
- GetPRStatus**: This function is for search customer information(PR Status) based the give customer Id(it can contain email address,...)
- SearchCustomerInfoV1**: This function is for search customer information/credit score and account status) based the give customer Id(it can...
- SearchCustomerInfo**: This function is for search customer information(credit score and account status) based the give customer Id(it can...

At the top right of the tools list, there is a red box around the '+ Create' button. Below the tools, there are pagination controls: 'Items per page: 12' and 'Items 1 to 4 of 4'.

Enter the information as follows. Then click add.

Add New Tool

TOOL NAME •

TOOL DESCRIPTION •

TOOL ICON SOURCE

INPUT SCHEMA •

Property	Type	Description	Requi...
staffname	string	Staff Name	✓ <input type="checkbox"/>
1 row selected		Rows per page: 10	1-1 of 1 < >

JAVASCRIPT FUNCTION •

```

1 staffname = $staffname.toLowerCase();
2
3 if (['loren@ibm.com', 'loren', '1111'].includes(staffname)) {
4     return { phone_number: 99012345 };
5 } else if (['matt@ibm.com', 'matt', '2222'].includes(staffname)) {
6     return { phone_number: 99012366 };
7 } else if (['hilda@ibm.com', 'hilda', '3333'].includes(staffname)) {
8     return { phone_number: 99012388 };
9 } else {
10    return { phone_number: "not found" };
11 }

```

VirtuWork Telephone Search

This function is to search for the staff's telephone number based on the staff's name.
-Input Staff's name
-output Staff's telephone number

```

staffname = $staffname.toLowerCase();
if (['loren@ibm.com', 'loren',
'1111'].includes(staffname)) {

    return { phone_number: 99012345 };

} else if (['matt@ibm.com', 'matt',
'2222'].includes(staffname)) {

    return { phone_number: 99012366 };

} else if (['hilda@ibm.com', 'hilda',
'3333'].includes(staffname)) {

    return { phone_number: 99012388 };

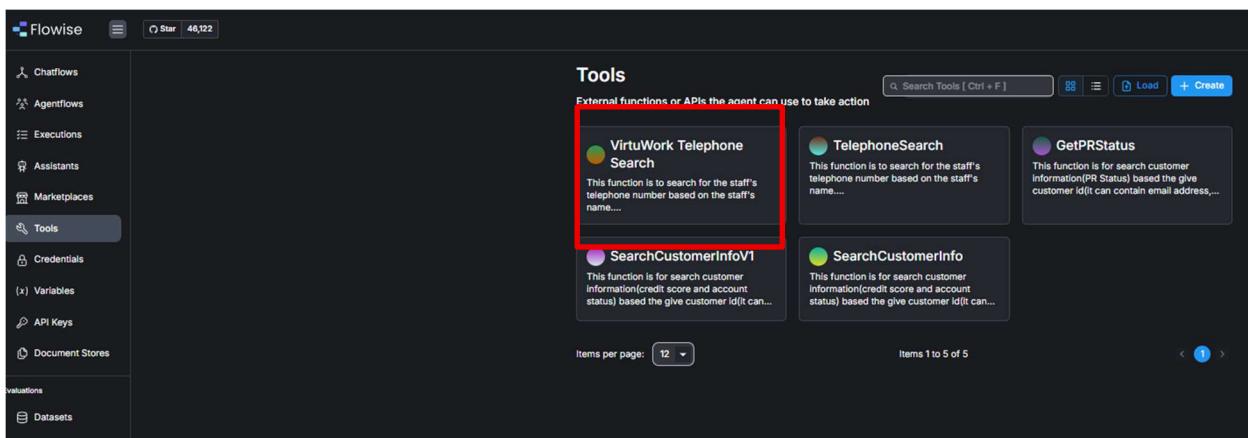
} else {

    return { phone_number: "not found" };
}

```

Page 27 of 43

Check a Tool: VirtuWork Telephone Search



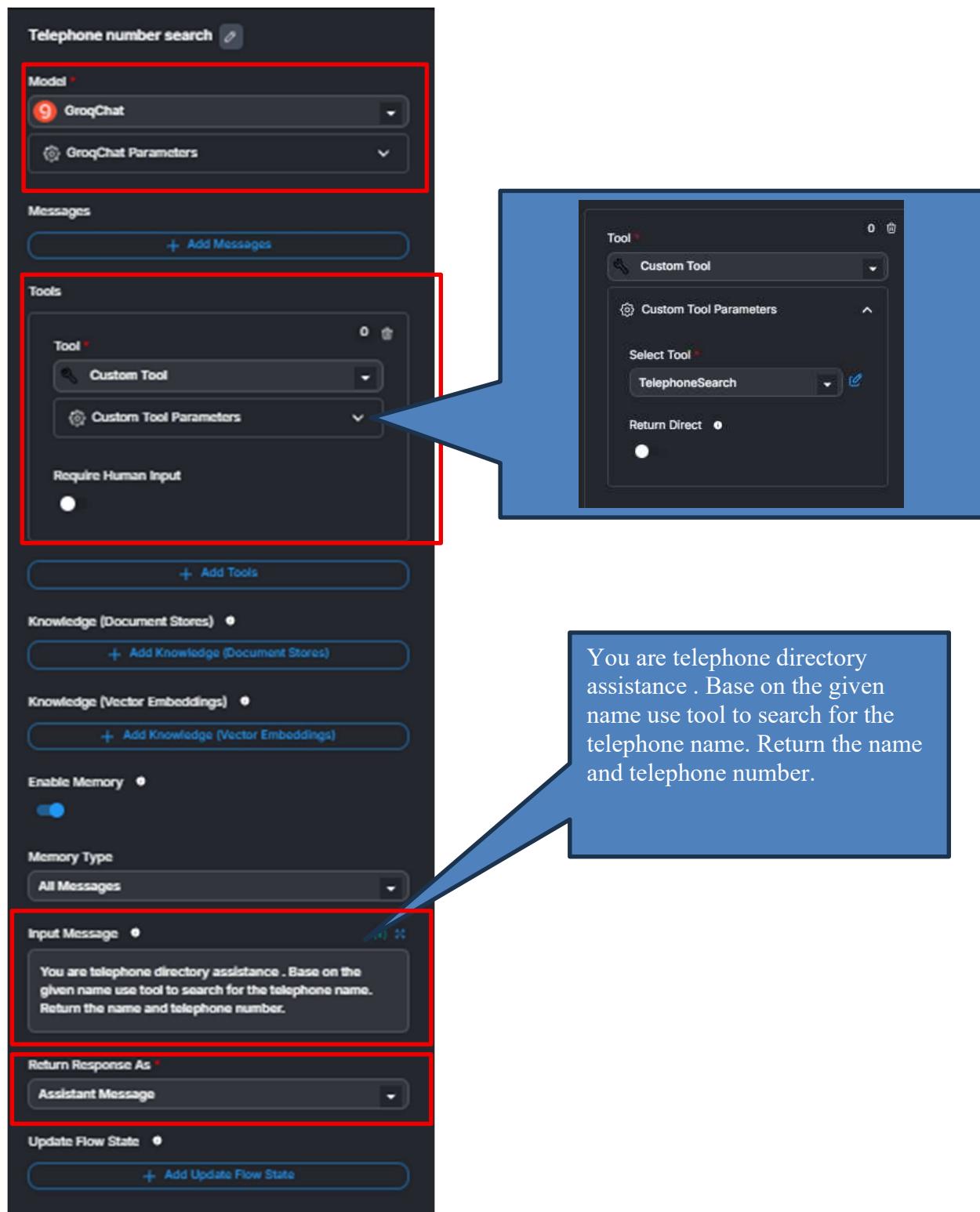
The screenshot shows the Flowwise interface with the 'Tools' section selected in the sidebar. The main area displays a list of external functions or APIs:

- VirtuWork Telephone Search**: This function is to search for the staff's telephone number based on the staff's name....
- TelephoneSearch**: This function is to search for the staff's telephone number based on the staff's name....
- GetPRStatus**: This function is for search customer information(PR Status) based the give customer id(it can contain email address,...)
- SearchCustomerInfoV1**: This function is for search customer information(credit score and account status) based the give customer id(it can...
- SearchCustomerInfo**: This function is for search customer information(credit score and account status) based the give customer id(it can...

Items per page: 12

Items 1 to 5 of 5

Back to the Telephone Search Agent. Set the following. Save the work.



Telephone number search

Model *

- GroqChat
- GroqChat Parameters

Messages

+ Add Messages

Tools

Tool *

- Custom Tool

Custom Tool Parameters

Require Human Input

+ Add Tools

Knowledge (Document Stores)

+ Add Knowledge (Document Stores)

Knowledge (Vector Embeddings)

+ Add Knowledge (Vector Embeddings)

Enable Memory

Memory Type

All Messages

Input Message

You are telephone directory assistance . Base on the given name use tool to search for the telephone name. Return the name and telephone number.

Return Response As *

Assistant Message

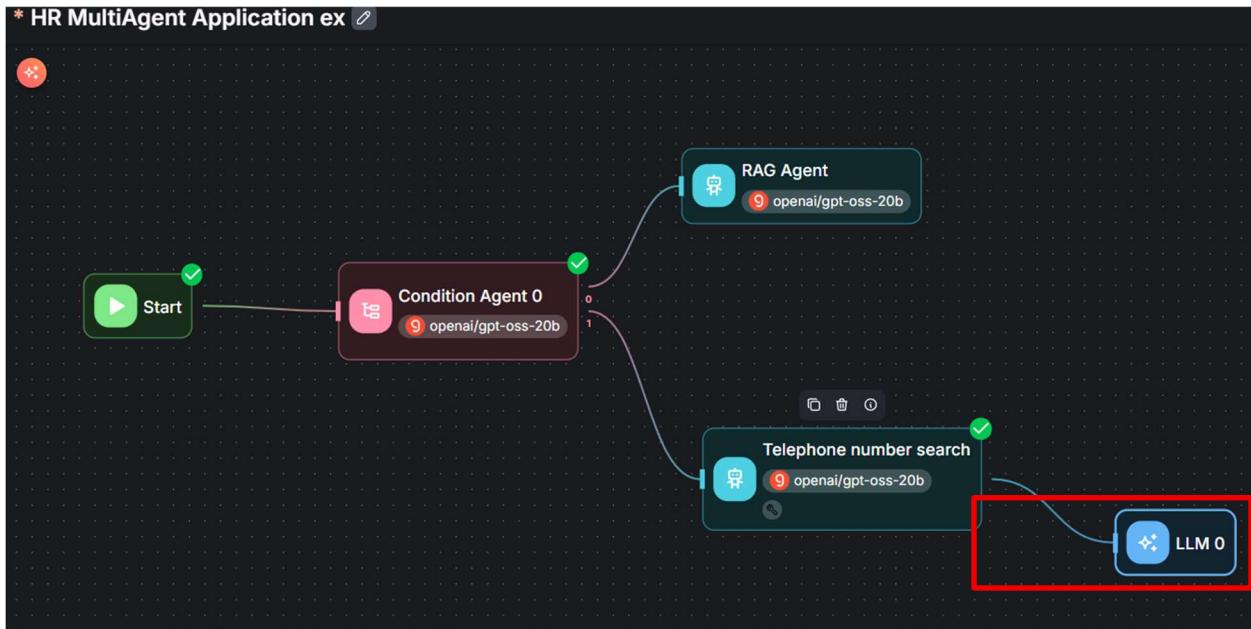
Update Flow State

+ Add Update Flow State

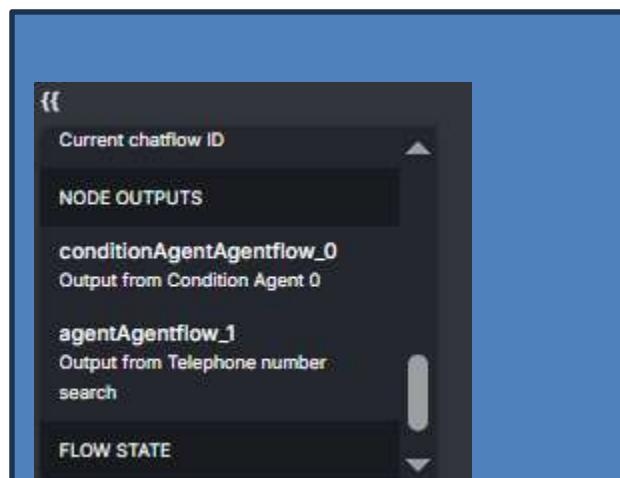
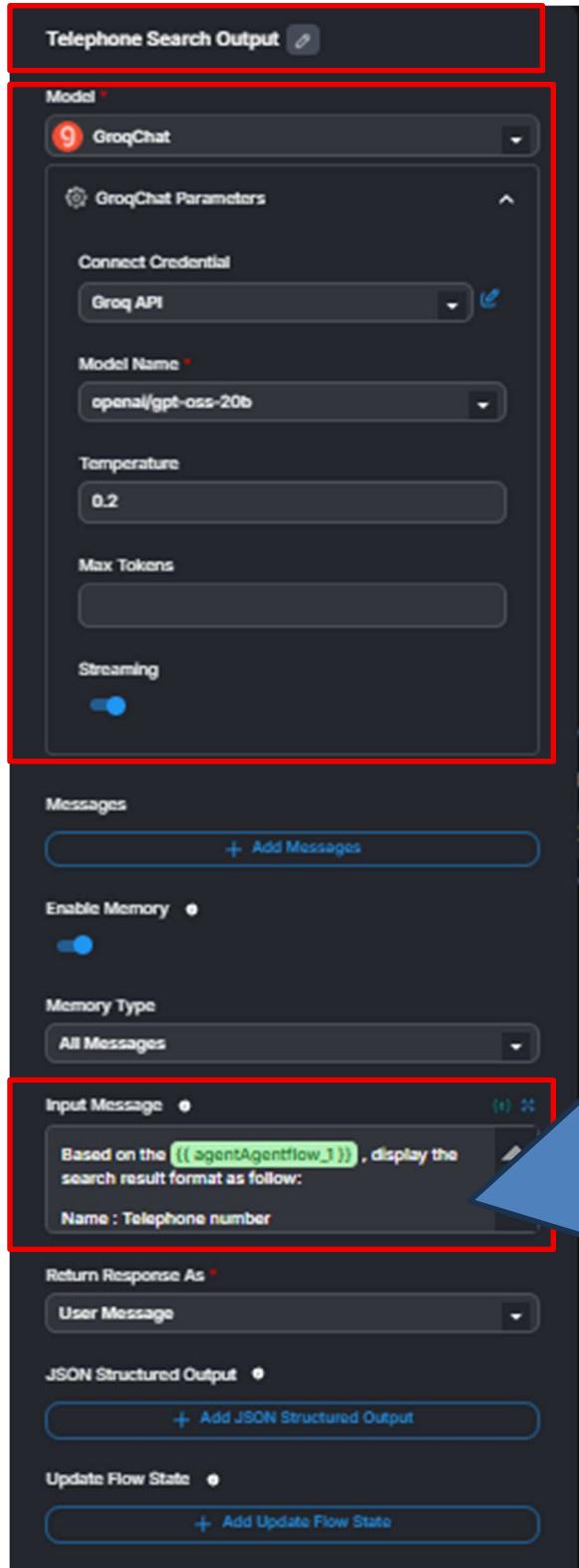
You are telephone directory assistance . Base on the given name use tool to search for the telephone name. Return the name and telephone number.

Entry:

Add a LLM Node to receive the search from telephone search agent and format the output.



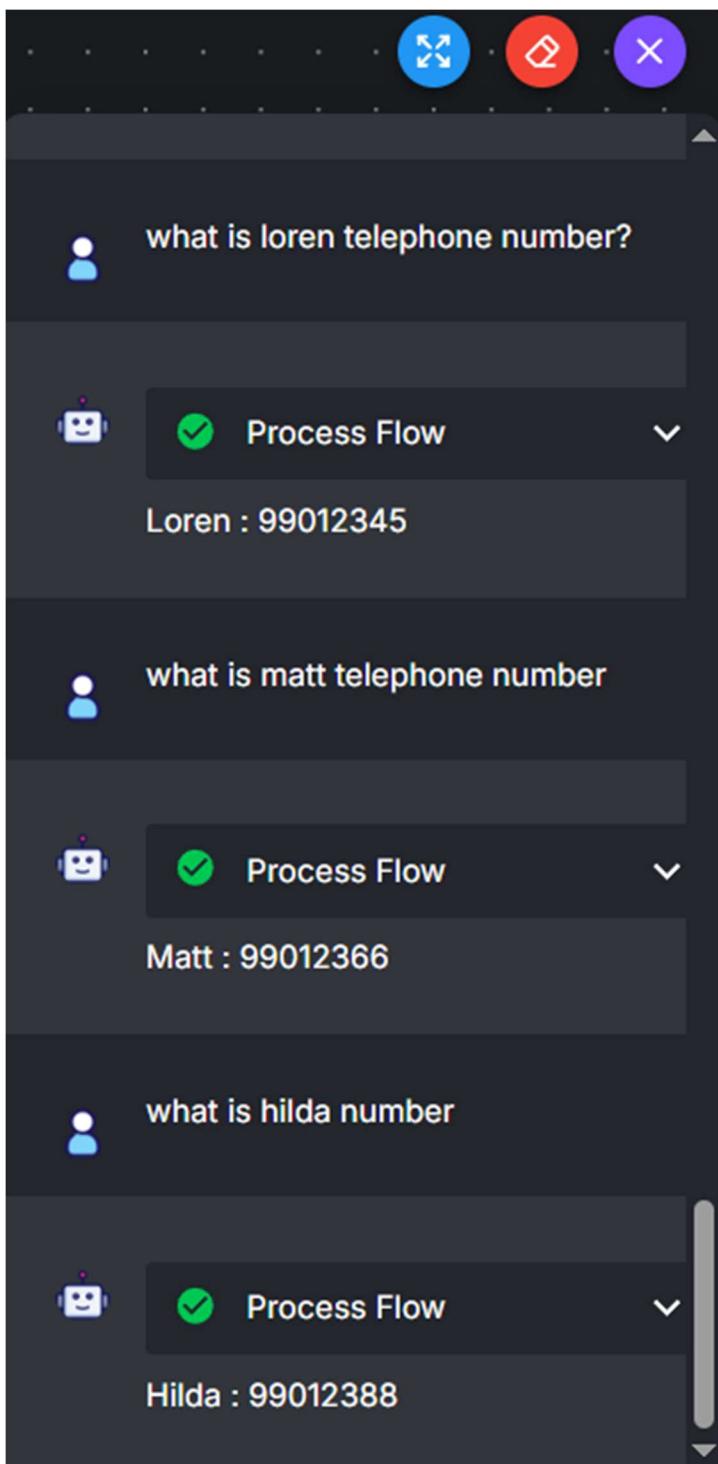
Configure the LLM as follows.



Based on the {{ agentAgentflow_1 }},
display the search result format as follow:

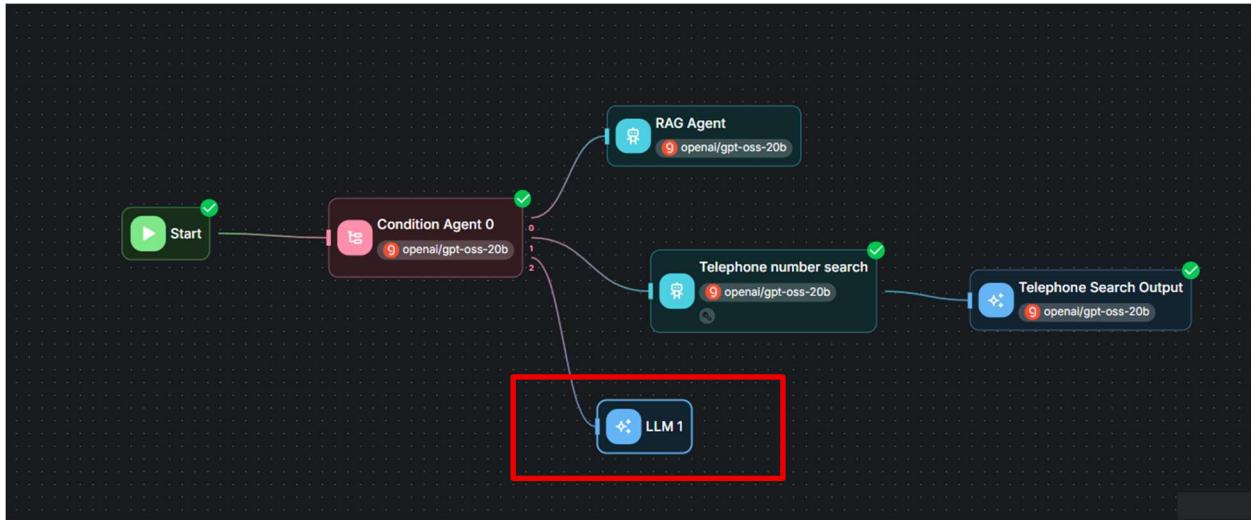
Name : Telephone number

Test the telephone search.

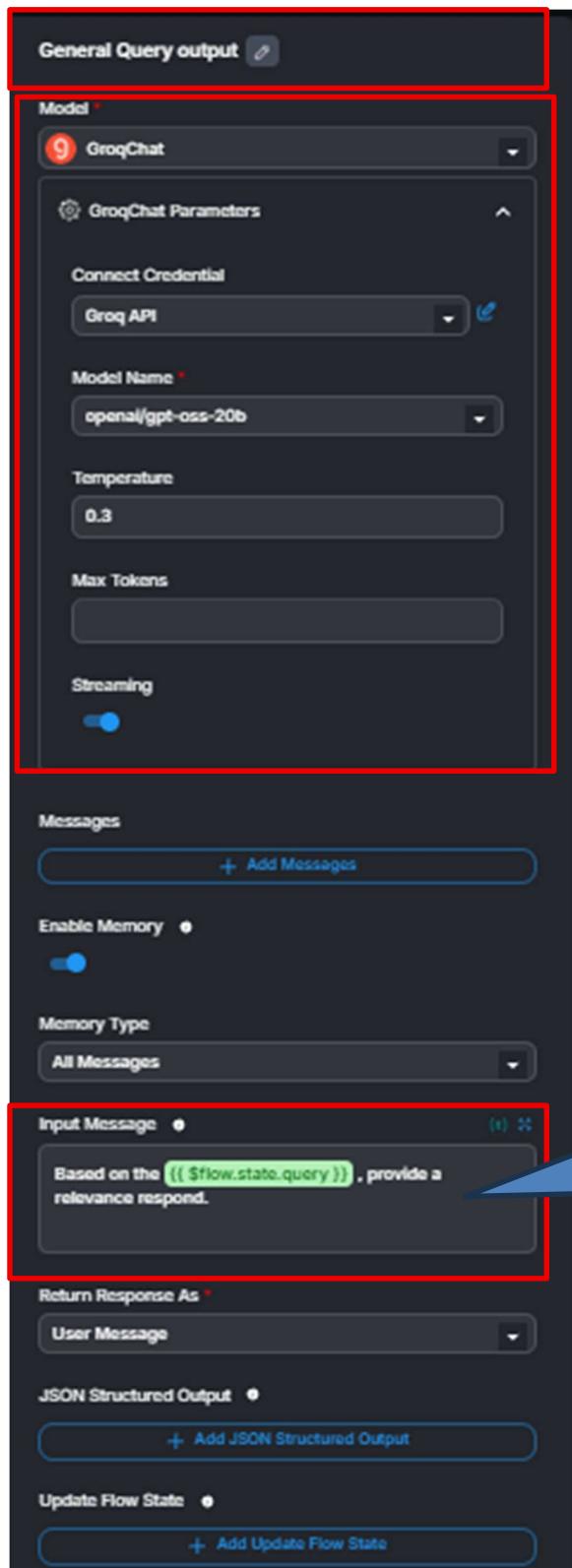


6. HR General query

Add a LLM node and connect to the Condition Agent 0 index 1.



Configure LLM as follows.



Based on the
`{{ $flow.state.query }}`, provide a
 relevance respond.

Test General Query:

What is good place for luch?

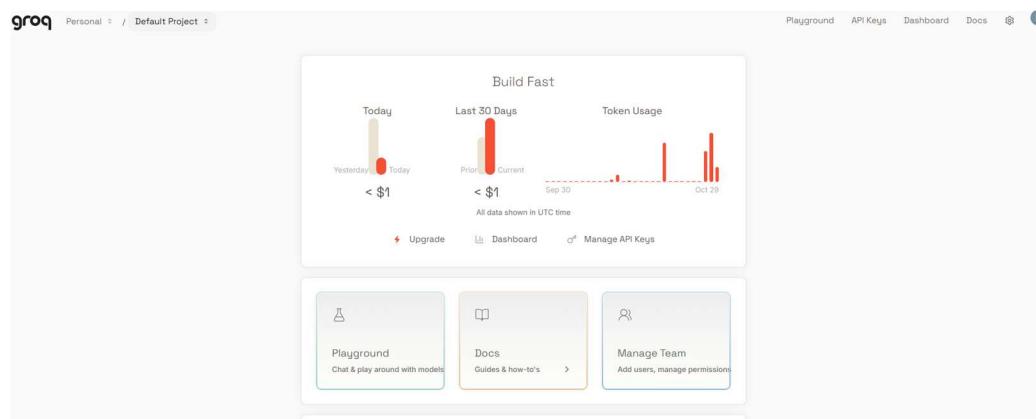
Appendix

A1. Groq-LLM API Key

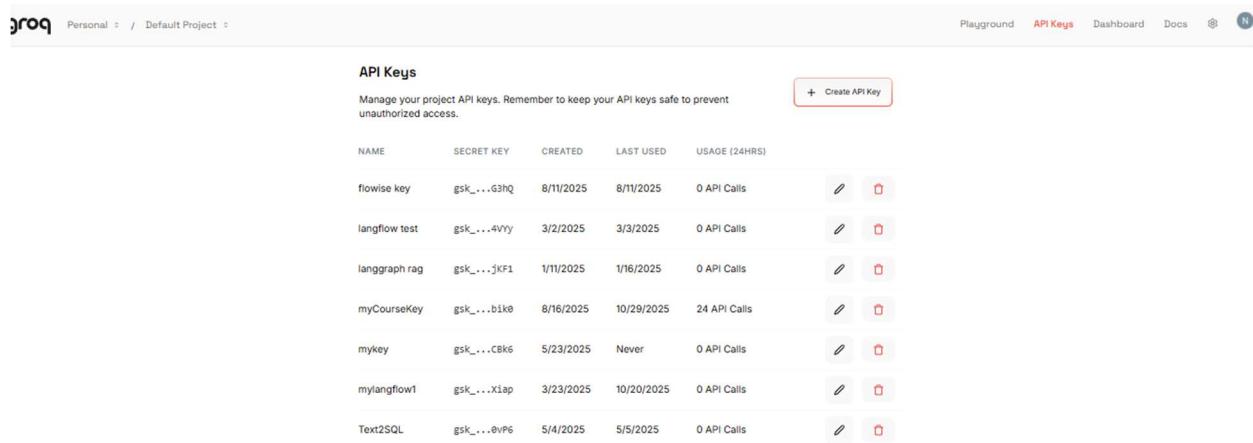
1. Use the browser goto <https://console.groq.com/login>

Register an account to create an API Key.

2. Login the groq using the account created. Select the Option on right top -> API Keys



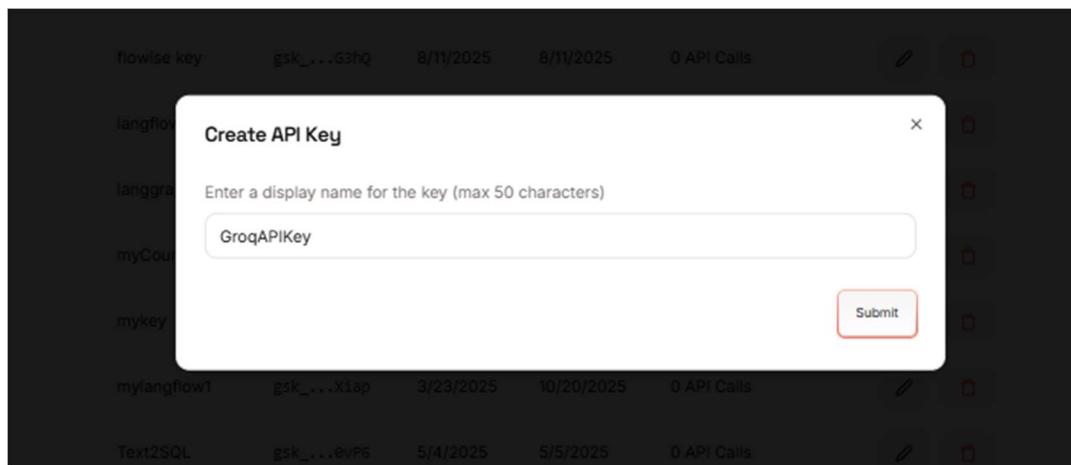
3. Select button on right -> + Create API Key



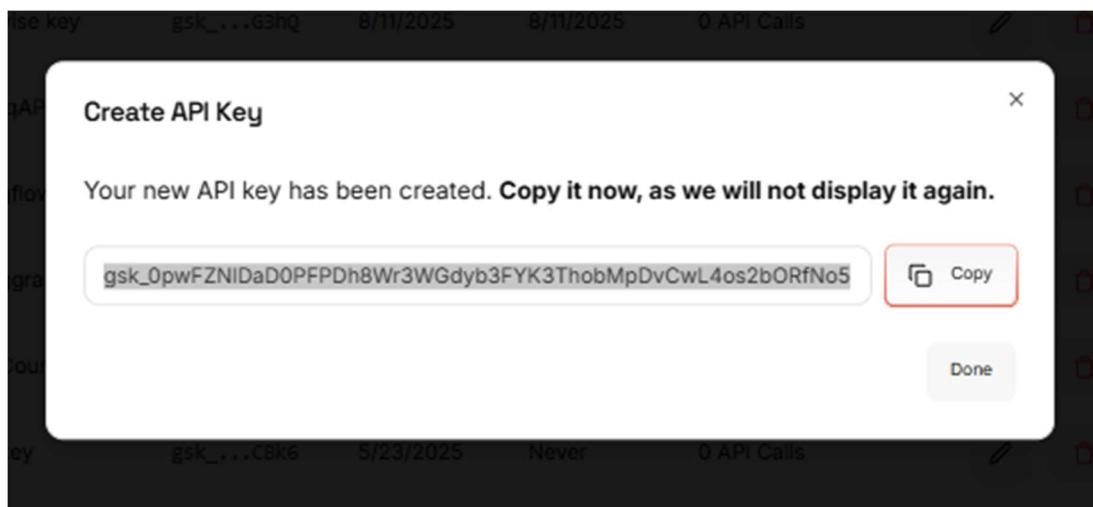
The screenshot shows the 'API Keys' section of the Groq dashboard. It displays a table of existing API keys and a red button labeled '+ Create API Key'.

NAME	SECRET KEY	CREATED	LAST USED	USAGE (24HRS)	Actions	
flowise key	gsk_...G3hQ	8/11/2025	8/11/2025	0 API Calls		
langflow test	gsk_...4Vyy	3/2/2025	3/3/2025	0 API Calls		
langgraph rag	gsk_...JKF1	1/11/2025	1/16/2025	0 API Calls		
myCourseKey	gsk_...bk0	8/16/2025	10/29/2025	24 API Calls		
mykey	gsk_...CBk6	5/23/2025	Never	0 API Calls		
mylangflow1	gsk_...Xiap	3/23/2025	10/20/2025	0 API Calls		
Text2SQL	gsk_...0vp6	5/4/2025	5/5/2025	0 API Calls		

4. Give a name to the API Key, then submit

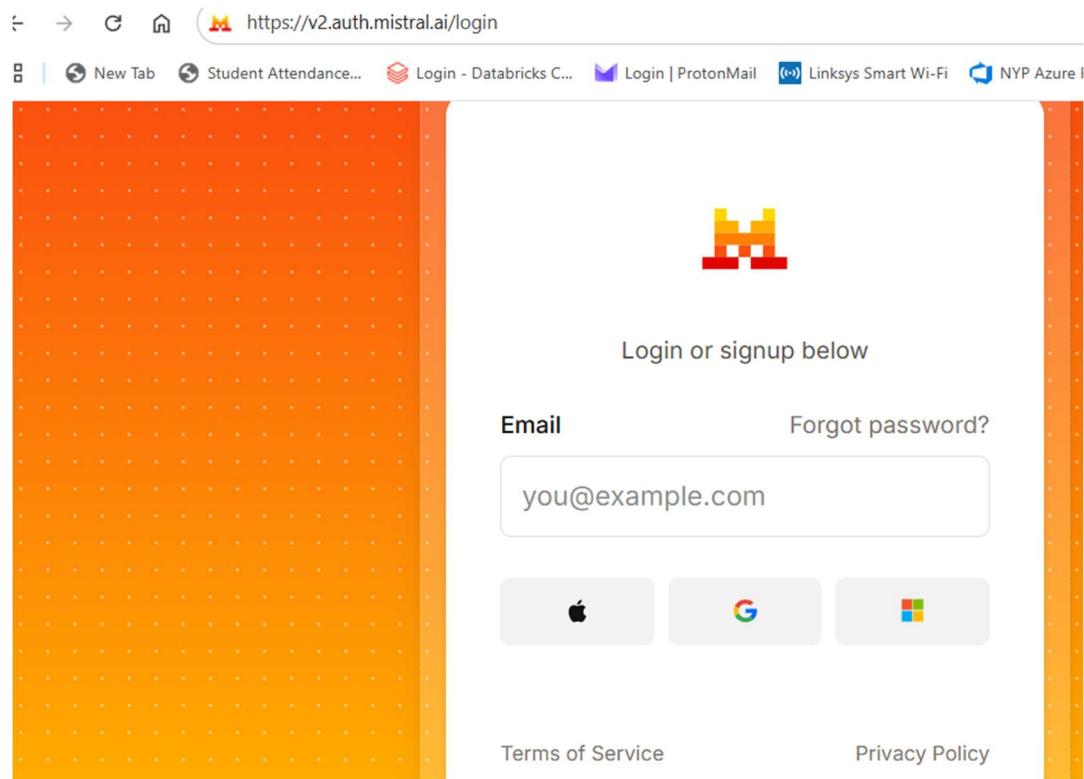


5. Copy the API key into a safe file. You will need the API key for Flowise development.

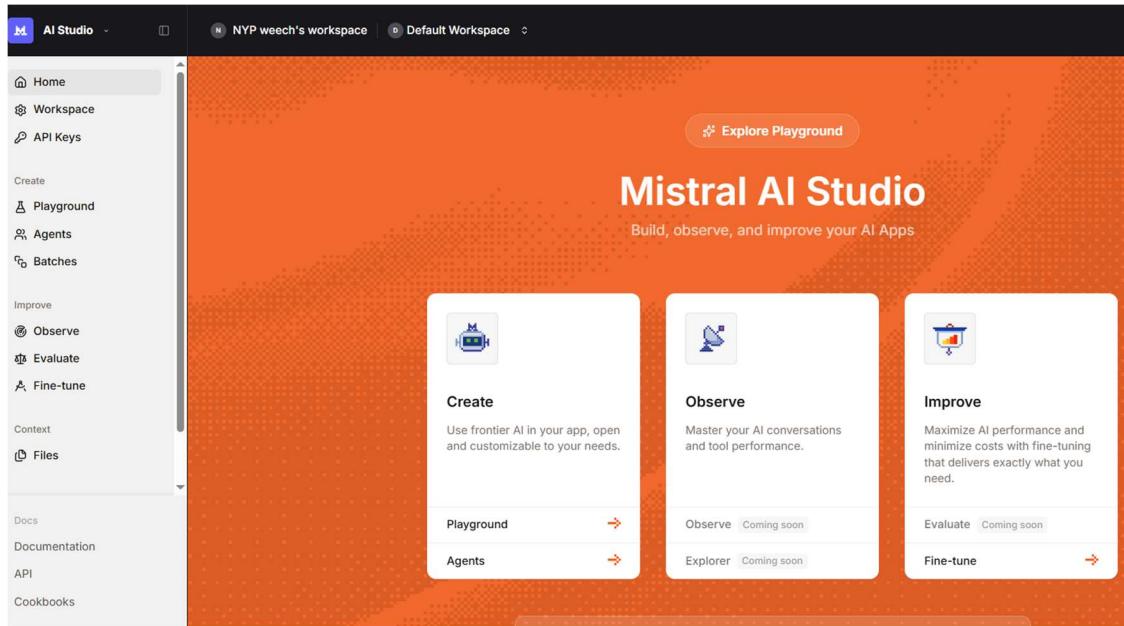


A2. Mistral -LLM API Key

1. Use the browser goto <https://v2.auth.mistral.ai/login>



2. Login in the mistral account created. Select from left menu-> API Keys.

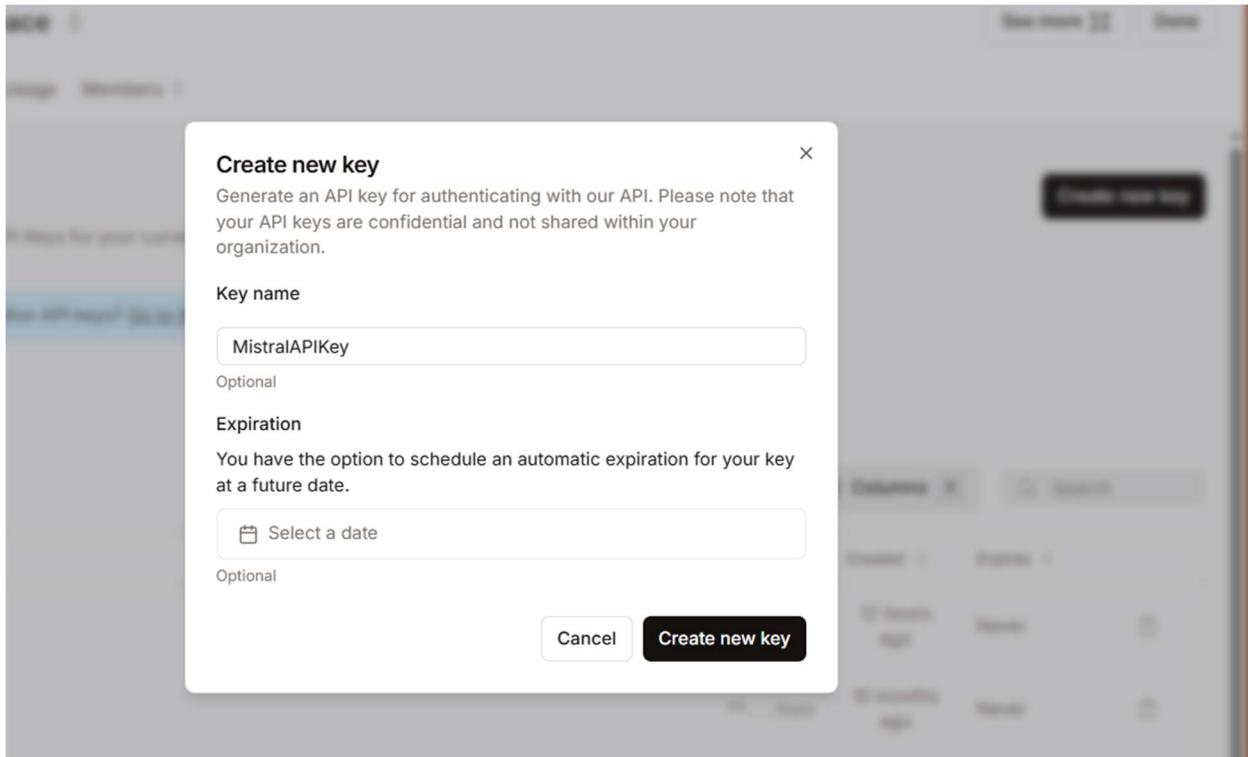


3. Create API key

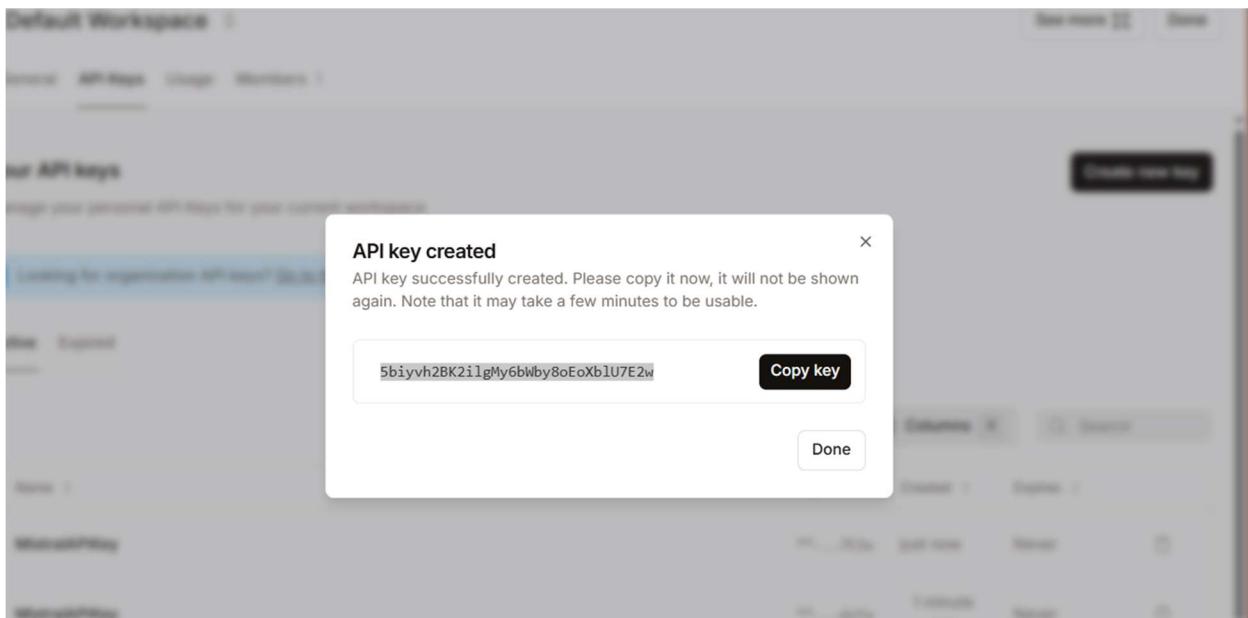
This screenshot shows the "Default Workspace" API Keys management screen. At the top, there are tabs for General, API Keys (which is selected), Usage, and Members 1. Below the tabs, a section titled "Your API keys" says "Manage your personal API Keys for your current workspace". A note says "Looking for organization API keys? Go to the admin console". There are "Active" and "Expired" buttons. A "Create new key" button is located in the top right. The main area is a table with columns: Name, Key, Created, Expires, and a delete icon. The table contains three rows:

Name	Key	Created	Expires	
flowise	**...LifI	12 hours ago	Never	
RAG	**...Ssoz	10 months ago	Never	
test_pixtral	**...i0yX	1 year ago	Never	

Enter Key name, select button create new key



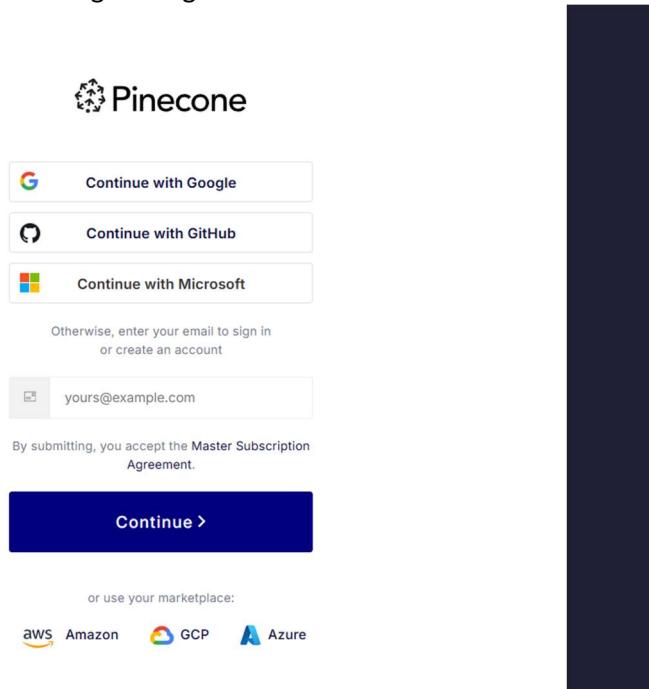
4. Store the API key in safe place, we will need during the exercise.



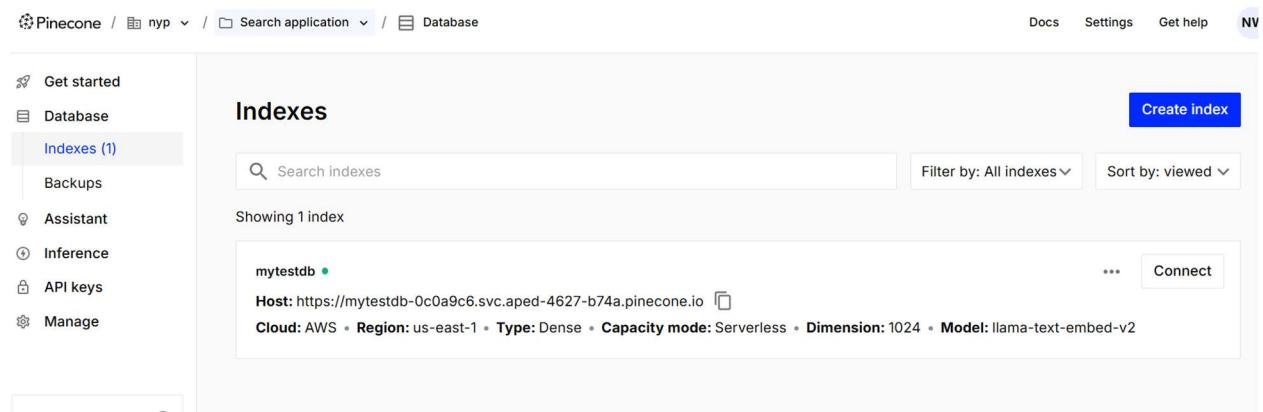
A3. Pinecone -Vector Database API Key

1. Use the browser goto <https://app.pinecone.io/?sessionType=signup>. Register a new account

2. Login Pinecone using the registered account



2. Login Pinecone using the registered account. Select API keys at the left menu.



The image shows the Pinecone web interface. The top navigation bar includes "Pinecone", a dropdown for "nyp", "Search application", and "Database". On the right are "Docs", "Settings", "Get help", and a user profile icon. The left sidebar has a "Get started" section and a "Database" section with "Indexes (1)" selected, showing "Backups", "Assistant", "Inference", "API keys", and "Manage". The main content area is titled "Indexes" with a "Create index" button. It features a search bar, filters, and a sort dropdown. Below is a table showing one index: "mytestdb". The table includes columns for "Host", "Cloud", "Region", "Type", "Capacity mode", "Dimension", and "Model". Buttons for "..." and "Connect" are also present.

3. Create a Pinecone API Key. Select button Create API Key

Pinecone / nyp / Search application / API keys

Docs Settings Get help NW

Name	Created on	Created by	Value	Permissions	Actions
flowisepineconekey	10/29/2025	NYP weech	pcsk_5CQQT_*****	All	...
mylim	1/17/2024	---	*****_*****_*****	All	...

+ Create API key

4. Provide a name to the API key name, then click Create key.

Generate new API key

Specify the name and permissions for your API key.

API key name

Permissions ⓘ

All Custom

[Upgrade now](#) to access advanced permissions, including the ability to manage and configure indexes and backups.

Cancel Create key

5. Store the API key in safe place. We will use during Flowise exercise.

