

**Utility**

# Accelerate Your AI Build Apps and Workflows at Lightning Speed

The Agentic AI Orchestrator

# Outline

- I. About Utility AI and Agentic AI Platforms
- II. Section 1: Quick start - Utility AI walk through
- III. Section 2: Quick start - Build your first conversational Bot
- IV. Section 3: Quick start - Build Your Own Financial Expert Bot
- V. Section 4: Quick start - Vector Database & RAG (Retrieval-Augmented Generation)
- VI. Section 5: Quick start - AI Agent with Tools in Utility

# About Utility AI & Agentic Platforms

# Single Agent

One agent performs a complete task with defined scope



## Characteristics:

- Performs one complete task
- Well-defined scope and capabilities
- Sequential processing
- Limited to its own knowledge and tools
- Simpler implementation

## Example Use Cases:

- Answering questions with supporting evidence
- Generating content based on specific criteria
- Analyzing a document or dataset
- Performing a specific task like code generation

# Multi-Agent System

Fleet of specialized agents interacting to achieve complex tasks



Coordinator Agent



Research Agent



User Proxy Agent



Service Agent



Analytics Agent



Web Service



Continuous communication & collaboration

## Characteristics:

- Multiple specialized agents working together
- Agents communicate and share information
- Parallel processing capabilities
- Diverse skills and knowledge domains
- Complex orchestration and coordination

## Example Use Cases:

- Complex research requiring multiple data sources
- Marketing campaign planning and analysis
- Software development with multiple components
- Business intelligence with diverse requirements



## Challenges of Multi-Agent Systems

- **Complexity:** More difficult to design, implement, and debug
- **Coordination:** Requires robust communication protocols
- **Resource Intensive:** Higher computational and API costs

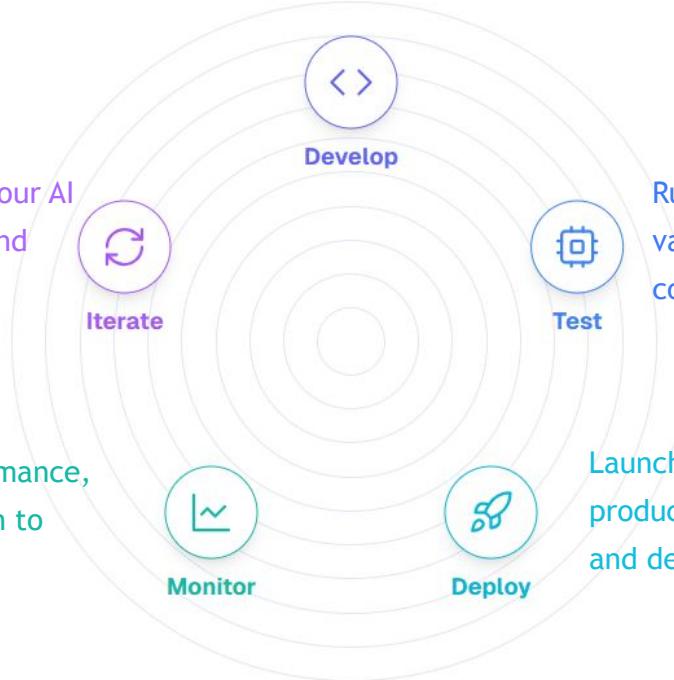
- **Reliability:** More points of potential failure
- **Consistency:** Maintaining coherent behavior across agents
- **Latency:** Communication overhead can slow performance

# The full AI Agentic development lifecycle in one place

Build AI agents at lightning speed with our powerful no-code Agentic Studio.

Continuously refine and enhance your AI agents to maximize their impact and performance.

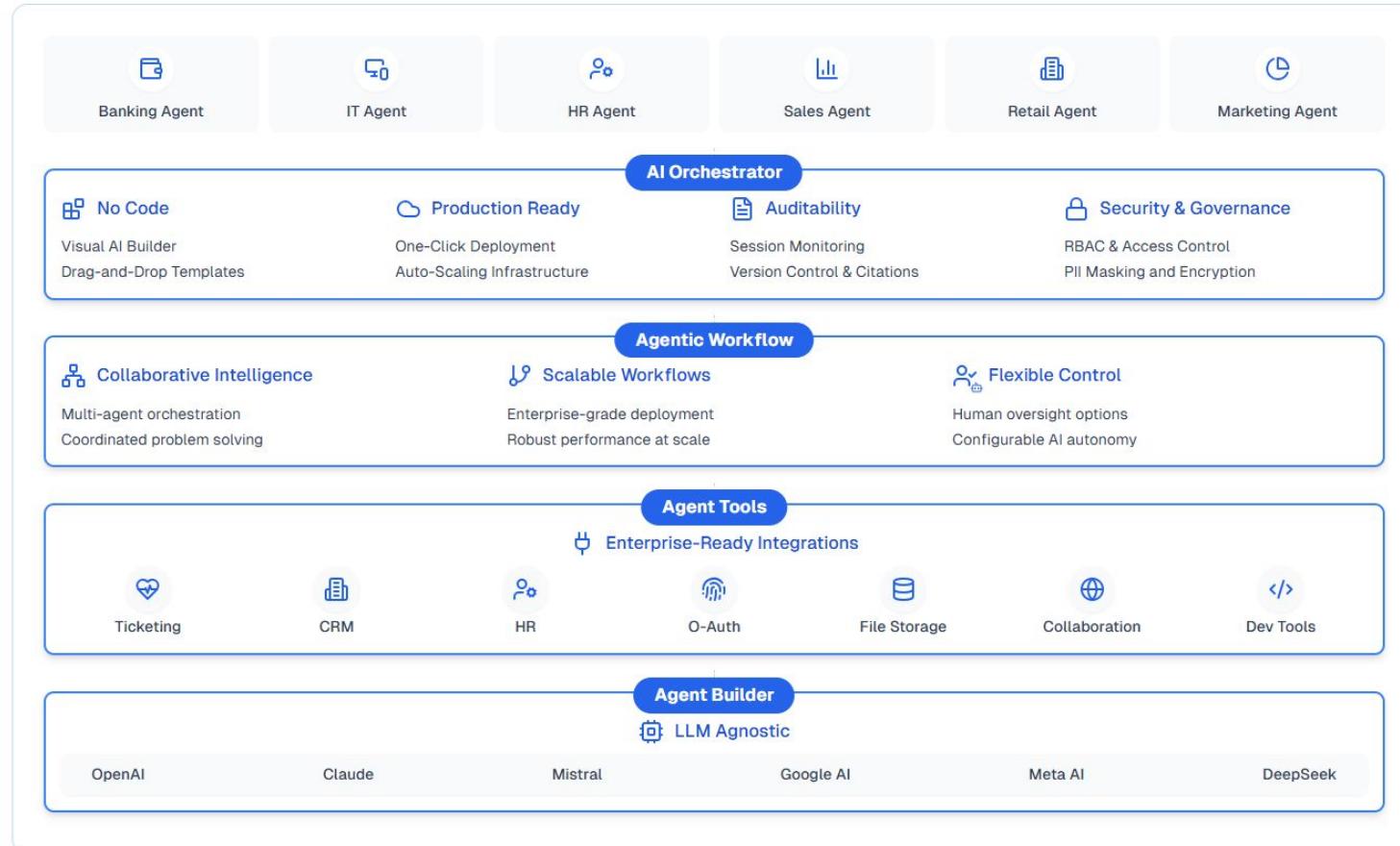
Gain real-time insights into performance, failures, and resource consumption to maintain optimal efficiency.



Run A/B tests on your agents and gather valuable feedback from your team for continuous improvement.

Launch instantly with a single click on secure, production-ready Microservices—fully cloud-agnostic and deployable on-premise.

# Utility Agentic Architecture



# The Agentic AI Use Cases Catalogue

# Customer Service



## Voice of the customer Agent

This AI agent analyzes customer interactions in real-time, detecting sentiment, intent, and complexity.



## Routing Agent

This AI agent escalates urgent cases, routes queries to the right agent, and update tickets.



## Knowledge Base Agent

This AI agent escalates urgent cases, routes queries to the right agent, and update tickets.



## Fulfillment Agent

Resolve customers' tickets instantly without any human interaction by connecting your LLM to your internal tools



## Human Assistant Agent

Assist your customers agent support team with information about your customers and solutions



## Proactive Support Agent

Anticipate customer needs by analyzing historical interactions and providing proactive support



# Marketing Agents



## SDR Agent

Automates lead qualification, outreach, and follow-ups to enhance sales pipeline efficiency.



## Content Creation Agent

Automates ideation, writing, and optimization, enhancing efficiency and creativity.



## AI Influencer Agent

Data-driven insights to engage audiences, personalize content, and optimize marketing strategies autonomously



## Social Listening & Trend Analysis Agent

Monitor and analyze social media and online conversations, identifying emerging trends, sentiments



## Dynamic Product Description Agent

Generates personalized product descriptions by analyzing customer behavior, preferences, to optimize sales



## Personalized Outreach Agent

An AI-powered personalized outreach agent uses data-driven insights to tailor communication strategies



# Procurement & Finance



## RFP generation

Streamlines proposal creation by analyzing requirements, suggesting relevant content, and ensuring compliance



## Streamline Invoice Matching with AI

Automates data extraction and validation, reducing errors and accelerating reconciliation.



## Boost Efficiency with AI-Powered Spend Analytics

Expense tracking, identifies cost-saving opportunities, and optimizes procurement decisions.



## Optimize Supplier Selection

Optimizing supplier selection analyzes performance, cost, and risk factors to recommend the best options



## Predict Demand with AI for Smarter Procurement

Analyzes historical data and market trends to forecast future demand accurately.



## Run Real-Time Competitive Pricing Analysis

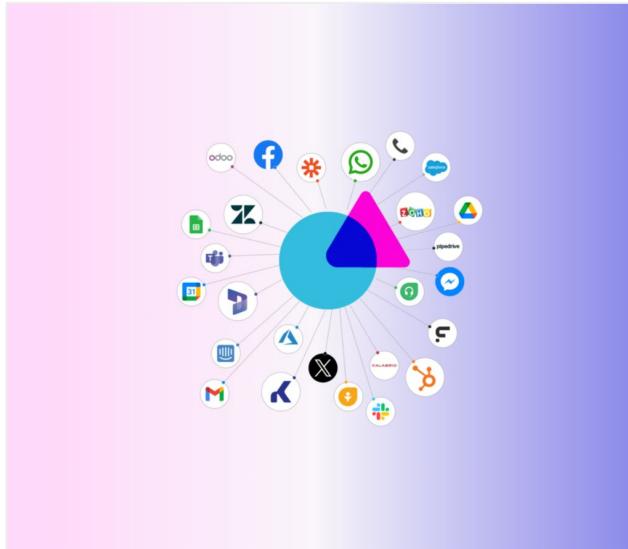
Gathers, analyzes, and derives insights from market trends, competitor strategies, and industry data.



# Section 1: Utility AI walk through

# Log In

Navigate to  
<https://cloud.ubilityai.com/>  
and log in to your Ubility  
account using your  
credentials.



Welcome to Ubility Workflow! 🎉

Please sign-in to your account and start the adventure

Email Address

Password  
 ⚡

**SIGN IN**

New on our platform? [Create an account](#)

or

 [Sign in with Google](#)

# Ubility Dashboard (Quick Overview)

The Dashboard gives you a summary of your activity on the platform. It shows:

- Workflow runs (pass/fail)
- Active webhooks, chatbots, and flows
- Total cost and token usage
- Indexes created
- Top performing and failing workflows

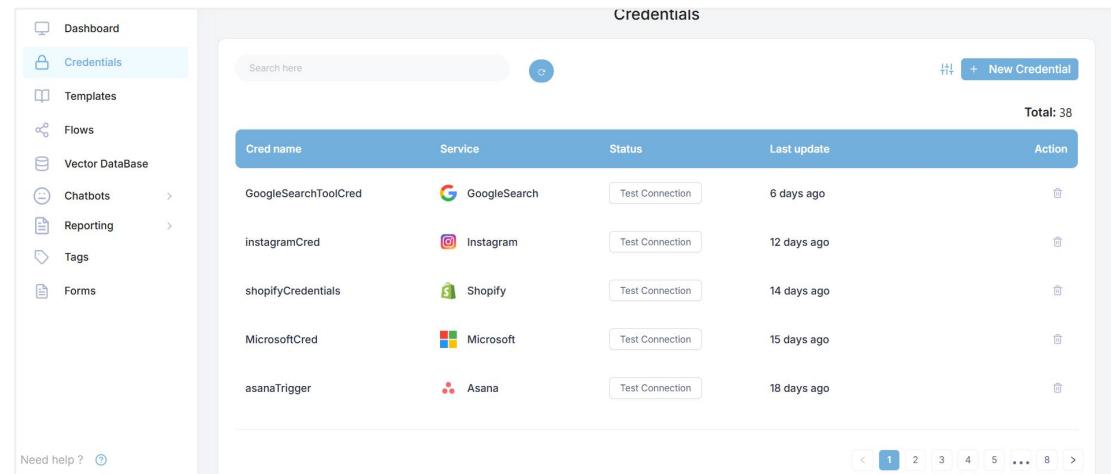


It's your go-to place to track performance and usage.

# Ubility Credentials (Quick Overview)

Ubility Credentials let you securely store and manage API keys, OAuth tokens, and access permissions for your workflows and agents.

They ensure safe, encrypted connections to external tools, so your automations run smoothly and securely.



The screenshot shows the Ubility platform's credential management interface. On the left, a sidebar menu includes options like Dashboard, Credentials (which is selected and highlighted in blue), Templates, Flows, Vector DataBase, Chatbots, Reporting, Tags, and Forms. A 'Need help?' link is at the bottom of the sidebar. The main area is titled 'Credentials' and displays a table with 38 entries. The columns are 'Cred name', 'Service', 'Status', 'Last update', and 'Action'. Each row shows a credential name, its associated service (e.g., GoogleSearch, Instagram, Shopify, Microsoft, Asana), its status (with a 'Test Connection' button), the last update time (e.g., 6 days ago, 12 days ago, 14 days ago, 15 days ago, 18 days ago), and an 'Action' column with a trash icon. A search bar and a 'New Credential' button are at the top right of the main table area. Navigation links for pages 1 through 8 are at the bottom right.

Cred name	Service	Status	Last update	Action
GoogleSearchToolCred	GoogleSearch	<button>Test Connection</button>	6 days ago	
instagramCred	Instagram	<button>Test Connection</button>	12 days ago	
shopifyCredentials	Shopify	<button>Test Connection</button>	14 days ago	
MicrosoftCred	Microsoft	<button>Test Connection</button>	15 days ago	
asanaTrigger	Asana	<button>Test Connection</button>	18 days ago	

# Templates Section (Quick Overview)

The Templates section offers ready-to-use workflows built for you.

Just select a template, customize it if needed, and run it, no need to build from scratch.

The screenshot shows the ubility platform's Templates section. On the left, a sidebar navigation includes: Dashboard, Credentials, Templates (selected), Flows, Vector DataBase, Chatbots, Reporting, Tags, and Forms. The main area is titled "Templates" and displays a grid of 10 workflow templates. Each template card includes a title, a brief description, and a "View Template" button.

Template	Description	Tags	Action
Ask questions about a PDF using AI	This workflow is designed to handle a variety of questions, making it a versatile tool for customer support tickets to higher priority levels or specific teams based on...	Qanda, LangChain, openAI +4 more	View Template
Auto Escalation Based on Ticket	Automatically escalate customer support tickets to higher priority levels or specific teams based on...	Escalation, Tickets Customer Support, +4 more	View Template
Automated USD Exchange Rate Notification	This use case showcases an automated workflow that fetches the latest USD exchange rates an...	ExchangeRates, currencies HTTP Request, +2 more	View Template
Chat with a database using AI	Use AI to interact with SQL databases through a chat interface, allowing users to easily...	Database, postgres, openAI +2 more	View Template
Chat with a GoogleSheet using AI	Imagine talking to your Google Sheet as if it were a teammate! This workflow lets you ask...	GoogleSheet, ChatGPT, Data +2 more	View Template
Chat with OpenAIs GPT via a simple Telegram bot	Leveraging the power of GPT-4 with a Telegram bot opens up endless possibilities for automate...	Telegram Trigger	View Template
Convert JSON data into GoogleSheet	Transform your JSON data into a Google Sheet. By doing this, you'll make data management a breeze...	JSON, GoogleSheet	View Template
Create a Task in ClickUp	This use case demonstrates how to automate the creation of a task in ClickUp using a schedule trigger ...	ClickUp, ClickUp API, JSON +2 more	View Template
Create Social Media Posts from a Blog Using ChatGPT	This workflow automates converting a blog post into an Instagram post. It captures the...	Instagram, ChatGPT, ClickUp +2 more	View Template
Customer Support And Sales Agent	This workflow processes Zendesk tickets by analyzing customer tone. If negative, it retrieves solutions...	Zendesk, ClickUp, ChatGPT +2 more	View Template

# Flows Section (Quick Overview)

The Flows section is where you build, manage, and access all your workflows.

Every automation you create on the platform is organized and saved here.

The screenshot shows the 'Flows' section of a platform. On the left is a sidebar with navigation links: Dashboard, Credentials, Templates, Flows (which is highlighted), Vector DataBase, Chatbots, Reporting, Tags, and Forms. Below the sidebar is a 'Need help?' button. The main area is titled 'Flows' and shows a grid of nine workflow cards. Each card includes icons representing different components like AI, databases, and messaging, the name of the flow, and the time it was created. A total count of 59 flows is displayed at the top right.

Flow Name	Created Ago
RAG-QA Chain With Memory	2 days ago
ActivityOne-Build Your Own...	2 days ago
A Quick Example	3 days ago
AI Agent Example	3 days ago
Summarization+Compare Doc	4 days ago
RAG-Q&A Chain	4 days ago
(Icon Placeholder)	4 days ago
(Icon Placeholder)	5 days ago

# Vector Database Section (Quick Overview)

The Vector Database section lets you create and manage indexes to store files or documents as embeddings.

It's used in RAG (Retrieval-Augmented Generation) workflows to answer questions based on your own data.

The screenshot shows the ubility platform's Vector DataBase section. On the left, a sidebar menu includes options like Dashboard, Credentials, Templates, Flows, Vector DataBase (which is highlighted), Chatbots, Reporting, Tags, and Forms. The main area is titled "Indexes" and displays four entries:

Index Name	Last Updated	Size	Action
isdb	4 days ago	Size: 33.9mb	trash
chatbotsalesdata	1 month ago	Size: 36.2mb	trash
customersdata	3 months ago	Size: 599.4mb	trash
crayon	3 months ago	Size: 15.7mb	trash

# Reporting Section (Quick Overview)

The Reporting section provides detailed insights into your workflows:

**Logs:** View all executions with full details and status.

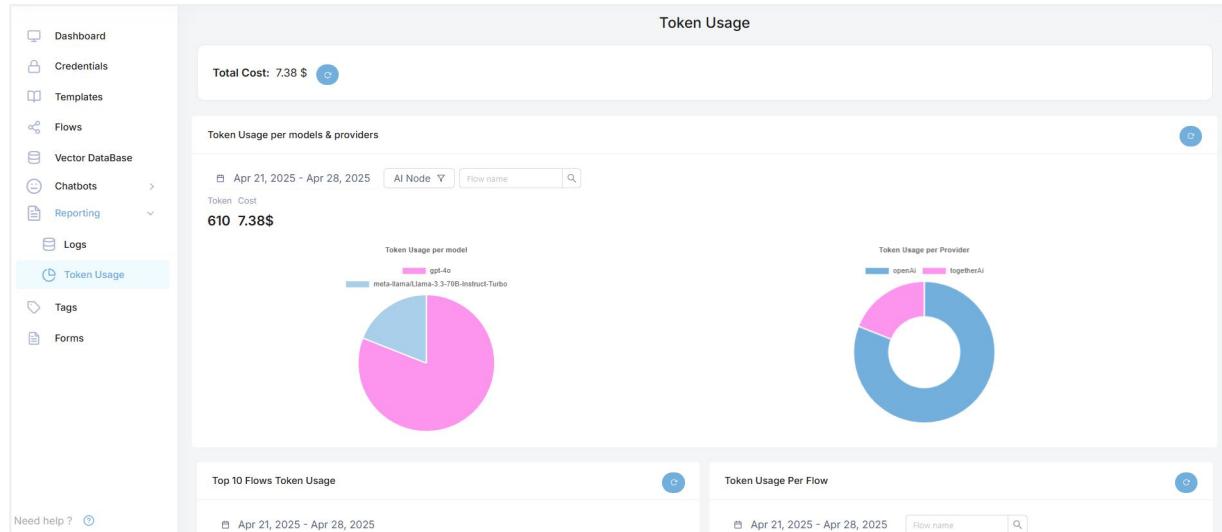
The screenshot shows the 'Logs' section of a reporting interface. On the left is a sidebar with navigation links: Dashboard, Credentials, Templates, Flows, Vector DataBase, Chatbots, Reporting (selected), Logs (selected), Token Usage, Tags, and Forms. A 'Need help?' link is at the bottom of the sidebar. The main area is titled 'Logs' and displays a table of execution logs. The table has columns: Flow Name, Date, Run Type, and Flow status. The data in the table is as follows:

Flow Name	Date	Run Type	Flow status
RAG-QA Chain With Memory	2025-04-25T19:08:08	TRIGGERED	PASS
Turn YouTube Videos into	2025-04-25T13:23:45	TESTING	FAIL
ActivityOne-Build Your Own Expert Bot	2025-04-25T12:59:57	TRIGGERED	PASS
A Quick Example	2025-04-25T11:06:03	TESTING	PASS
Draft	2025-02-05T12:01:33	TESTING	PASS

At the top right, there are summary statistics: Total: 114, PASS: 108, FAIL: 6. Below the table are pagination controls: < 1 2 3 4 5 ... 23 >.

# Reporting Section (Quick Overview)

**Token Usage & Costs:**  
Track usage and spending  
per model and provider.



## Section 2: Quick start - Build your first conversational Bot

# What we are going to achieve:

By building our first conversational bot, we'll learn how to integrate AI into a workflow using LangChain and a Chat Trigger.

We'll explore how to structure prompts, capture user questions, and return intelligent responses. This hands-on activity lays the foundation for creating advanced AI-driven automation.

# What is LangChain?:

LangChain is a framework that helps developers connect AI language models with real-world data and tools.

Instead of just talking to a chatbot that knows nothing, LangChain lets the AI search databases, use tools like calculators or search engines, and even remember past conversations.

That's what makes it perfect for building smart, interactive bots and assistants.

[LangChain Link](#)

# Start a new flow

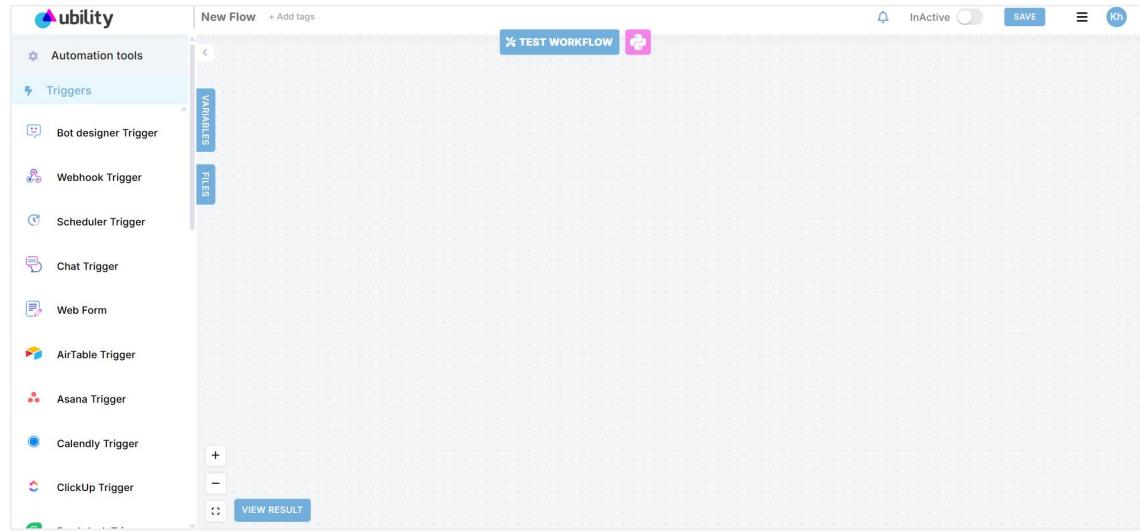
To create a workflow in Ubility,  
head into the Flows section,  
and click on 'New Flow' button

The screenshot shows the 'Flows' section of the Ubility interface. On the left, a sidebar menu includes 'Dashboard', 'Credentials', 'Templates', 'Flows' (which is selected and highlighted in blue), 'Vector DataBase', 'chatbots', 'Reporting', 'Tags', and 'Forms'. Below the sidebar is a 'Need help?' link. The main area is titled 'Flows' and displays a grid of 12 flow cards. Each card contains a thumbnail icon, the flow name, and a timestamp. A red box highlights the 'New Flow' button in the top right corner of the main area. The total number of flows is indicated as 'Total: 59'.

Flow Name	Created
RAG-QA Chain With Memory	2 days ago
ActivityOne-Build Your Own...	2 days ago
A Quick Example	3 days ago
AI Agent Example	3 days ago
Summarization+Compare Doc	4 days ago
RAG-Q&A Chain	4 days ago
(Icon not clearly visible)	4 days ago
(Icon not clearly visible)	4 days ago
(Icon not clearly visible)	5 days ago

# RPA Designer

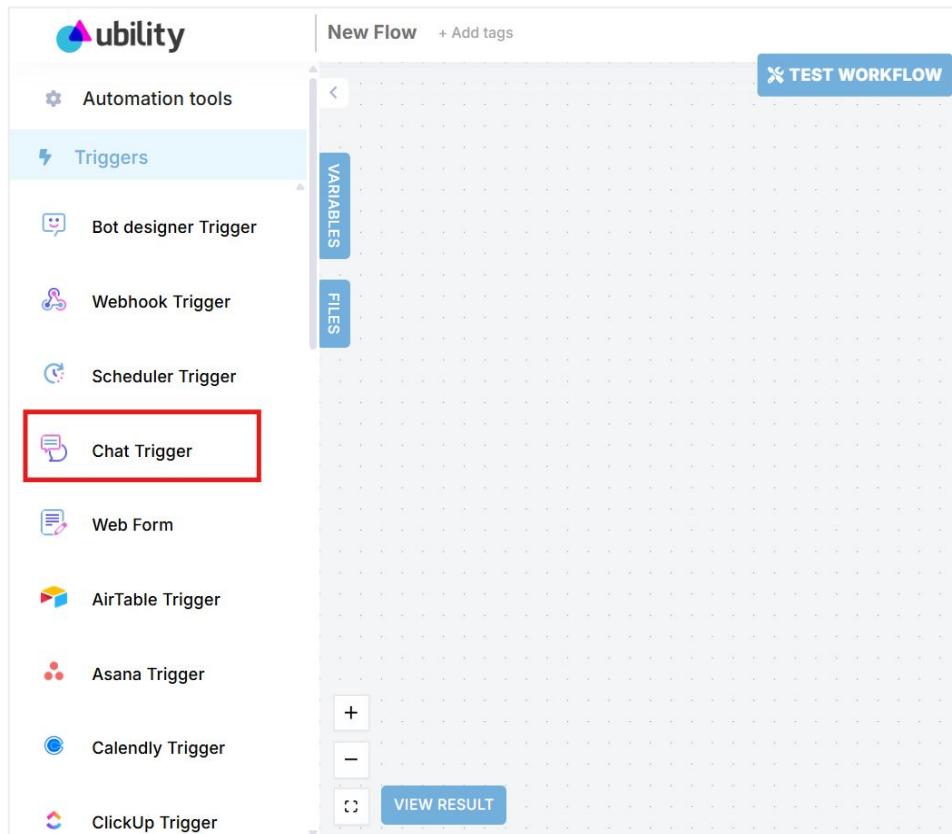
This will take you to the RPA Designer page where you can create and start building your workflows



# Step 1 – Chat Trigger

First, let's drag and drop a Chat Trigger into the designer.

But before, let's explain what is a trigger.



# What is a trigger?

A trigger is the starting point of any workflow.

It listens for a specific event like receiving a message, a webhook call, or a scheduled time, and activates the workflow when that event happens.

Think of it as the signal that tells your automation when to begin.

# What is Chat Trigger?

The Chat Trigger is designed for workflows involving LangChain nodes.

Use the Chat Trigger node when building AI workflows for chatbots and other chat interfaces.

**ChatbotQuestion:** This field is where you input the question you want to ask within the workflow. The question is utilized as a variable in the LangChain nodes.

**Chat Node:** This field lets you select which LangChain node to connect with.

**ChatbotResponse:** This field automatically stores the response generated by the LangChain node and links it to this variable for further interaction.

# Step 2 – Chat Trigger ChatbotQuestion

Enter a very simple question or greetings such as 'Hello!' in the ChatbotQuestion field

The screenshot shows a workflow editor interface with a central canvas and a right-hand sidebar for configuration.

**TEST WORKFLOW** (Workflow Name) | Python icon

**Chat Trigger** (Selected node, indicated by a blue border)

**Chat Variables** (Section)

**ChatbotQuestion** (Section)

Hello! (Value entered in the ChatbotQuestion field)

**Chat Node** (Section)

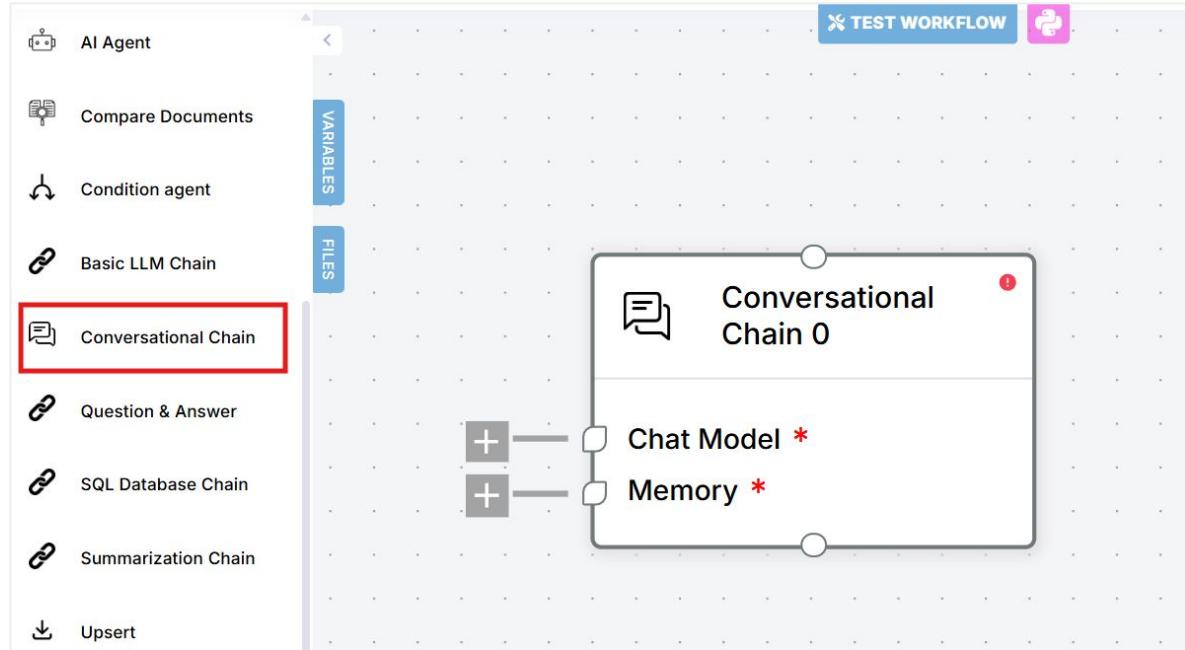
The value will be taken from the (Description)

**ChatbotResponse** (Section)

None (Value entered in the ChatbotResponse field)

# Step 3 – Conversational Chain Node

Next, drag and drop a conversational chain node into the designer

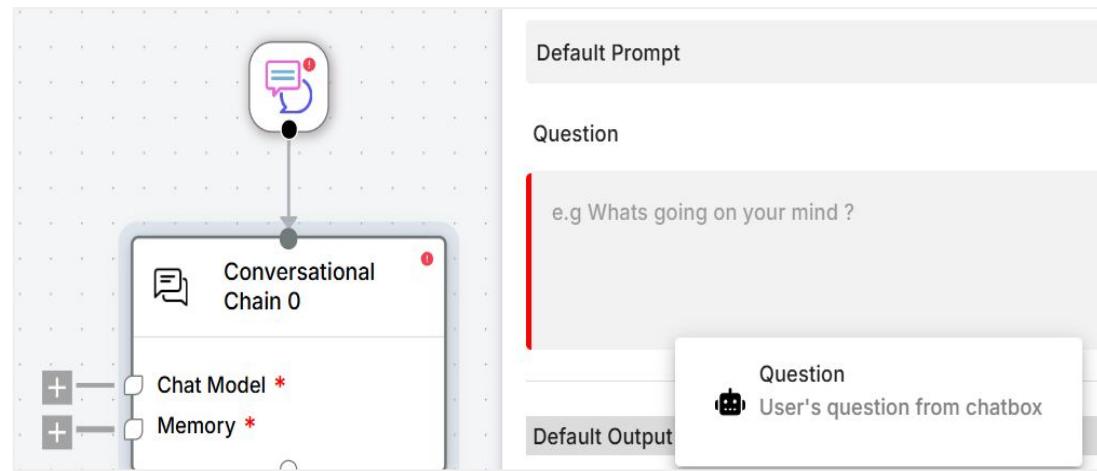


# Step 4 – Conversational Chain Node Question

Connect it to the chat trigger,  
and assign the question from  
the chatbox.

## What does that mean?

This means that the question  
we provided earlier in the  
chatbotQuestion is used as a  
question in the conversational  
chain

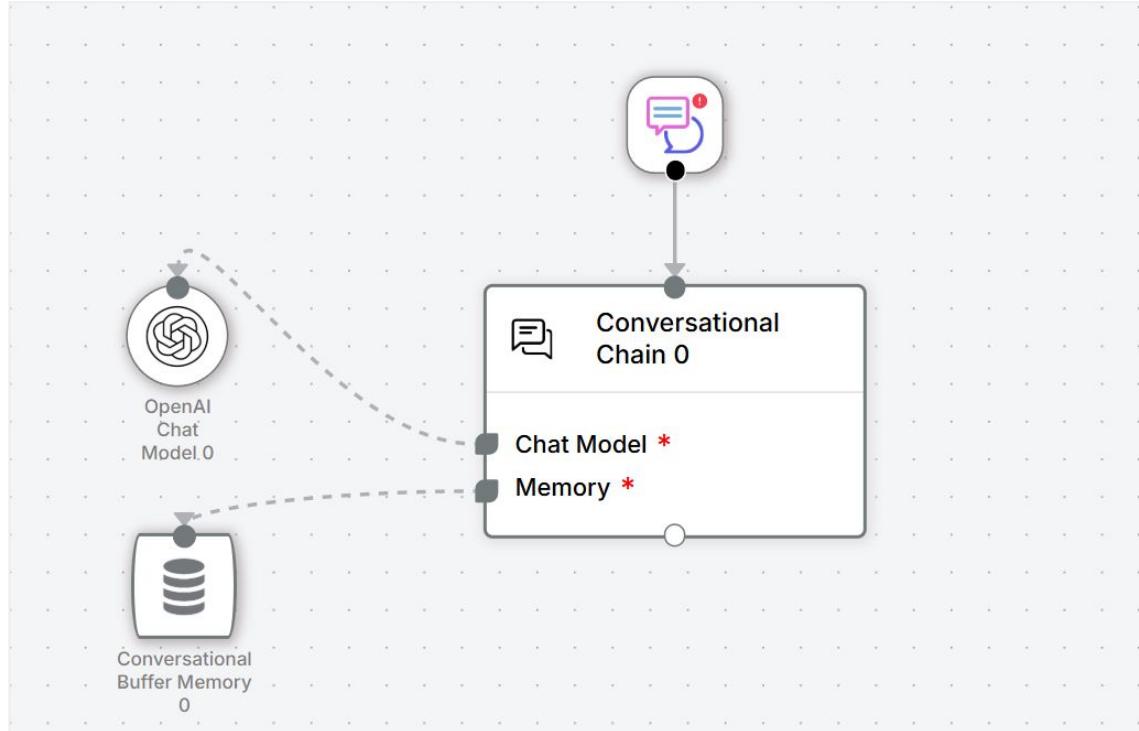


# Step 5 – Conversational Chain Sub-nodes

Next, add the two main sub-nodes of the conversational chain:

- 1- Chat model
- 2- Memory

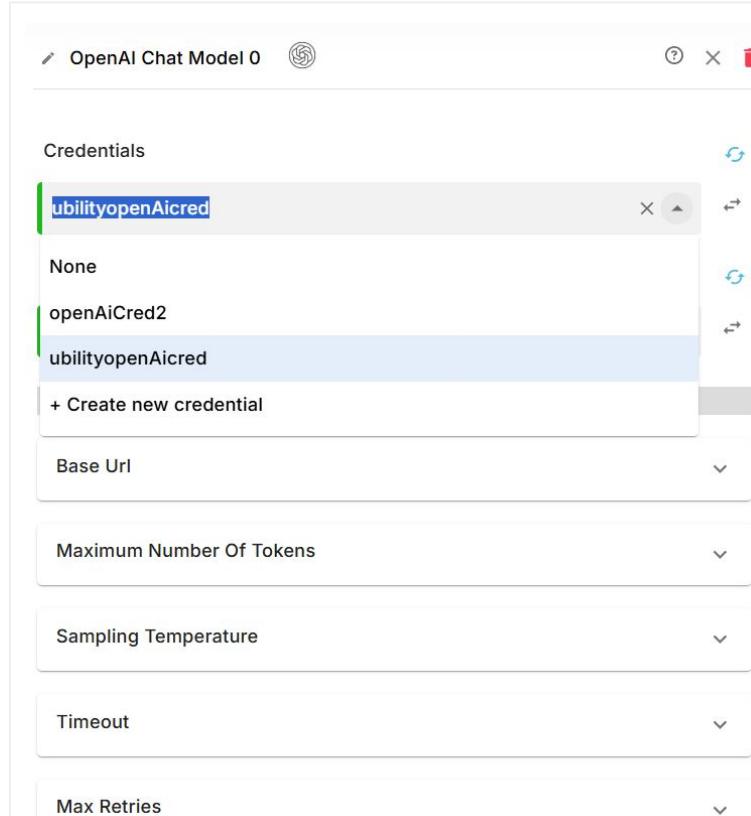
In this example we are using OpenAI chat model and the conversational Buffer Memory



# Step 6 – OpenAi Chat Model Sub-node

To use OpenAi chat model or any other chat model, you have to create credentials for it.

But first, let's discover what are credentials.



# What are credentials?

Credentials in Ubility allow access to external connectors securely

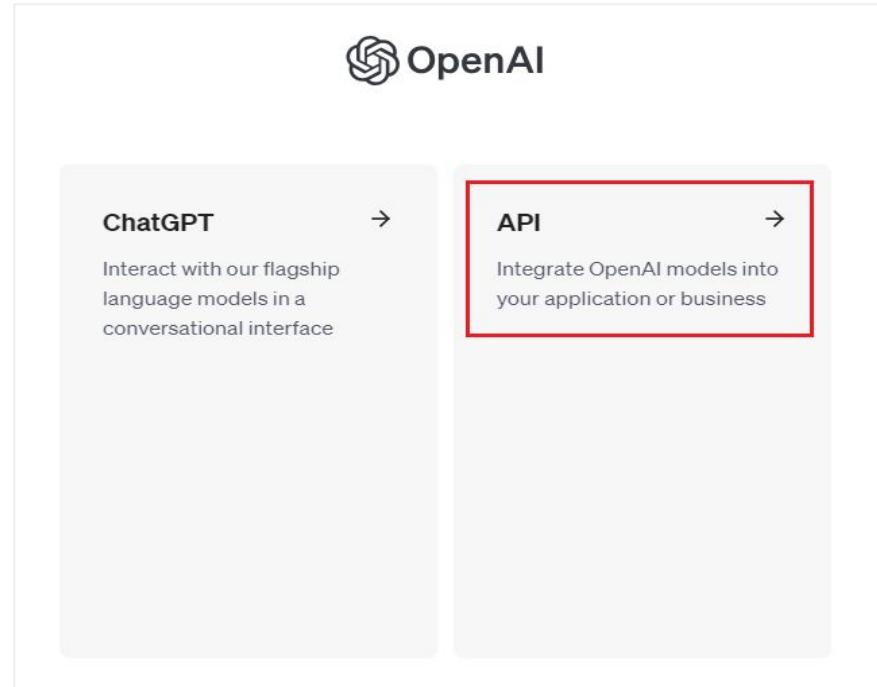
The screenshot shows the Ubility platform interface. On the left, there's a sidebar with icons for Credentials (highlighted with a red border), Templates, Flows, Vector DataBase, Chatbots, Reporting, Tags, and Forms. The main area is titled 'Credentials' and shows a list of three credentials: 'telegramCredential' (Service: Telegram, Status: Test Connection, Last update: 7 months ago), 'slackCredential2' (Service: Slack, Status: Test Connection, Last update: 7 months ago), and 'ubilityopenAicred' (Service: OpenAI, Status: Test Connection, Last update: 7 months ago). A search bar at the top is labeled 'Search here'. In the top right, there's a '+ New Credential' button and a 'Total: 38' indicator. At the bottom right, there's a navigation bar with page numbers from 1 to 8.

Cred name	Service	Status	Last update	Action
telegramCredential	Telegram	Test Connection	7 months ago	
slackCredential2	Slack	Test Connection	7 months ago	
ubilityopenAicred	OpenAI	Test Connection	7 months ago	

# How To Create OpenAi Credentials?

1- Log in to openAi

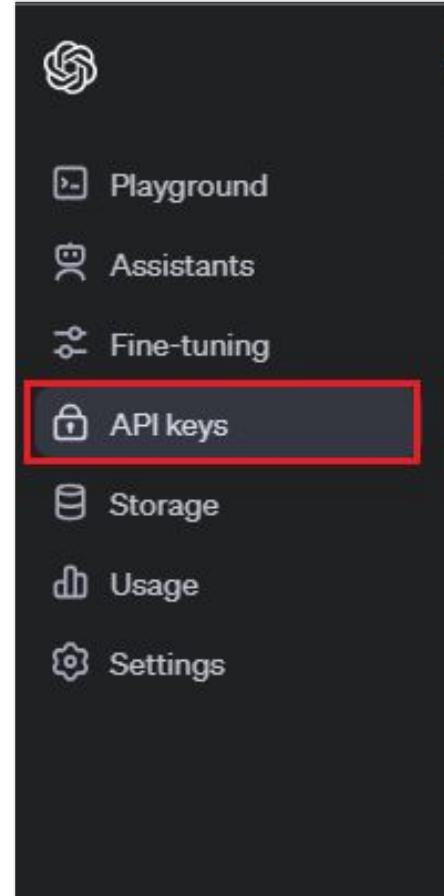
2- Choose API Section



# How To Create OpenAi Credentials?

## 3.1- Creating Security Key:

From the homepage head to the menu on the right and hover over the lock icon, as it represents the API Keys section, click on it.



# How To Create OpenAi Credentials?

3.2- Now click on ‘Create new security Key’

The screenshot shows the 'API keys' section of the OpenAI dashboard. A prominent red rectangular box highlights the '+ Create new secret key' button at the bottom left. The page contains the following text:

**API keys**

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically disable any API key that we've found has leaked publicly.

Enable tracking to see usage per API key on the [Usage page](#).

You currently do not have any API keys  
Create one using the button below to get started

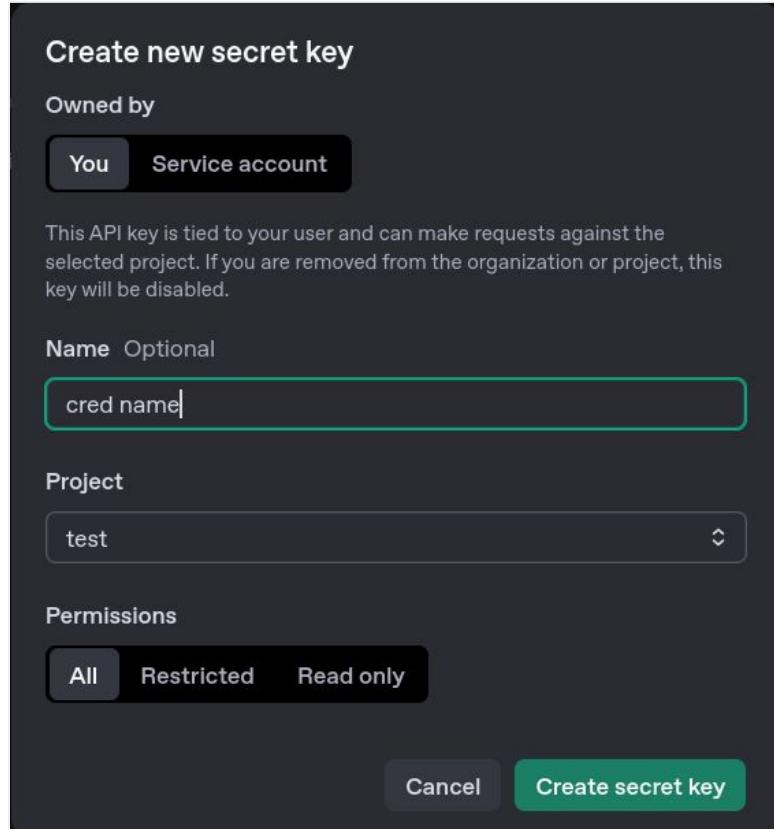
+ Create new secret key

# How To Create OpenAi Credentials?

## 4- Security Key creation process

First, name your key.

Then choose the type of permission and click on 'Create security key'



# How To Create OpenAi Credentials?

## 5- Your Key has been generated

**Note:** Please make sure that you save the key someplace you can access it again since this key will only appear once

**Save your key**

Please save your secret key in a safe place since **you won't be able to view it again**. Keep it secure, as anyone with your API key can make requests on your behalf. If you do lose it, you'll need to generate a new one.

[Learn more about API key best practices](#)

Copy

**Permissions**

Read and write API resources

Done

# How To Create OpenAi Credentials in Ubility?

1. Log in to ubility

1. head to the Credentials tab, and create a new credential.

The screenshot shows the Ubility platform interface. On the left, there is a sidebar with icons for Credentials (selected), Templates, Flows, Vector DataBase, Chatbots, Reporting, Tags, and Forms. The main area is titled 'Credentials' and contains a search bar and a 'Total: 38' count. A blue button labeled '+ New Credential' is highlighted with a red box. Below this, a table lists three credentials:

Cred name	Service	Status	Last update	Action
telegramCredential	Telegram	<button>Test Connection</button>	7 months ago	<span>trash</span>
slackCredential2	Slack	<button>Test Connection</button>	7 months ago	<span>trash</span>
ubilityopenAicred	OpenAI	<button>Test Connection</button>	7 months ago	<span>trash</span>

Pagination controls at the bottom right indicate page 8 of 8.

# How To Create OpenAi Credentials in Ubility?

3. Give it a name

4. Enter the API key that you have generated

5. the organization section is optional

6. Click on create.

Add new Credential X

Name

Service or app to connect to  
 OpenAI ✖

API Key  
 ✖

Organization (Optional)

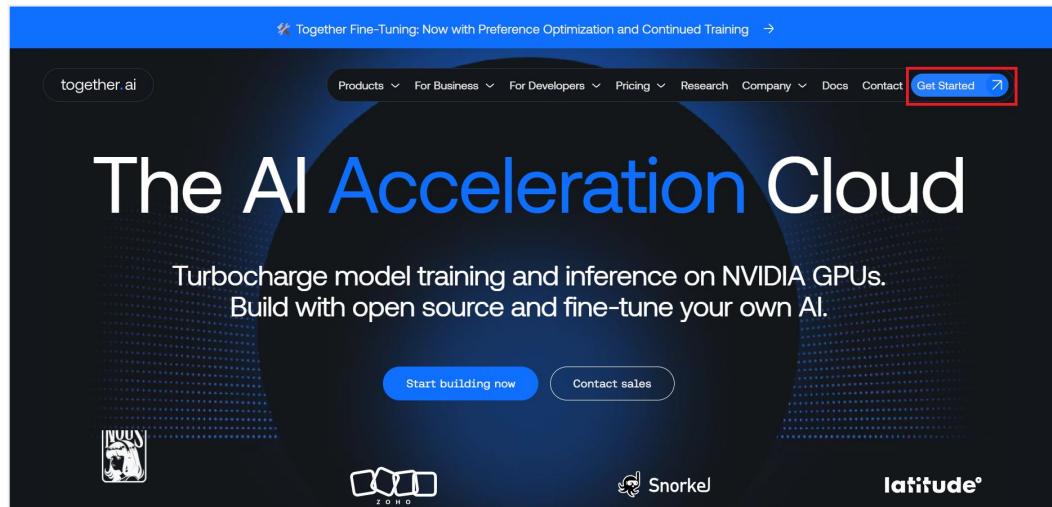
Cancel Create

# How To Create TogetherAi Credentials?

To create credentials for the TogetherAi model, we have to get TogetherAi API key.

First, head over TogetherAi website and click on **Get Started** button.

[TogetherAi](#)

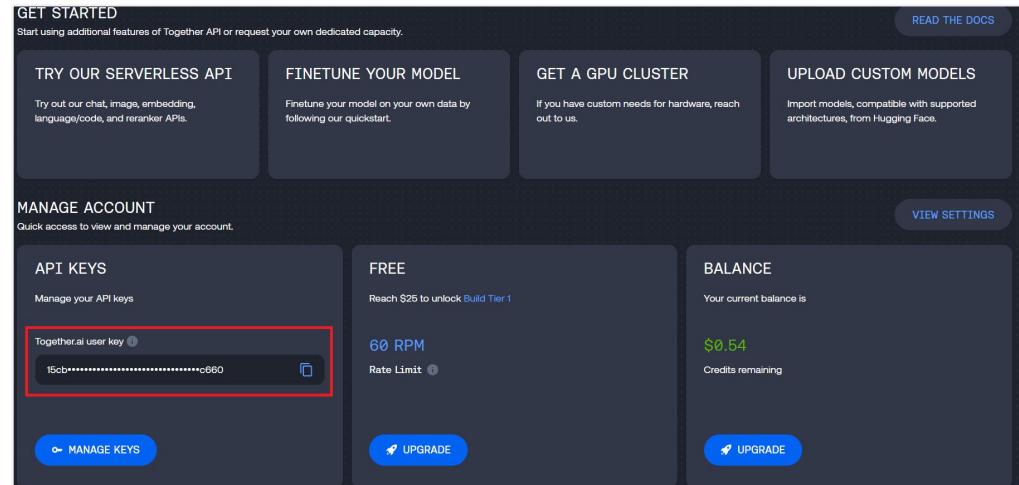


# How To Create TogetherAi Credentials?

Sign up with your account, and then go to the dashboard section.

There, you can find an API key.

Copy it and save it.



# How To Create TogetherAi Credentials?

Now go to the credentials section in Utility and click on **create credentials** button.

Provide a name for your credentials.

Choose TogetherAi as a service or app to connect to.

And finally paste the api key you copy it earlier.

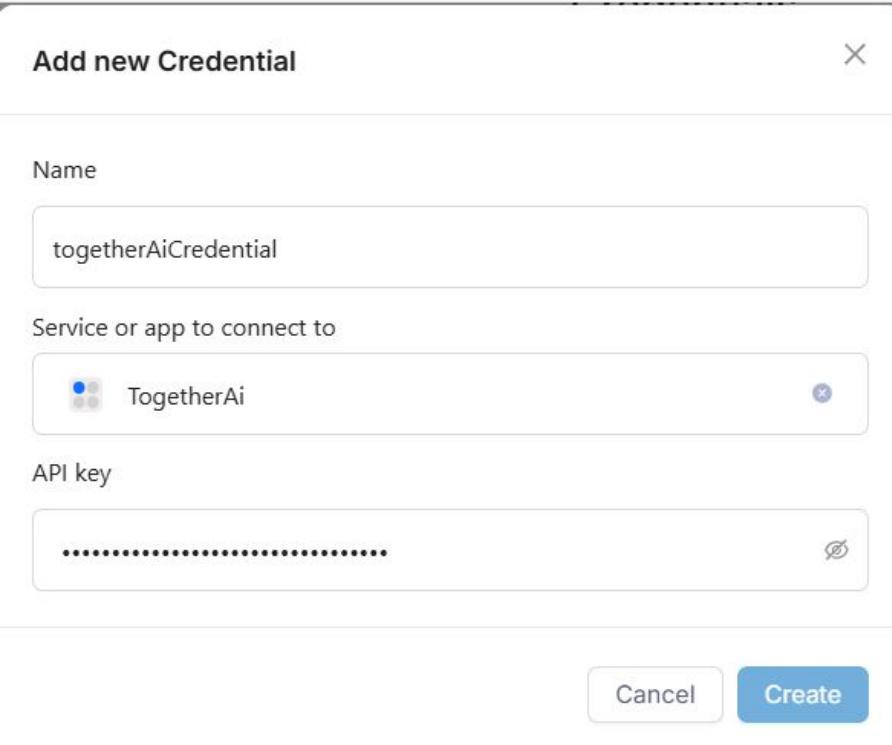
Add new Credential X

Name  
togetherAiCredential

Service or app to connect to  
 TogetherAi

API key  
..... (copy)

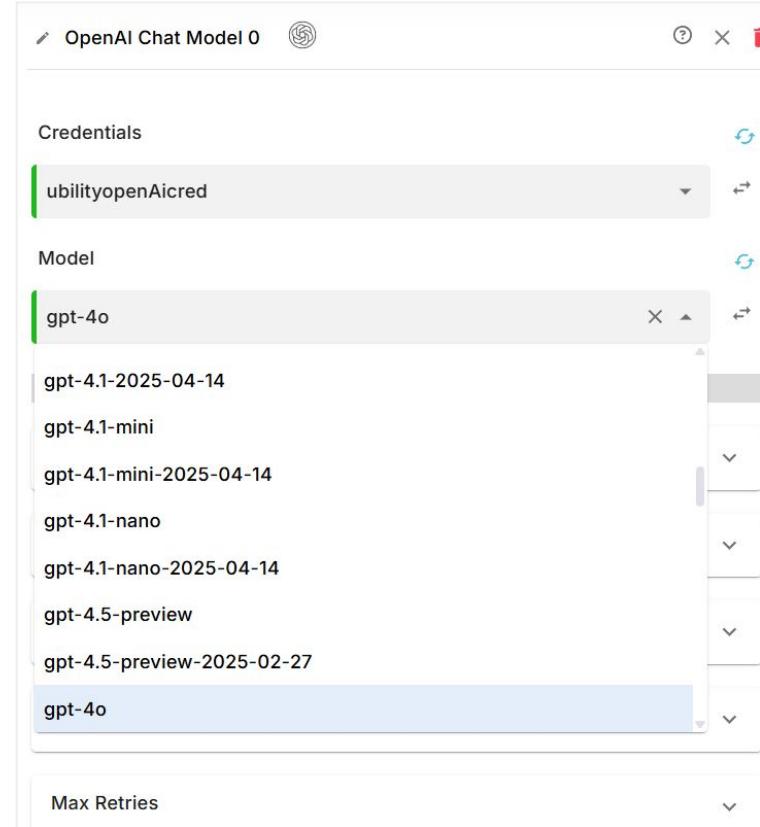
Cancel Create



# Step 7 – OpenAi Choosing a Chat Model

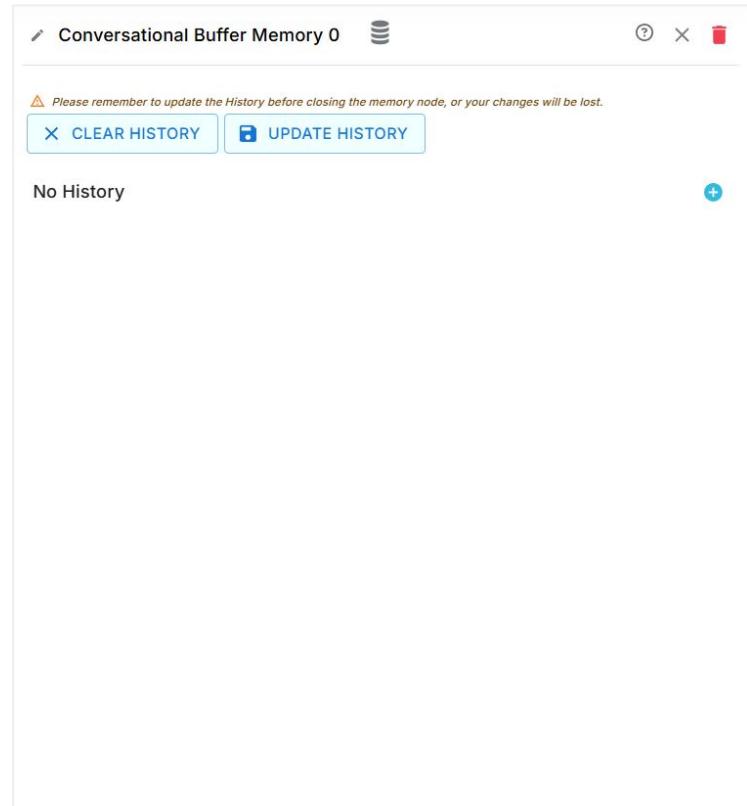
After creating credentials for OpenAi, now it is the time to choose the chat model we want to work with.

In this example we will work with gpt-4o model



# Step 8 – Conversational Buffer Memory

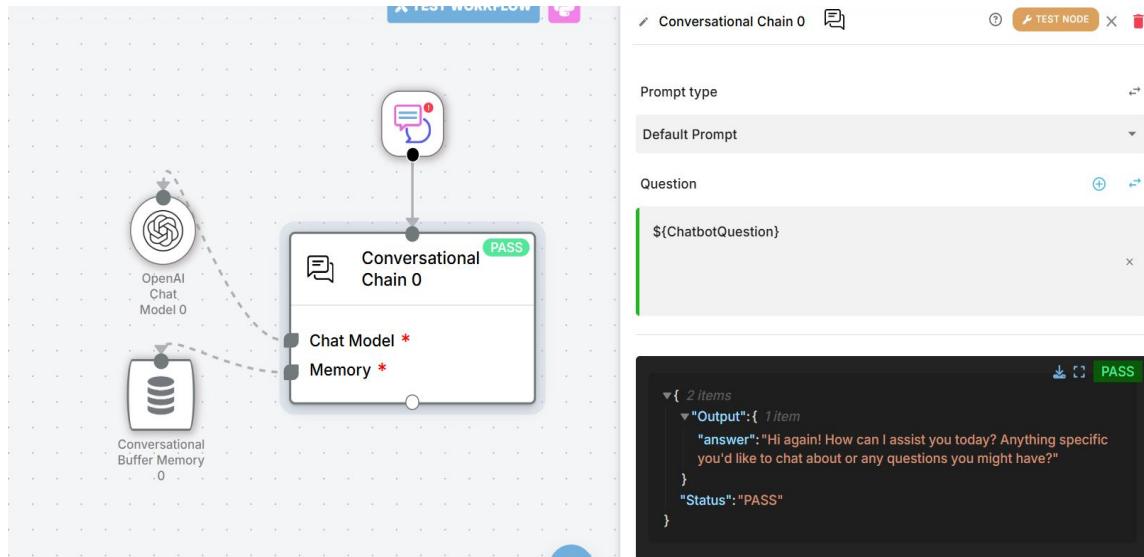
**For the memory:** we will be using the 'Conversational Buffer Memory' to maintain the context of the conversation.



# Step 9 – Test Node

After setting up the chat model and the memory, we click on ‘Test Node’ button to get the output result of the question we asked.

In this example it's a simple greeting (Hello!)



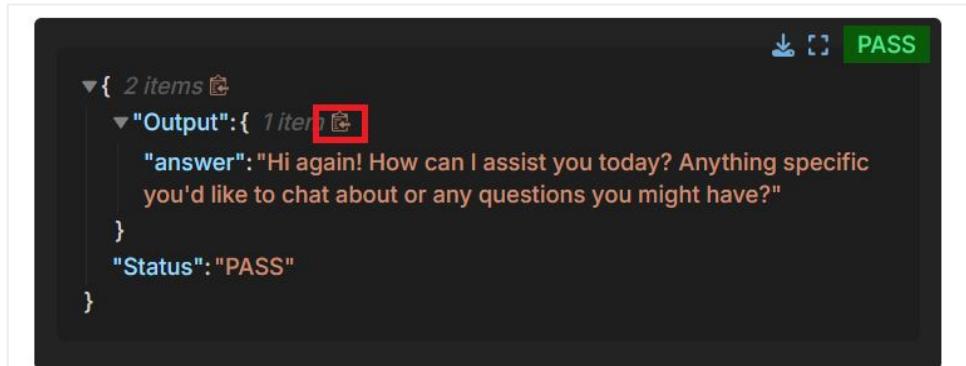
# Step 10 - Assign a Variable Name

After clicking the clipboard icon and seeing the popup window:

Enter a name for the variable in the input field (e.g., responseText or savedAnswer).

Make sure the name is descriptive enough so you can easily recognize its purpose later.

Then, click “Save” or confirm the action in the popup.



A screenshot of a JSON editor interface. At the top right, there are download and save icons, and a green "PASS" button. The JSON code shown is:

```
{ 2 items
  "Output": { 1 item
    "answer": "Hi again! How can I assist you today? Anything specific you'd like to chat about or any questions you might have?"
  }
  "Status": "PASS"
}
```



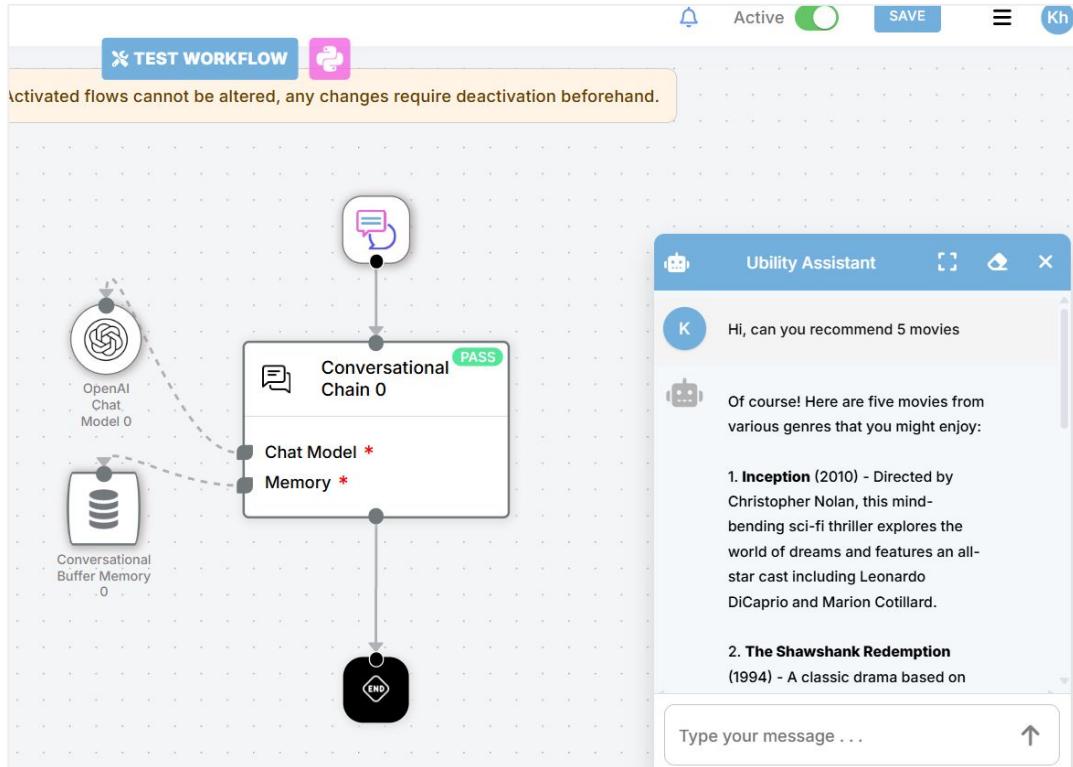
# Step 11: Confirm the Variable Assignment

You can see that the response now is assigned to the chatbotResponse

The screenshot shows a configuration interface for a "Chat Trigger". At the top, there's a toolbar with icons for Chat Trigger, Chat Variables, Help, Close, and Delete. Below the toolbar, the "Chat Variables" section contains the variable "ChatbotQuestion" with the value "Hello!". In the "Chat Node" section, a "Conversational Chain 0" is selected. A tooltip indicates that this node only supports selecting the variable name of the "Output" key from its "ChatbotResponse". In the "ChatbotResponse" section, the variable "response" is assigned. There are also small double-headed arrow icons between the Chat Node and ChatbotResponse sections, and between the ChatbotQuestion and ChatbotResponse sections.

# Step 12: Activate the Workflow

Finally, activate the workflow and start chatting with your assistant



# **Section 3: Quick start - Build Your Own Financial Expert Bot**

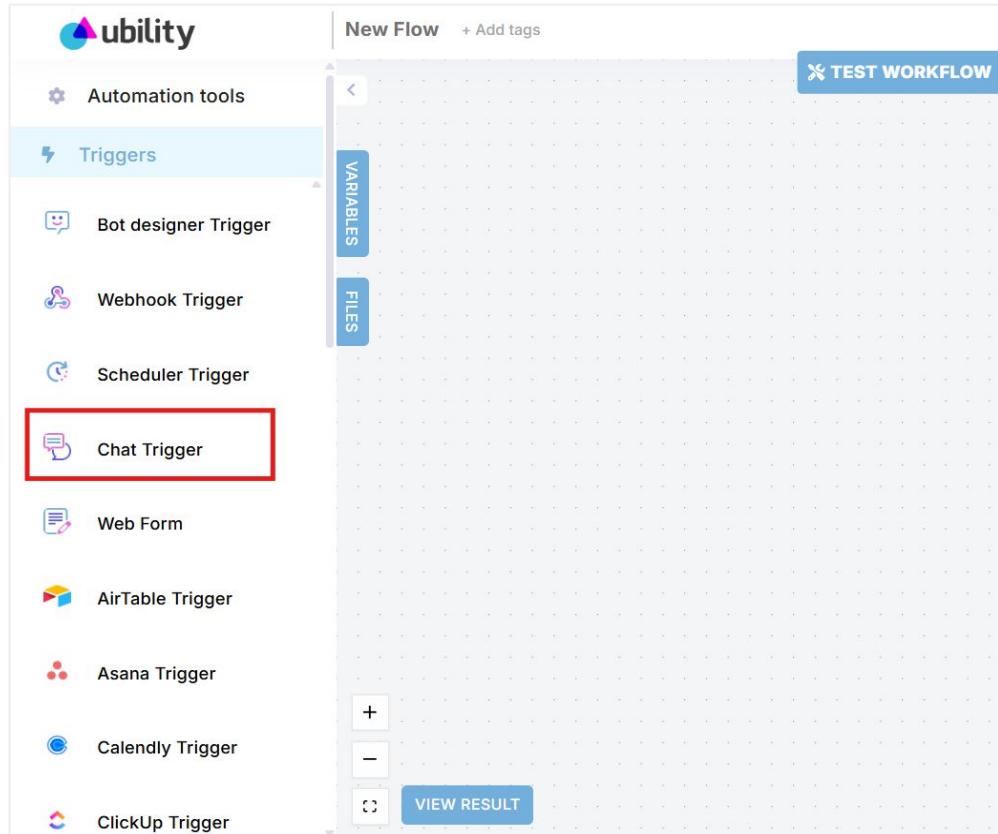
# What we are going to achieve:

By building our own Financial Expert Bot, we'll learn how to apply AI to create specialized, intelligent assistants using LangChain and prompt engineering.

We'll explore how to design effective prompts, capture financial questions from users, and deliver expert-like answers. This hands-on activity lays the foundation for developing custom AI solutions that can automate professional advice and support.

# Step 1 – Chat Trigger

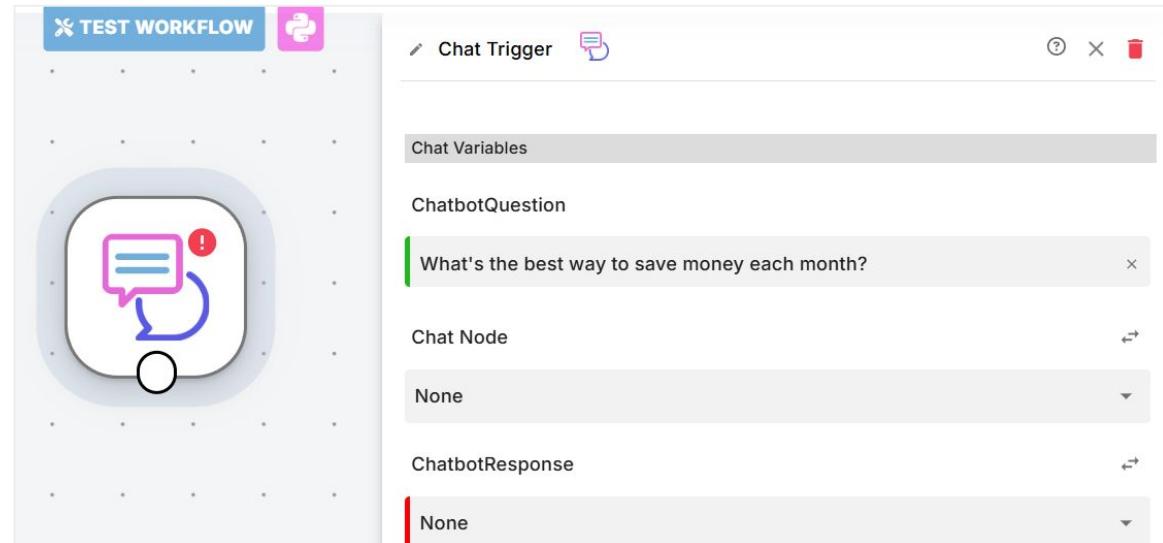
First, let's drag and drop a Chat Trigger into the designer



# Step 2 – Chat Trigger ChatbotQuestion

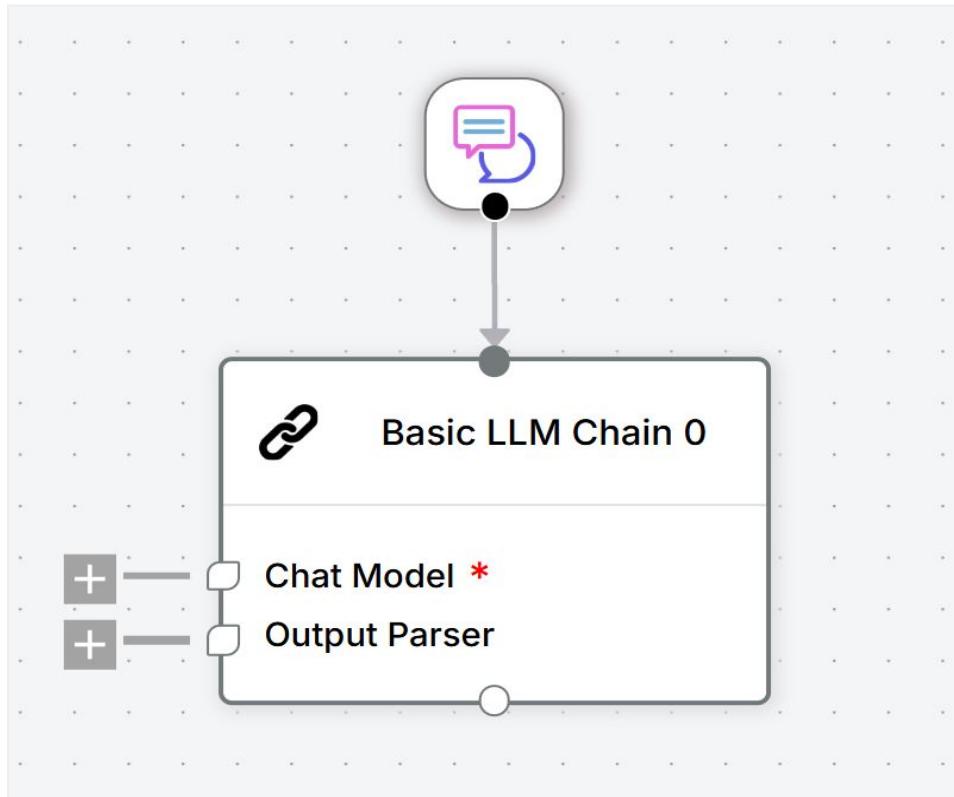
In the chatbotQuestion, let's ask a question like:

"What's the best way to save money each month?"



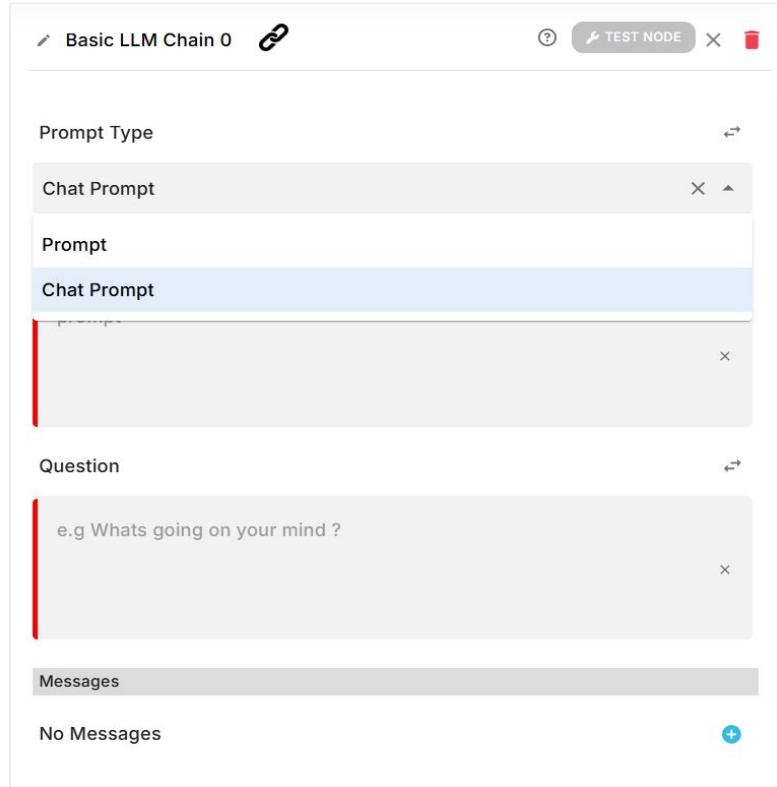
# Step 3 – Basic LLM Chain Node

Add a Basic LLM Chain Node,  
And connect it to the chat  
trigger



# Step 4 – Basic LLM Chain Node Question

In the Basic LLM Chain, we choose the prompt type as a ‘Chat Prompt’. This option allows us to write a specific prompt for the Basic chain.



# What are Prompts?

## Prompts

Prompts are instructions you give the language model to guide its behavior, tone, and responses, helping it produce more accurate and relevant outputs.

Good prompts = Better results.

Prompt Example:

"Act as a financial advisor and answer this query: \${user\_question}"

# Step 5 – Basic LLM Chain Template

In the prompt field (Template), enter:

"You are a certified financial advisor. Give a smart and realistic answer to my question"

And in the question field, choose the default question coming from the chatbox.  
(The one in the chatbotQuestion, chat trigger)

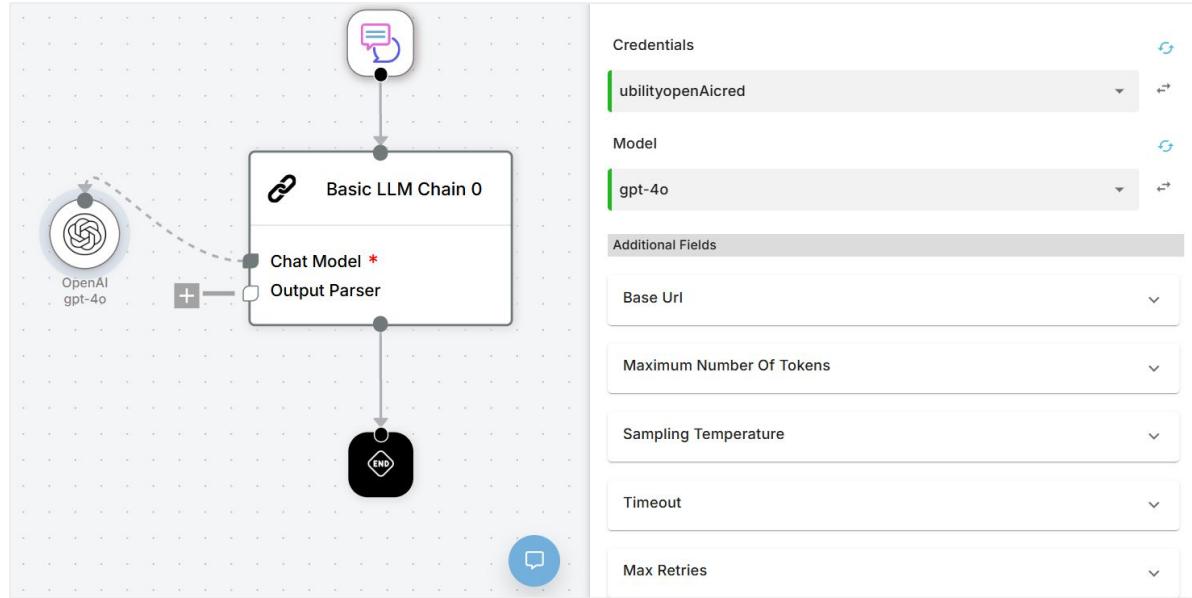
The screenshot shows the ubility platform's configuration interface for a "Basic LLM Chain 0". The interface is divided into several sections:

- Prompt Type:** Chat Prompt
- Template:** You are a certified financial advisor. Give a smart and realistic answer to my question.
- Question:** \${ChatbotQuestion}
- Messages:** No Messages

At the top right, there are buttons for "TEST NODE", a question mark icon, and a trash bin icon.

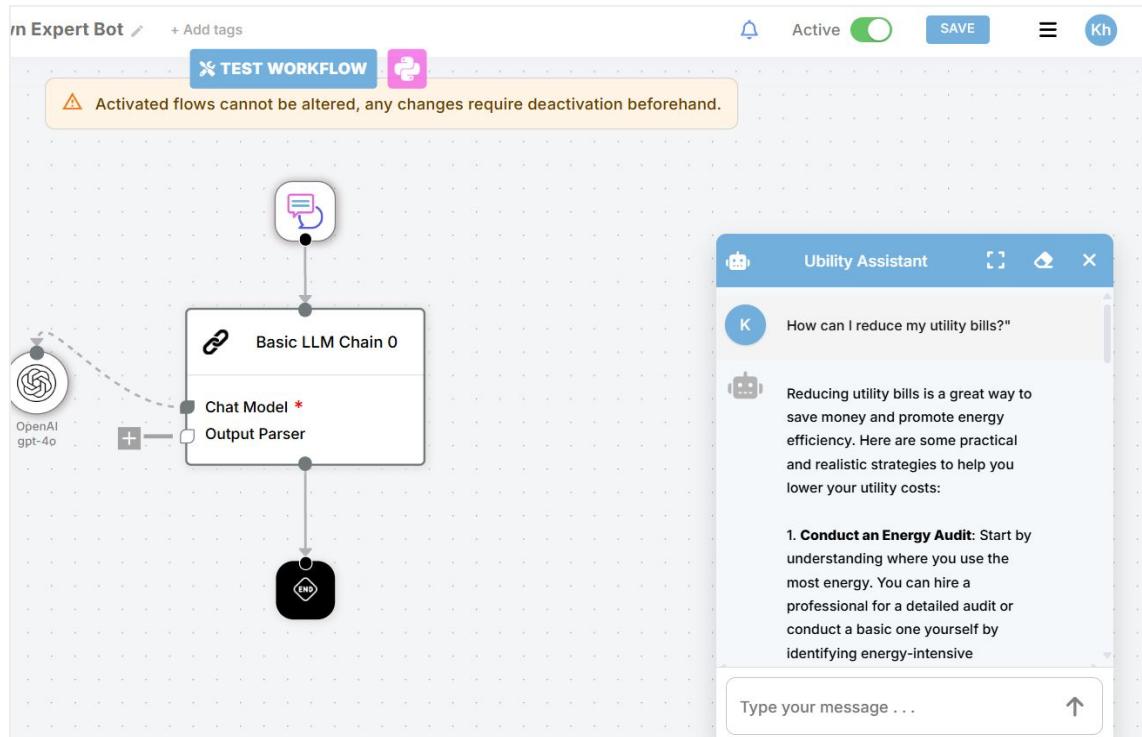
# Step 6 – Basic LLM Chain Sub-node

For the Chat model, we choose OpenAi chat model gpt-4o



# Step 7 – Flow Activation

Finally, activate the workflow and start chatting with your financial expert bot.



# Sharpen Your Prompting Skills

## Bonus Challenge

Enhance your prompt to make the AI's responses either more concise or more creative.

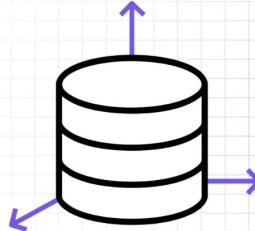
You can guide the AI with specific instructions, for example:

1. Respond in a funny yet informative tone.
1. Summarize the answer in 3 bullet points

# **Section 4: Quick start - Vector Database & RAG (Retrieval-Augmented Generation)**

# What is a Vector Database?

A Vector Database stores embeddings, which are numerical representations of text, images, or any content. These embeddings help AI models understand semantic meaning, so instead of looking for exact words, they look for similar ideas or concepts.



Understanding  
**Vector Database**

# Using Utility Vector DB for Smarter AI Search and Context

In Utility, you can use Utility Vector DB to:

- Store documents as vectors
- Search semantically, not just by keywords
- Power up AI tools with more context



# What is RAG (Retrieval-Augmented Generation)?

RAG is a technique that combines retrieval + generation.

Here's how it works:

You ask a question like:

"What's the main idea of the Agile Manifesto?"

The system retrieves relevant documents from the vector database (based on embeddings).

Then the AI generates a response using both the question and the retrieved documents.

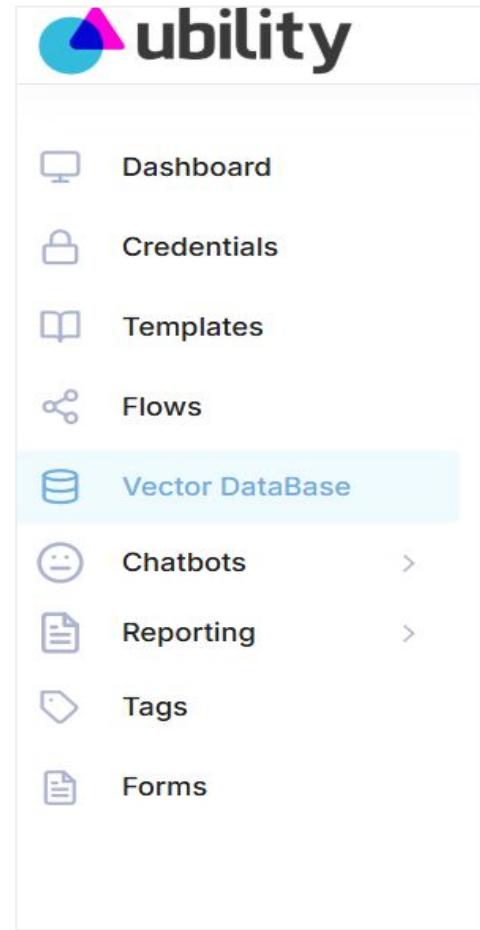


# How RAG Works in Ubility?

# Step 1 – Vector Database

To implement RAG in Utility:

1. Head to Utility Vector DataBase



# Step 2 – Indexes

To implement RAG in Utility:

2- Create a new index

The screenshot shows a user interface for managing indexes. At the top left, there is a blue circular icon with a white letter 'C'. To its right, the word 'Indexes' is displayed in a large, bold, black font. On the far right, a light gray box contains the text 'Total: 4' in a black font. In the bottom right corner of this box, there is a blue button with a white plus sign and the text 'New Index'. This 'New Index' button is highlighted with a thick red rectangular border.

# Step 3 – Create Index and Generate Embeddings

To implement RAG in Ubility:

3- Fill the necessary data to create this index + Use Embeddings Node (e.g., OpenAI or Google) to convert text into vectors.

Create Index

Index Name: isdb

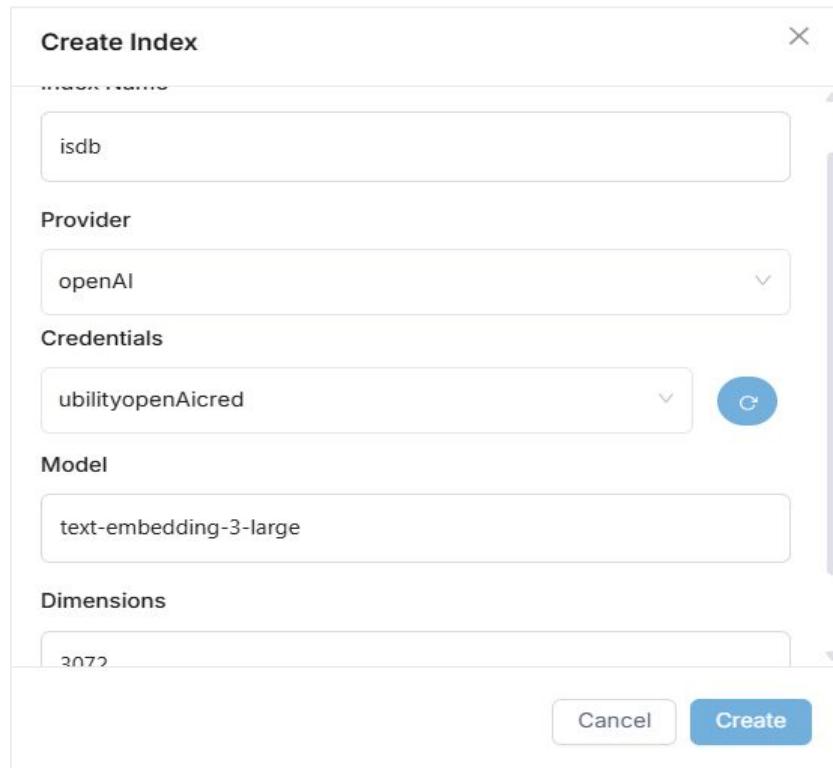
Provider: openAI

Credentials: utilityopenAicred

Model: text-embedding-3-large

Dimensions: 3072

Cancel Create



# Step 4- Data

To implement RAG in Utility:

4- Once created, you can start to insert data.

Index: isdb / Documents

+ Insert Data

# Step 5 - Load Data and Choose Format

## To implement RAG in Ubility:

5- Provide a description, if you have the document locally on your laptop choose the "Basic Data Loader".

if the data is a url choose "Webpage Loader".

6- For the data form, you have several options: pdf, text, csv and json

Insert Data X

Description  
IsDB Indonesia MCPS\_Final

Loader  
Basic Data Loader

Data Type  
PDF

Data From  
Upload file

IsDB Indonesia MCPS\_Final.pdf

> Chunk Size

Cancel Create

# Step 6 - Data Stored in Utility Vector Database

To implement RAG in Utility:

7- Now you have data in your Utility Vector Store

Index: isdb / Documents

[Go Back](#) [Insert Data](#)

Search here  Total: 2

**Provider:** openAi  
**Model:** text-embedding-3-large

<b>Description:</b> IsDB Indonesia MCPS_Final <b>Source:</b> IsDB Indonesia MCPS_Final.pdf	<b>Description:</b> IsDB Bangladesh MCPS <b>Source:</b> IsDB Bangladesh MCPS.pdf
---	---

1 day ago 1 day ago

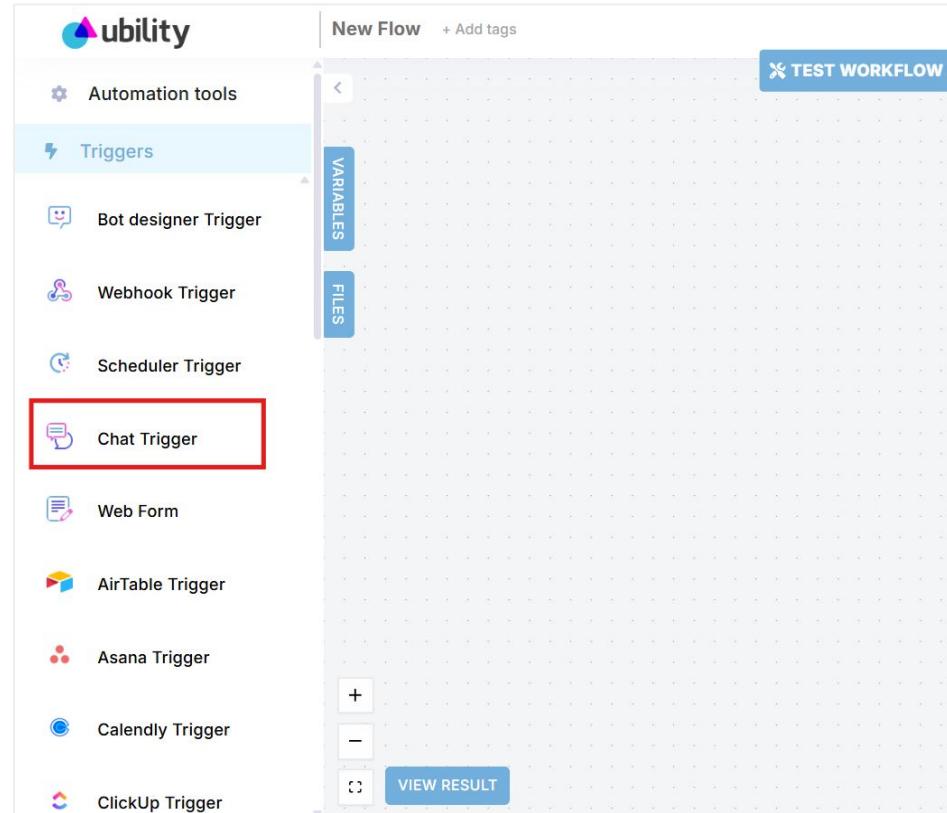
 



**Now let's implement a simple example on  
how we can use Vector Store and RAG!**

# Step 7 - Chat Trigger

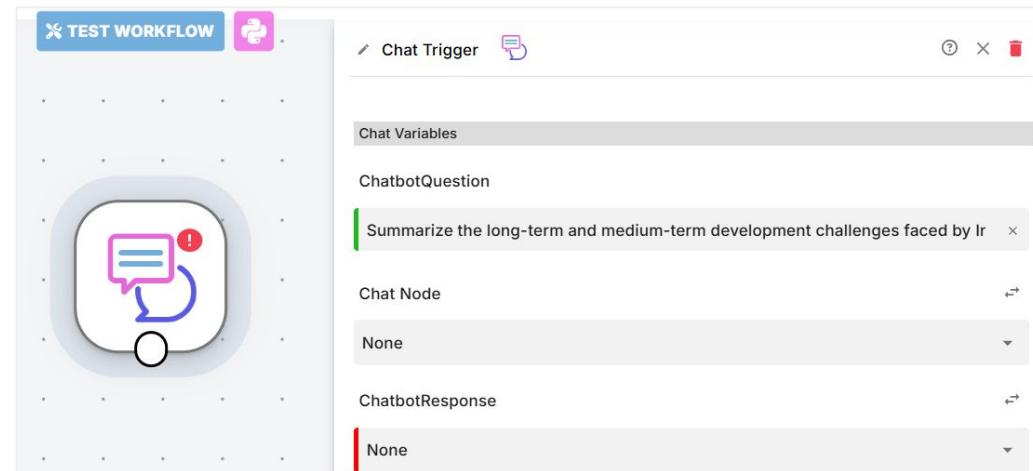
First, let's drag and drop a Chat Trigger into the designer



# Step 8 – Chat Trigger ChatbotQuestion

In the chatbotQuestion, let's ask a question related to the documents we inserted earlier in Utility Vector Store.

Example: "Summarize the long-term and medium-term development challenges faced by Indonesia including demographic characteristics and infrastructure constraints"

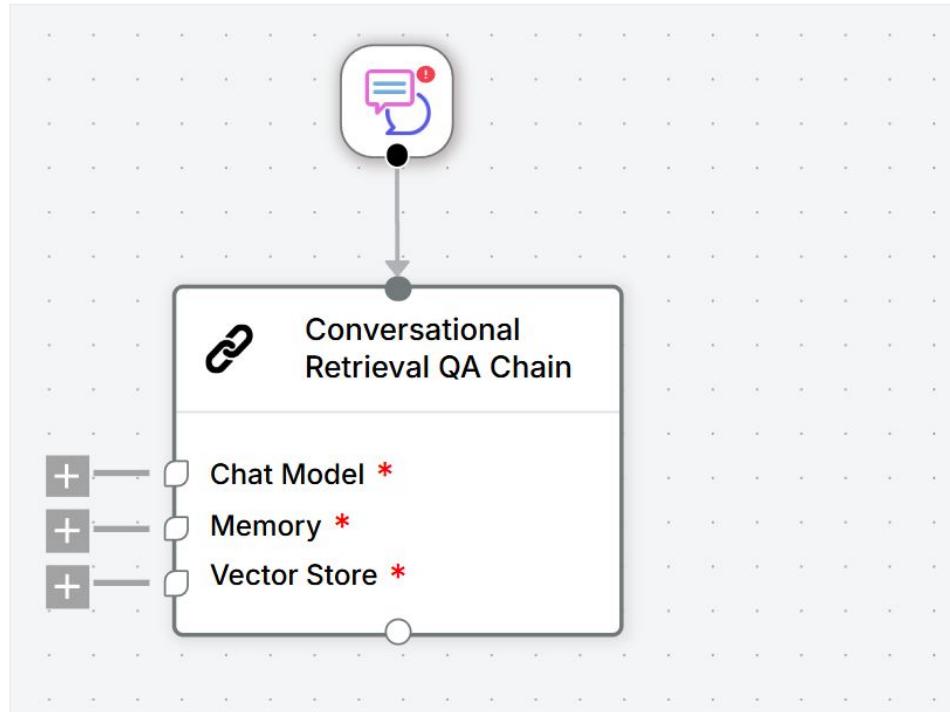


# Step 9 - Add and Connect QA Chain

Add a Conversational Retrieval QA Chain Node,  
And connect it to the chat trigger.

All QA Chains in Ubility are RAG  
(Retrieval-Augmented Generation)  
chains.

They rely on a Vector Database  
(VDB) to retrieve relevant information  
and generate accurate answers.



# Step 10 - Conversational Chain Node Question

In the question field, choose the default question coming from the chatbox.  
(The one in the chatbotQuestion, chat trigger)

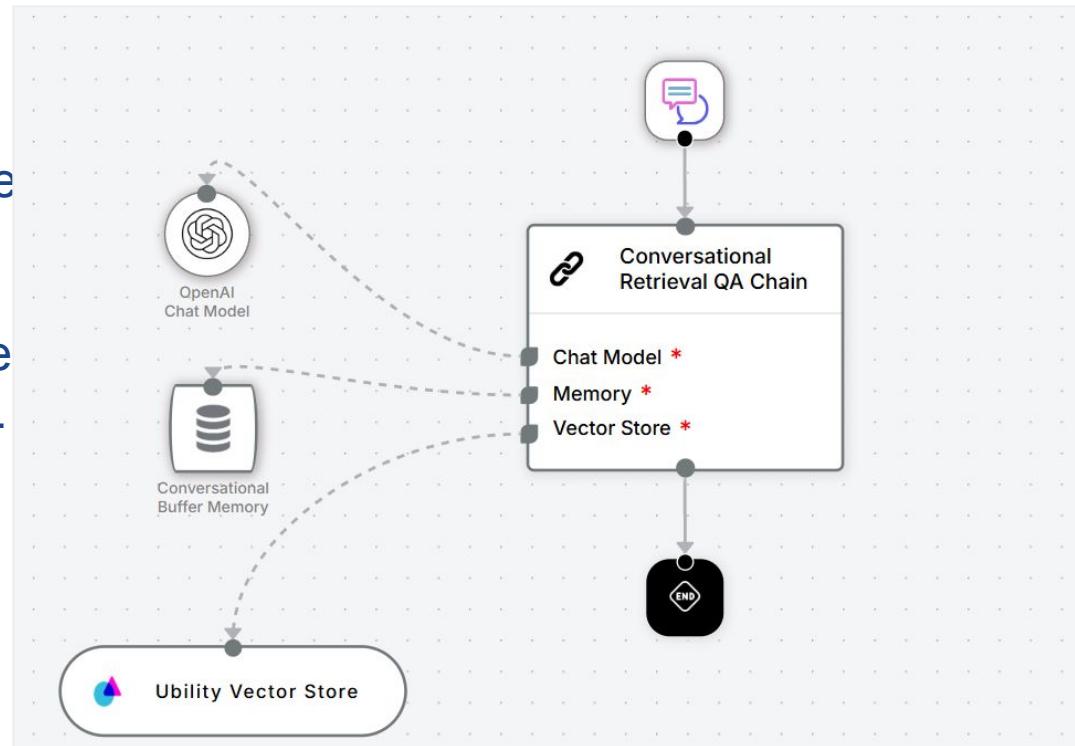
The screenshot shows the configuration interface for a 'Conversational Retrieval QA Chain' node. The 'Question' field contains the placeholder `\${ChatbotQuestion}`. Below it are sections for 'Rephrase Prompt' and 'Response Prompt', both currently collapsed. At the bottom, the 'Default Output' section is expanded, showing a JSON object with three items:

```
{ 3 items
  "Error": ""
  "Output": { 1 item
    "answer": ""
  }
  "Status": ""
}
```

# Step 11 - Set Up Model, Memory, and Store

For the Chat model, we choose OpenAi chat model gpt-4o.

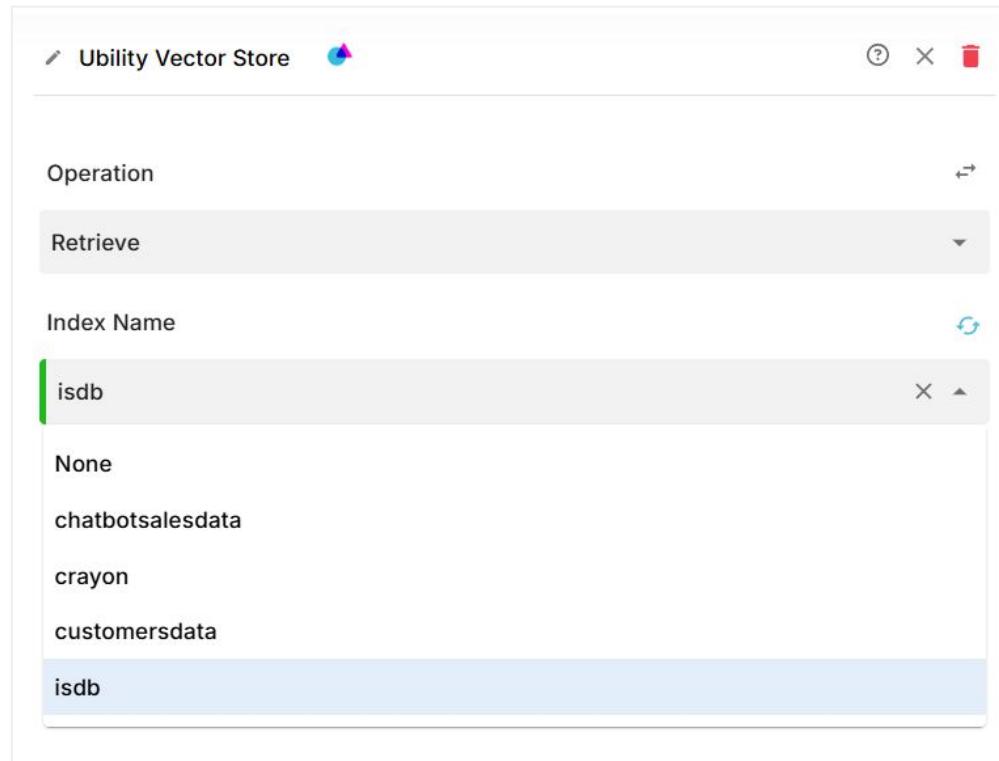
For the Memory we choose the Conversational Buffer Memory. And for the Vector Store we choose Ubility Vector Store.



# Step 12 – Select Retrieve Operation and Index

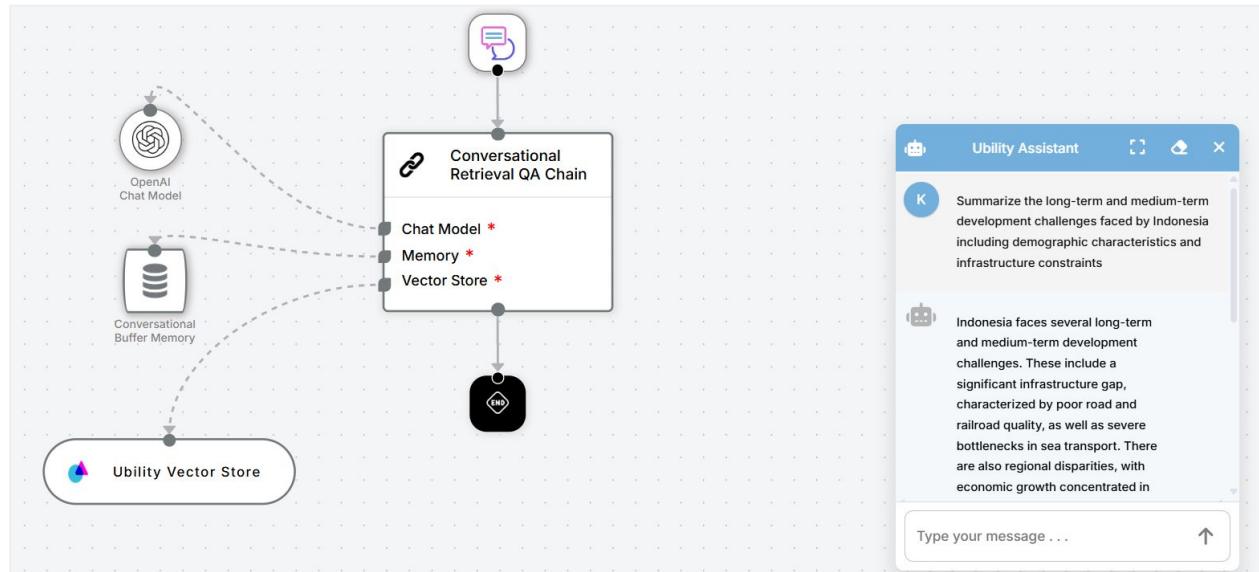
In the vector store, we choose the ‘Retrieve’ operation as we are retrieving data.

For the index name we choose the index we want to retrieve data from. In this example it's isdb.



# Step 13 – Activate and Ask Questions

Finally, activate the workflow and start asking questions related to your documents!

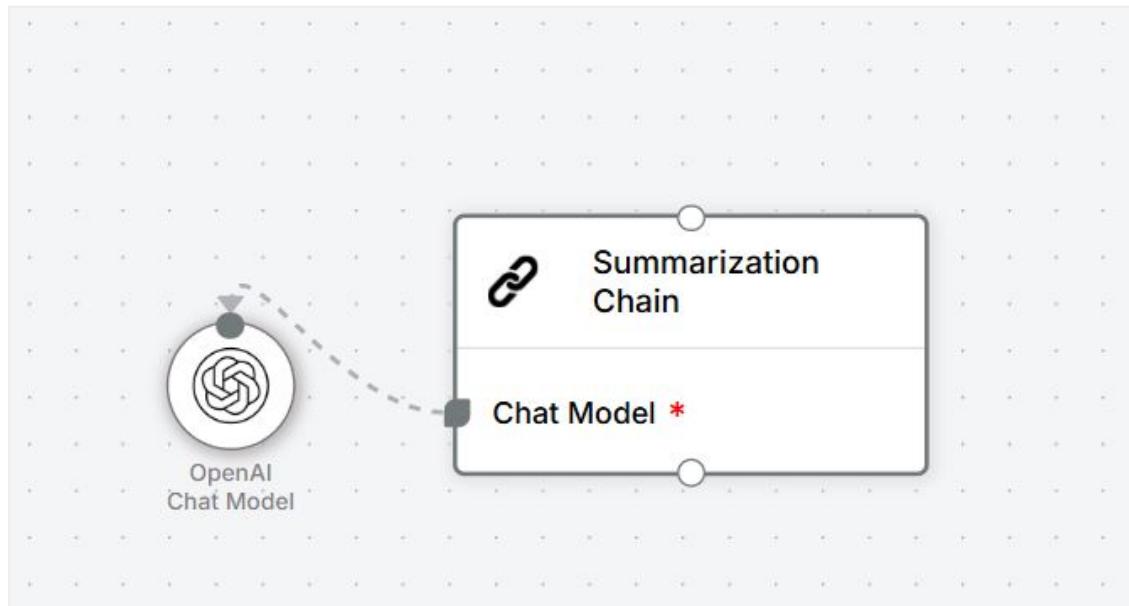


# **Section 4: Quick start - Summarization Chain And Compare Documents Chain**

# Step 1 – Set Up Summarization Chain

## Summarization Chain:

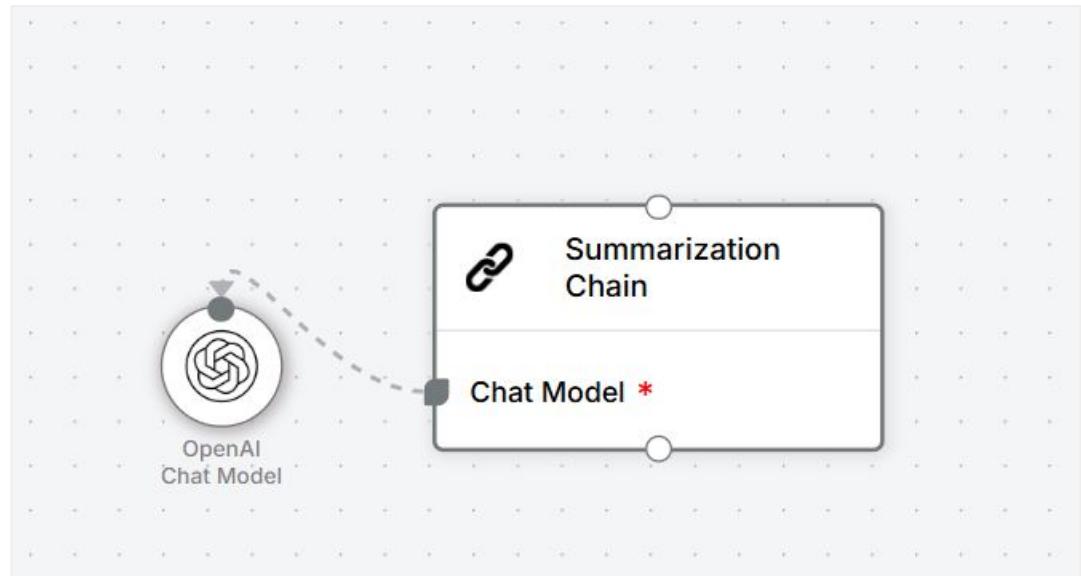
The Summarization Chain is a LangChain node in Utility that takes long pieces of text or documents and generates a short, meaningful summary.



# Step2 - Summarize Key Points

## Summarization Chain:

It's useful when you want to quickly understand the key points of a large document without reading everything.



# Step 3 - Summarization chain Data

## How it works:

1. You choose the data form, whether it's binary data or URL.
1. If you choose URL, provide the URL you want to summarize.
1. Then choose the summarization method.

The screenshot shows a user interface for a 'Summarization Chain' node. At the top right, there are buttons for '? TEST NODE', a refresh icon, and a trash bin icon. The main area is titled 'Data Form' and contains several input fields:

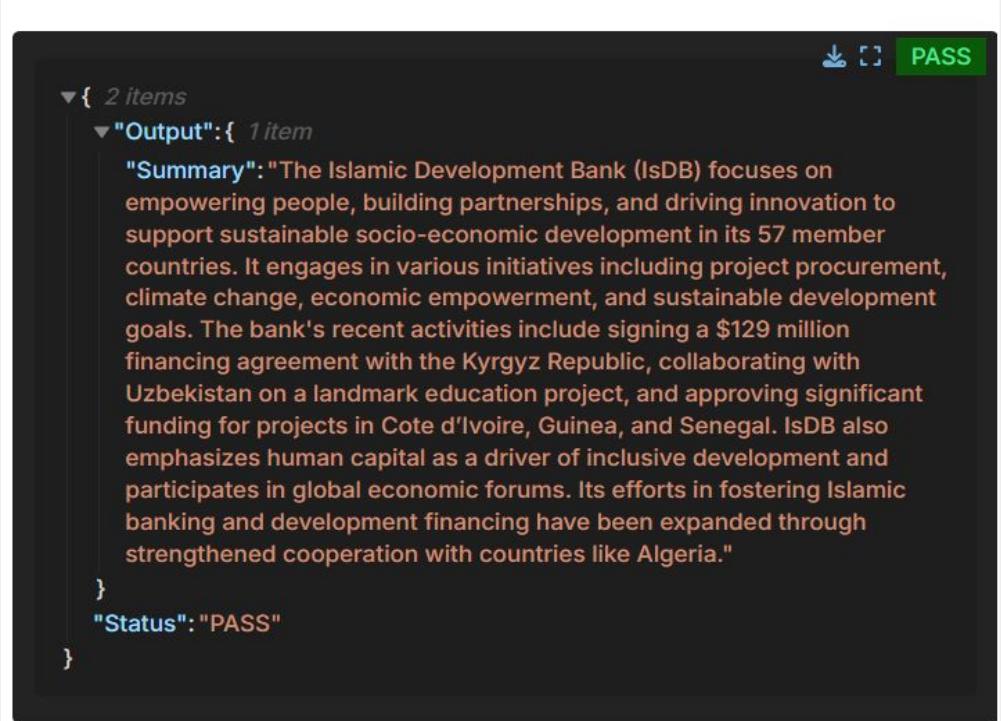
- 'URL': A dropdown menu currently set to 'URL'. Below it, a text input field contains the URL 'https://www.isdb.org/'.
- 'Data': A dropdown menu currently set to 'Data'.
- 'Summarization Method': A dropdown menu currently set to 'Stuff'.
- 'Prompt': A dropdown menu currently set to 'Prompt'.
- 'Text Splitter': A dropdown menu currently set to 'Text Splitter'.
- 'Chunk Size': A dropdown menu currently set to 'Chunk Size'.
- 'Chunk Overlap': A dropdown menu currently set to 'Chunk Overlap'.

Each input field has a small double-headed arrow icon to its right, indicating they can be expanded or collapsed.

# Step 4 - Summarize Documents and Support

## When to use:

1. Summarizing research papers or reports.
1. Creating short summaries for customer service tickets or support emails.
1. Reducing document size for dashboards or quick reviews.



A screenshot of a JSON editor interface. At the top right, there are download, copy, and 'PASS' buttons. The JSON code is displayed below:

```
{ "Output": { "Summary": "The Islamic Development Bank (IsDB) focuses on empowering people, building partnerships, and driving innovation to support sustainable socio-economic development in its 57 member countries. It engages in various initiatives including project procurement, climate change, economic empowerment, and sustainable development goals. The bank's recent activities include signing a $129 million financing agreement with the Kyrgyz Republic, collaborating with Uzbekistan on a landmark education project, and approving significant funding for projects in Cote d'Ivoire, Guinea, and Senegal. IsDB also emphasizes human capital as a driver of inclusive development and participates in global economic forums. Its efforts in fostering Islamic banking and development financing have been expanded through strengthened cooperation with countries like Algeria." }, "Status": "PASS" }
```

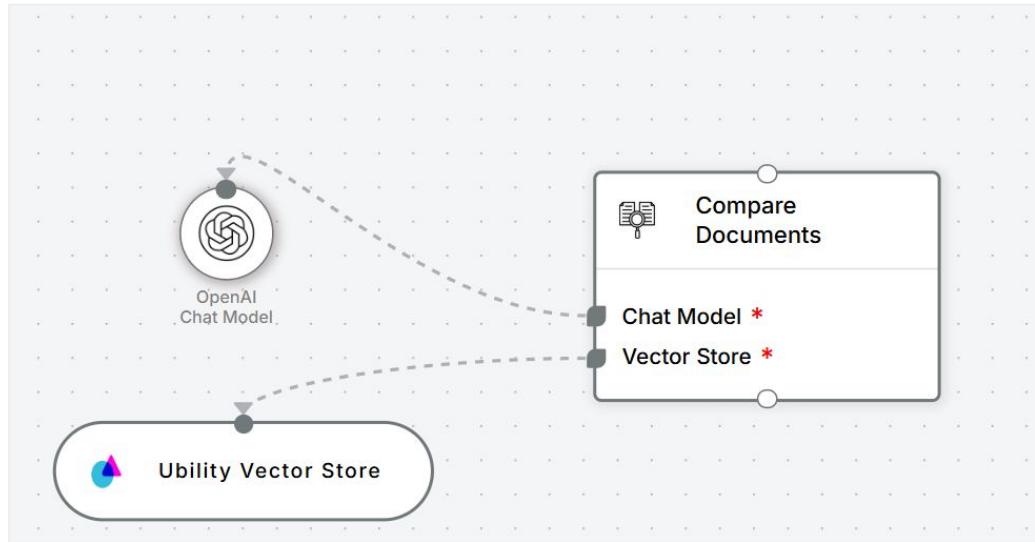
# Step 5 - Compare Documents

## Compare Documents Chain:

### What is it?

The Compare Documents Chain is used to find similarities or differences between two or more documents.

It uses AI to read and understand each document, and then highlights:  
Key similarities,  
Key differences and  
Unique parts of each document



# Step 6- Input Documents and Compare

## How it works:

1. You input two or more documents (pdf, word files, etc.). Those files should be already inserted in the vector database you are using.
1. The chain analyzes the content using an LLM.
1. It outputs a comparative summary, side-by-side insights, or highlights.

The screenshot shows a user interface titled "Compare Documents". At the top right, there are buttons for "TEST NODE", a help icon, and a close button. The main area has a "Question" section containing the text "Compare IsDB Indonesia MCPS\_Final document with IsDB Bangladesh MCPS document". Below this is a "Files" section with a header "Files (2)". Under "File Name 1", the text "IsDB Indonesia MCPS\_Final" is listed. Under "File Name 2", the text "IsDB Bangladesh MCPS" is listed. Each file entry has a green plus sign icon to its right and a red minus sign icon to its right, indicating the ability to add or remove files.

# Step 7 - Use Cases for Document Comparison

## When to use:

1. Compare legal contracts.
1. Reviewing two versions of a document (before/after edits).
1. Checking different vendor proposals or policy documents.

```
▼ { 2 items
  ▼ "Output": { 2 items
    "input": "Compare IsDB Indonesia MCPS_Fi..."
    "output": "The comparison between the IsDB Indonesia MCPS (2022-2025) and the IsDB Bangladesh MCPS (2024-2026) documents highlights their strategic approaches and alignment with national priorities. Here's a detailed comparison: 1. **Timeframe and Alignment**: - **Indonesia**: The MCPS covers 2022 to 2025, aligning with the National Medium-Term Development Plan (RPJMN). - **Bangladesh**: It spans 2024 to 2026, aligned with the Eighth Five Year Plan (FYP), Vision 2041, and Delta Plan 2100. 2. **Foundational Studies and Analyses**: - **Indonesia**: Based on four studies including Country Portfolio Review Note (CPRN), Country Diagnostic Study (CDS), Sector, Thematic & Fiduciary Analyses, and Donor Coordination Matrix (DCM). - **Bangladesh**: Informed by a Country Diagnostic Study (CDS), CPRN by the IsDB Regional Hub of Dhaka, and government meeting minutes. 3. **Strategic Objectives and Themes**: - **Indonesia**: Focuses on reducing regional disparities, deepening the Islamic financial industry, and regional integration. Driven by IsDB's Strategic Realignment. - **Bangladesh**: Builds on lessons from past MCPS (2013-2018), focusing on pillars aligned with national priorities. 4. **Implementation and Oversight**: - **Indonesia**: Involves ongoing projects and a Country Portfolio Review Note. - **Bangladesh**: Overseen by a steering committee of the Regional Hub of Dhaka and the government. 5. **Financial Aspects**: - **Indonesia**: The financing envelope was adjusted from US$ 5.2 billion to US$ 2.24 billion with a 92% achievement rate by the end. - **Bangladesh**: Specific financial details are not provided, but align with strategic goals. Both documents reflect the IsDB's strategic partnership with Indonesia and Bangladesh, tailored to each country's development needs and priorities, while integrating lessons from previous engagements."
  }
```

# **Section 5: Quick start - AI Agent with Tools in Utility**

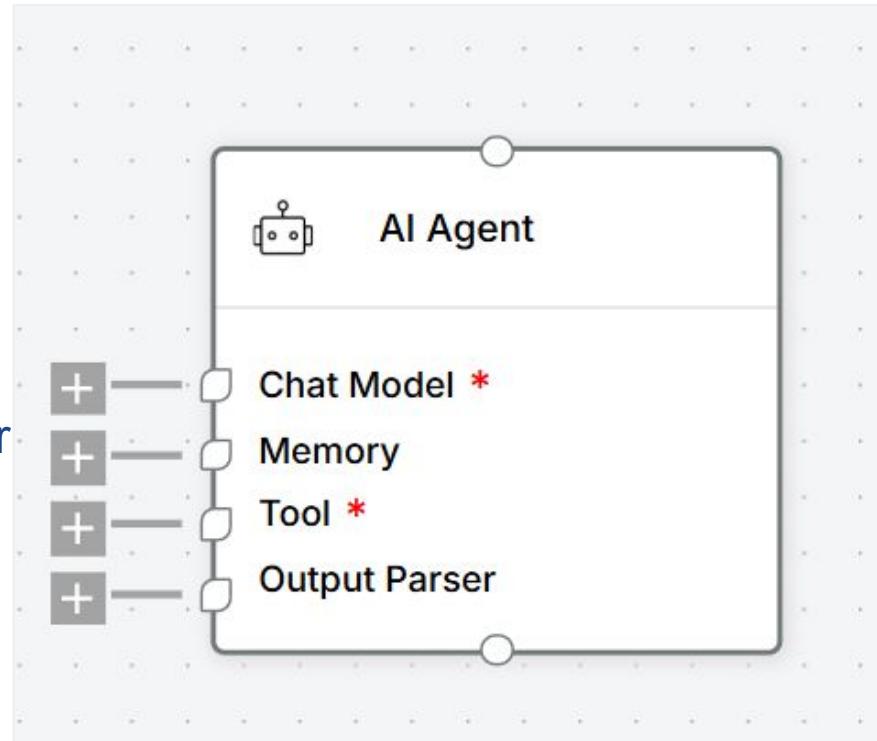
# What is an AI Agent:

An AI Agent in Utility is like a smart assistant that can think, decide, and use tools to complete a task.

Unlike regular chains that follow a straight path, the AI Agent is dynamic.

It analyzes the user's input, chooses the right tool(s), and uses them in the right order autonomously.

It's designed for multi-step, flexible workflows where the answer might require using more than one tool or data source.



# What are tools?

Tools are extra abilities you can give to the AI Agent. Think of them like plugins.

Examples of Tools available in Utility:

- **Google Search Tool:** Get real-time answers from the web.
- **Calculator Tool:** Solve math problems.
- **Wikipedia Tool:** Pull knowledge from Wikipedia.
- **Custom Code Tool:** Run your own python or javascript logic.
- **SerpAPI Tool:** Search results from Google via API.

The screenshot shows a sidebar titled 'Tools' with a search bar at the top. Below the search bar are six tool cards, each with an icon, a name, and a brief description. The tools listed are: Google Search (useful for real-time access to Google search results), Calculator (makes it easier for AI agents to perform arithmetic), Custom Code Tool (allows writing tools in JS or Python), SerpApi (useful for real user search results from major engines), Wikipedia (for searching in Wikipedia), and Custom Tool (uses another utility workflow as a tool).

Tool	Description
Google Search	Useful to get real-time access to Google search results
Calculator	Make it easier for AI agents to perform arithmetic
Custom Code Tool	Write a tool in JS or Python
SerpApi	Useful to get real user search results from all major search engines
Wikipedia	Search In Wikipedia
Custom Tool	Uses another utility workflow as a tool.

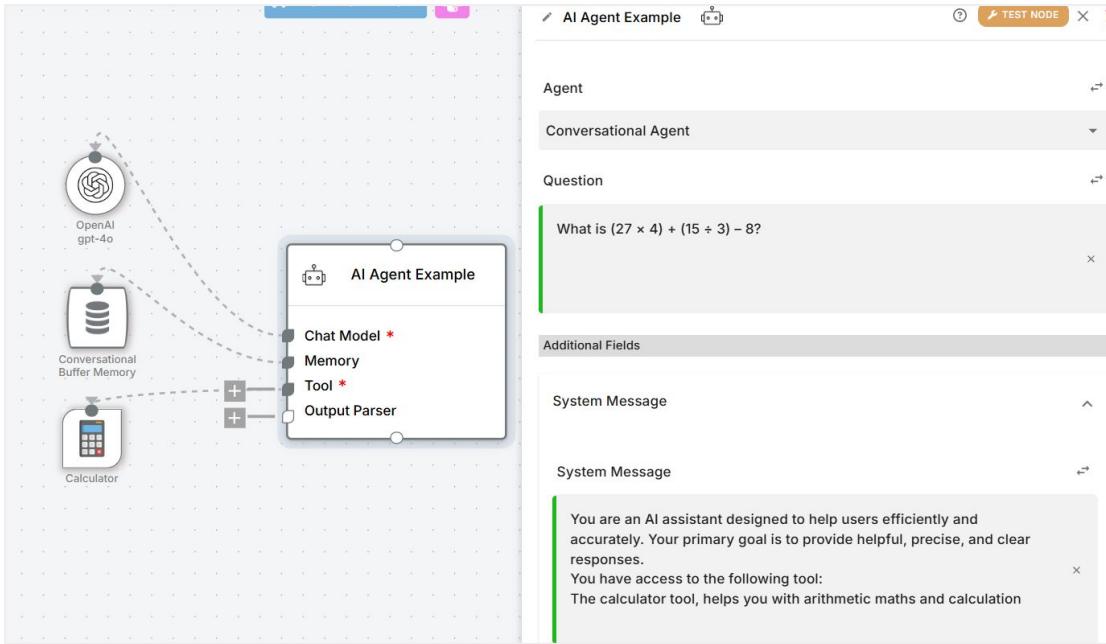
# Example: Solving Math with AI

## Example Use Case:

Let's say a user asks:

'What is  $(27 \times 4) + (15 \div 3) - 8$ ?'

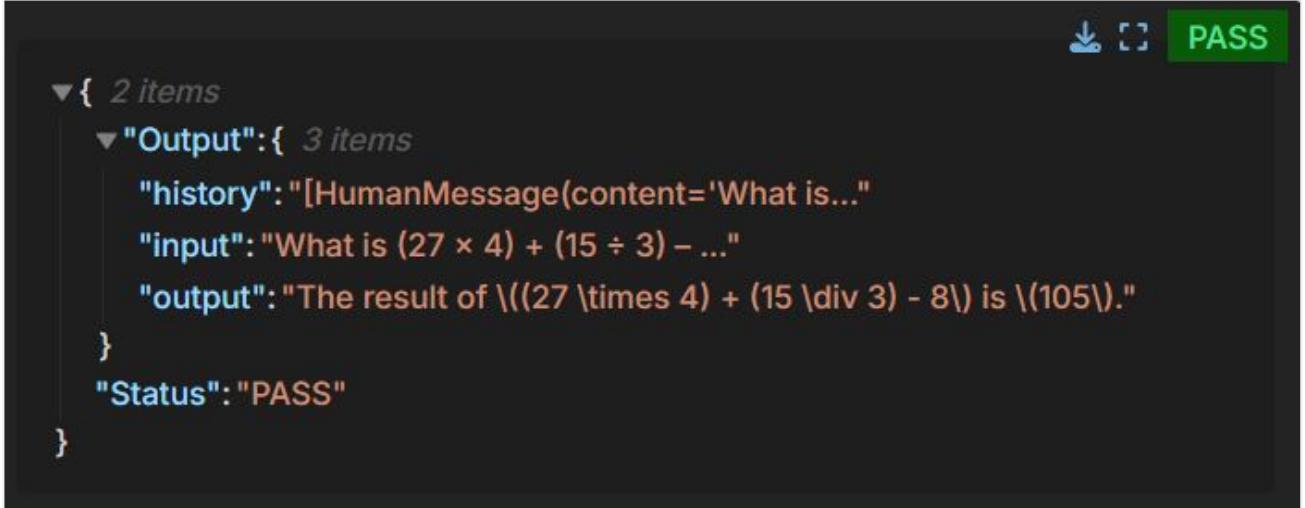
The AI Agent will:  
Use the calculator tool to compute  
and calculate this math problem.



# Results

## Example Use Case:

Then it will return a comprehensive answer, all in one workflow.



A screenshot of a software interface showing a test result. The top right corner features a green button labeled "PASS" with a checkmark icon. To its left are icons for download, refresh, and search. The main area displays a JSON-like data structure representing the test output:

```
▼{ 2 items
  ▼"Output":{ 3 items
    "history": "[HumanMessage(content='What is...')"
    "input": "What is (27 × 4) + (15 ÷ 3) - ..."
    "output": "The result of \((27 \times 4) + (15 \div 3) - 8\)\ is \((105)\)."
  }
  "Status": "PASS"
}
```

**Here are some resources to help you better understand the Ubility platform:**

- Ubility Documentation (still beta version): [Ubility Documentation](#)
- Ubility Website: [Ubility](#)