

# Developing web-based AI applications for social sciences

Dr. Moritz Joerling

# Agenda/Overview

- Use Cases
- App architecture and implementation with JavaScript
- App architecture and implementation with Python
- Retrieval Augmented Generation

# Some use cases

- [Interview bot](#)
- [Chatbot intervention](#) (e.g., [can chatbots reduce conspiracy beliefs](#))
- Observing individuals' interaction with generative AI
- ...

Please participate in an interview about music. [Make sure your speakers are on so you can hear the questions.](#) The 'Continue' button will become available after 10 minutes, but you may continue the interview for as long as you like.

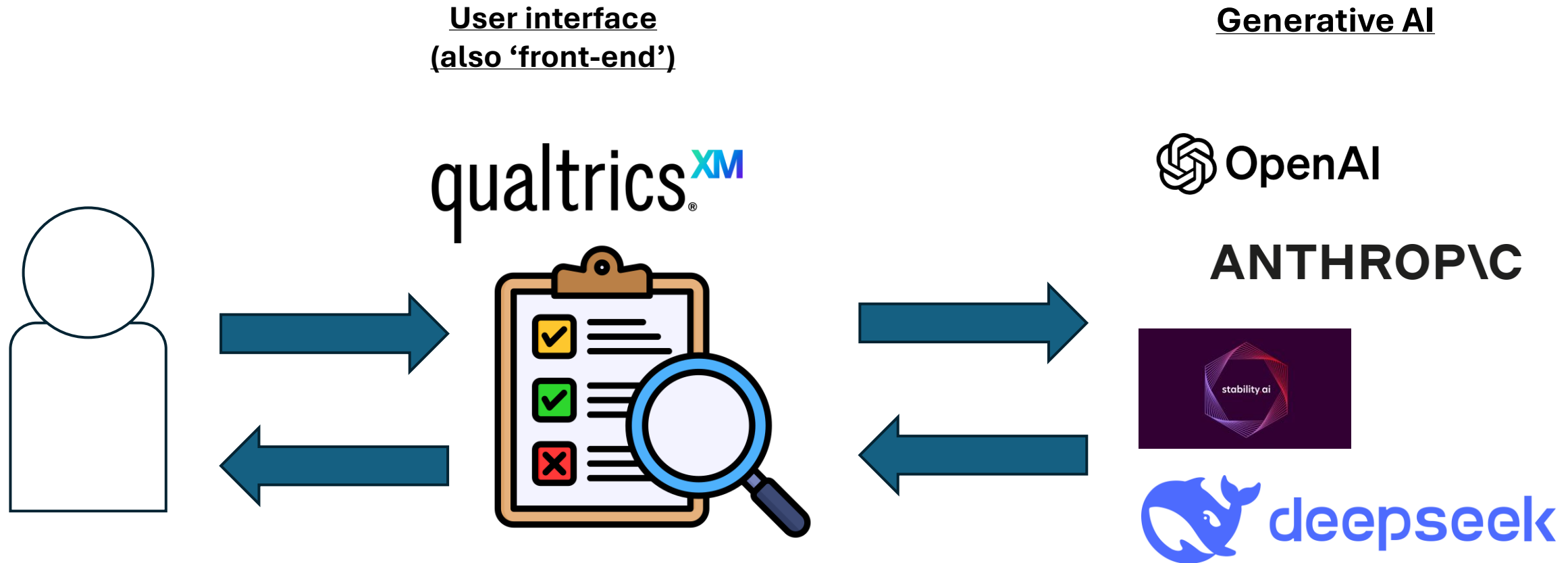
Assistant is speaking \*\*\*

Type your response here...

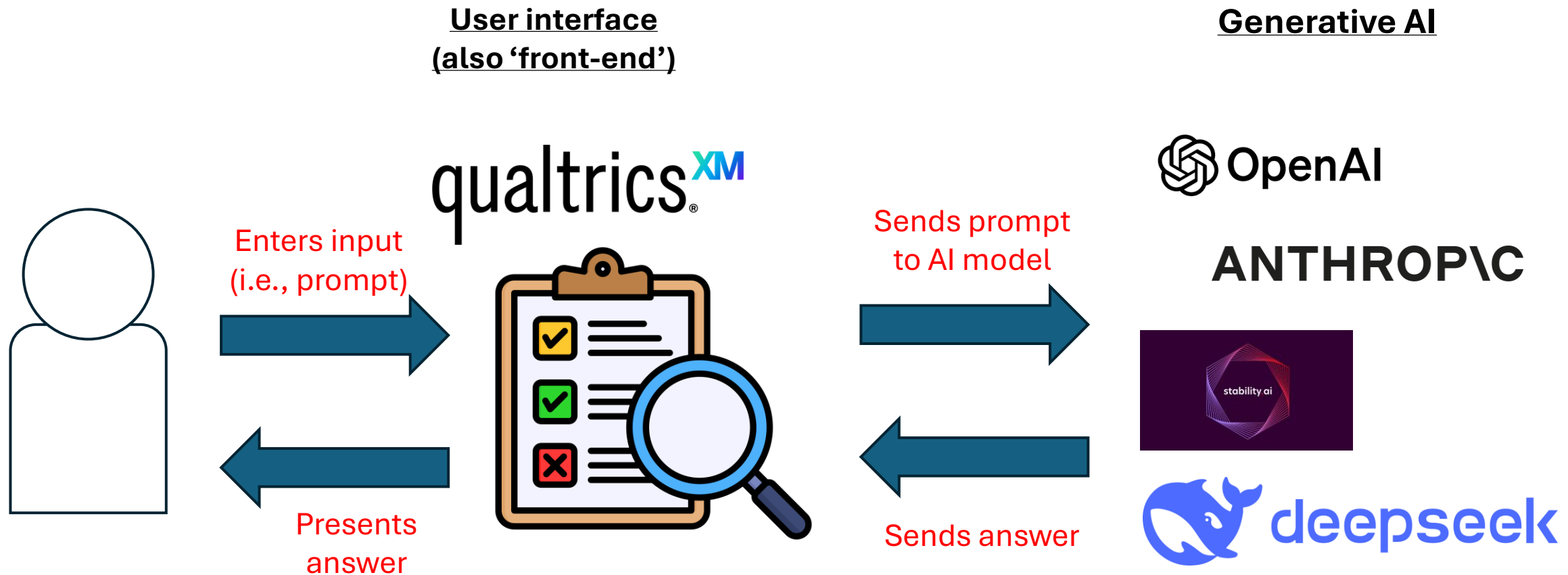
Submit Response



# The basic architecture of a web-based GenAI app

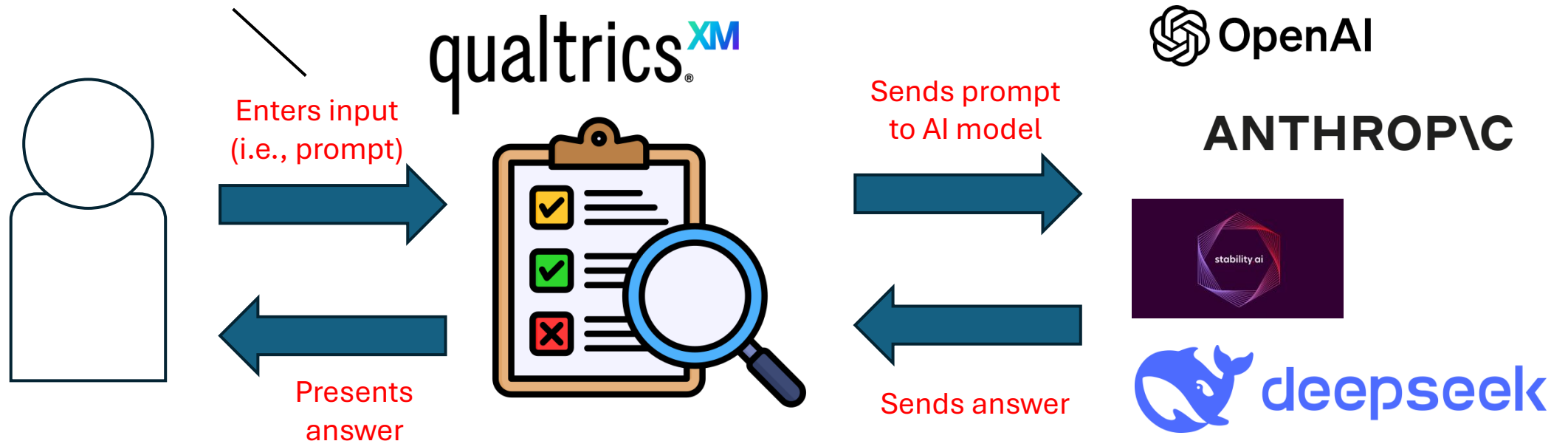


# The basic architecture of a web-based GenAI app



# Examples of what to investigate and manipulate?

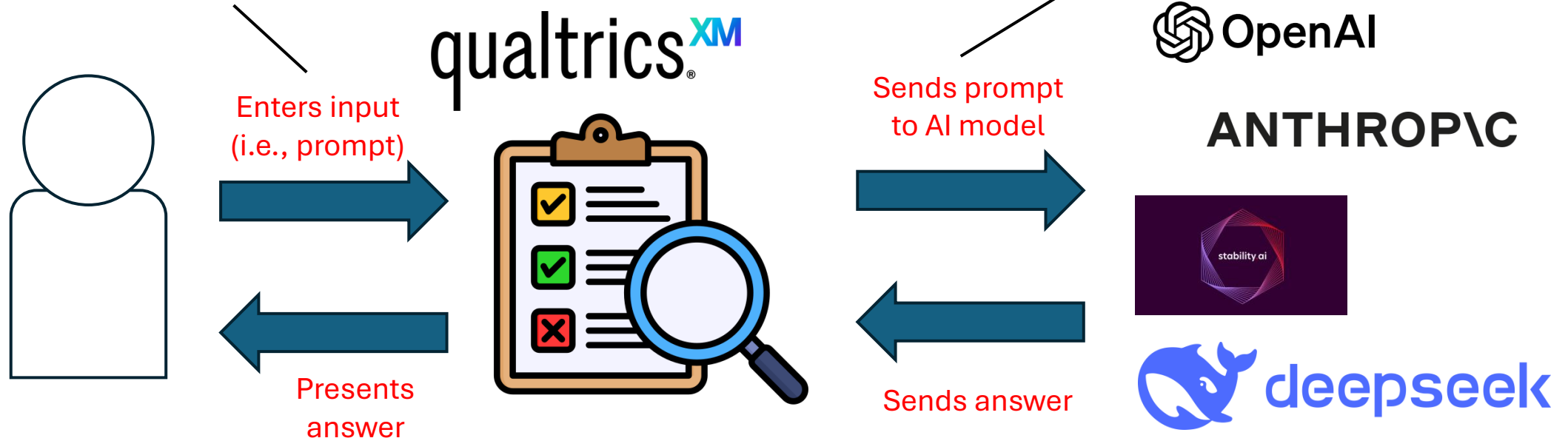
- How does the modality of data input affect people's interaction with generative AI?
- How does the input length affect the people's satisfaction or behavior with AI?



# Examples of what to investigate and manipulate?

- How does the modality of data input affect people's interaction with generative AI?
- How does the input length affect the people's satisfaction or behavior with AI?

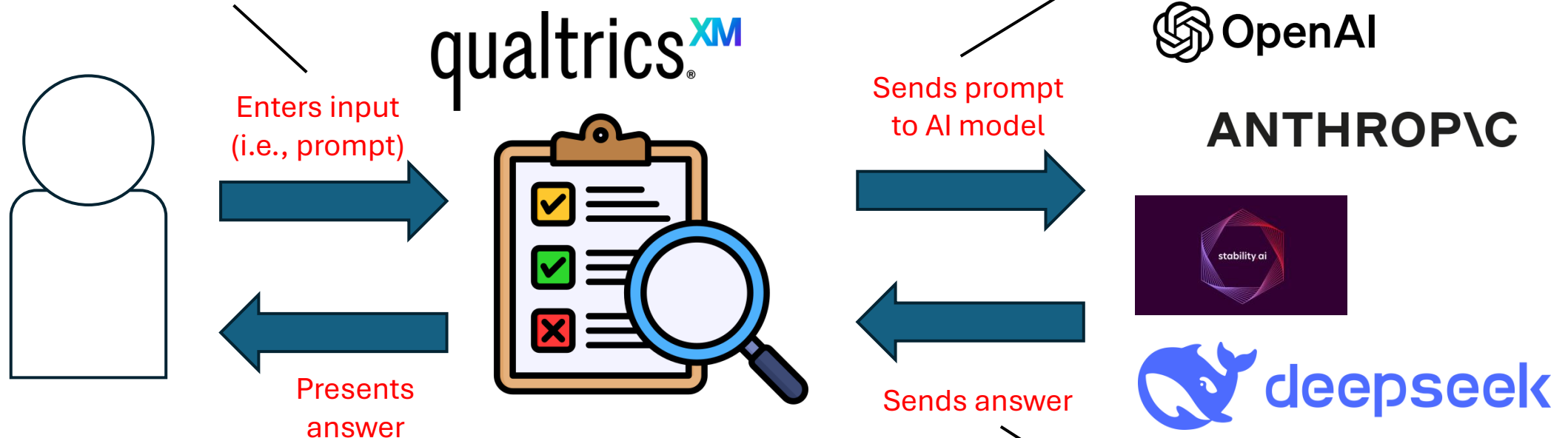
- How do the system adjustments (e.g., temperature, system prompt etc.) affect model output and user interaction?



# Examples of what to investigate and manipulate?

- How does the modality of data input affect people's interaction with generative AI?
- How does the input length affect the people's satisfaction or behavior with AI?

- How do the system adjustments (e.g., temperature, system prompt etc.) affect model output and user interaction?



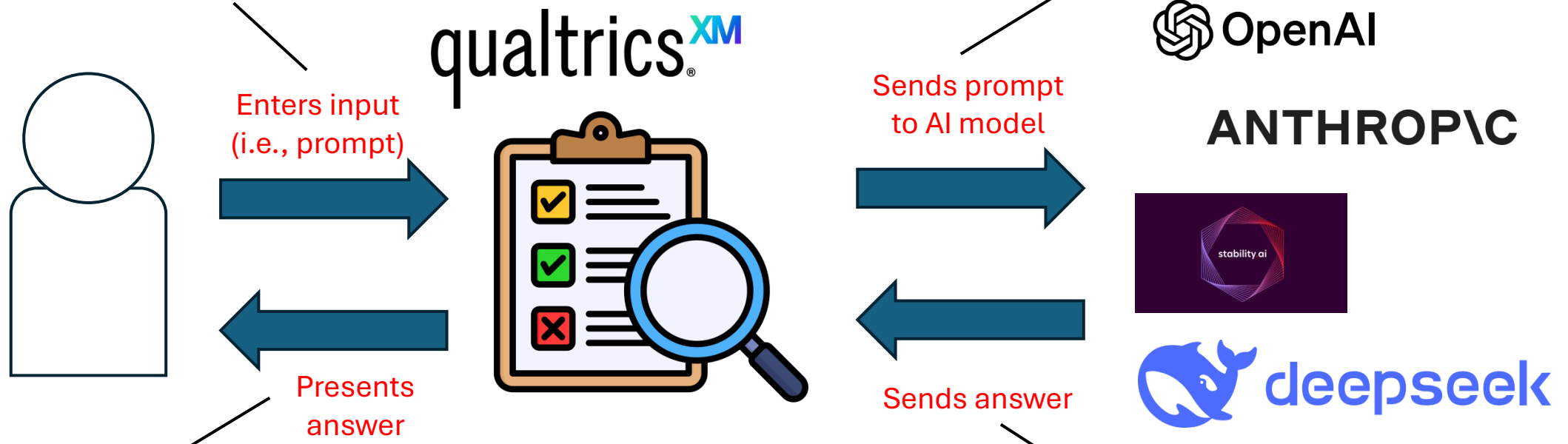
- What model sends the best answer for which task?



# Examples of what to investigate and manipulate?

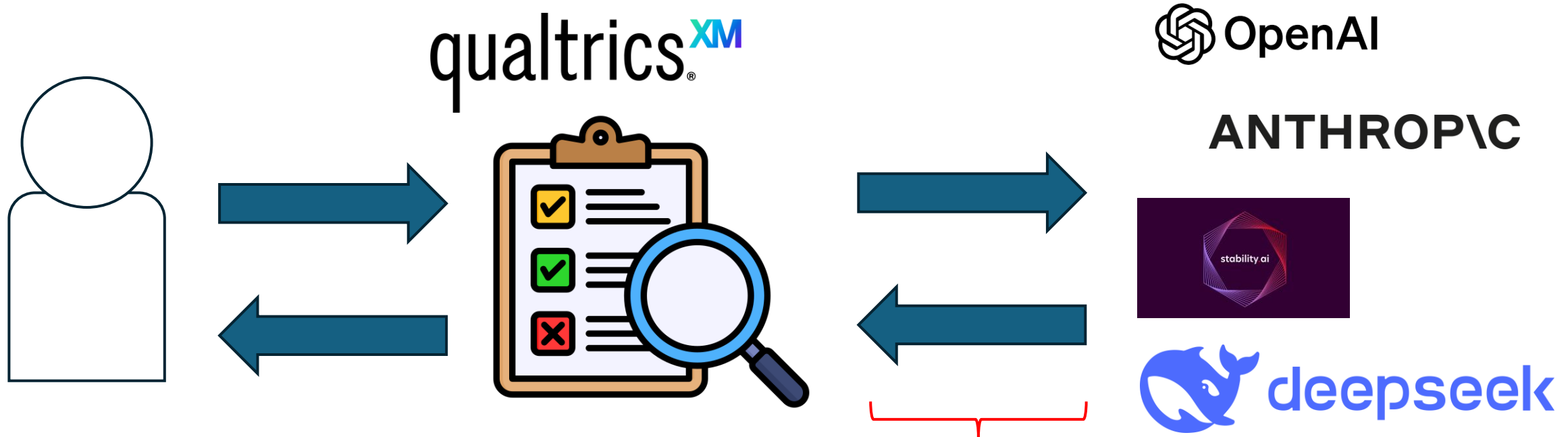
- How does the modality of data input affect people's interaction with generative AI?
- How does the input length affect the people's satisfaction or behavior with AI?

- How do the system adjustments (e.g., temperature, system prompt etc.) affect model output and user interaction?



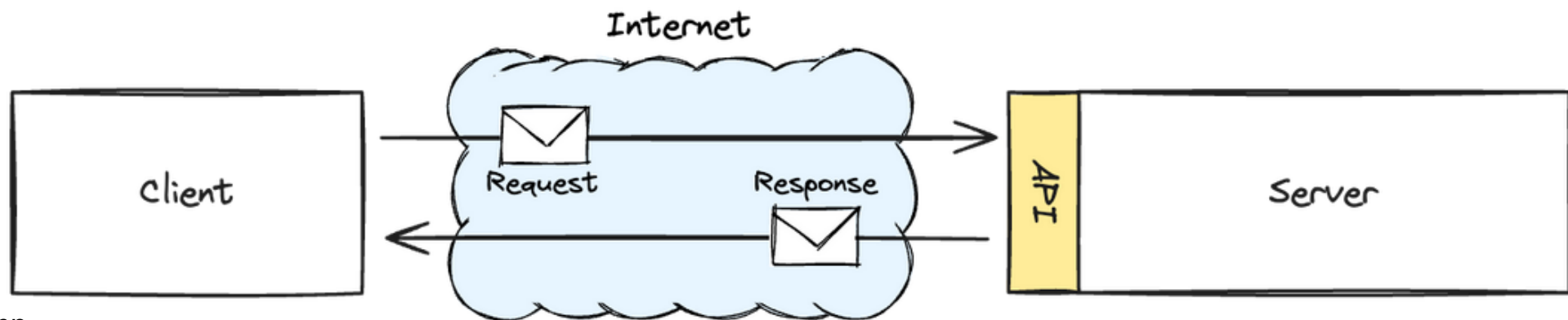
- How does the modality of data output affect people's interaction with generative AI?
- Is the reasoning process visible and how does it affect user experience?

- What model sends the best answer for which task?



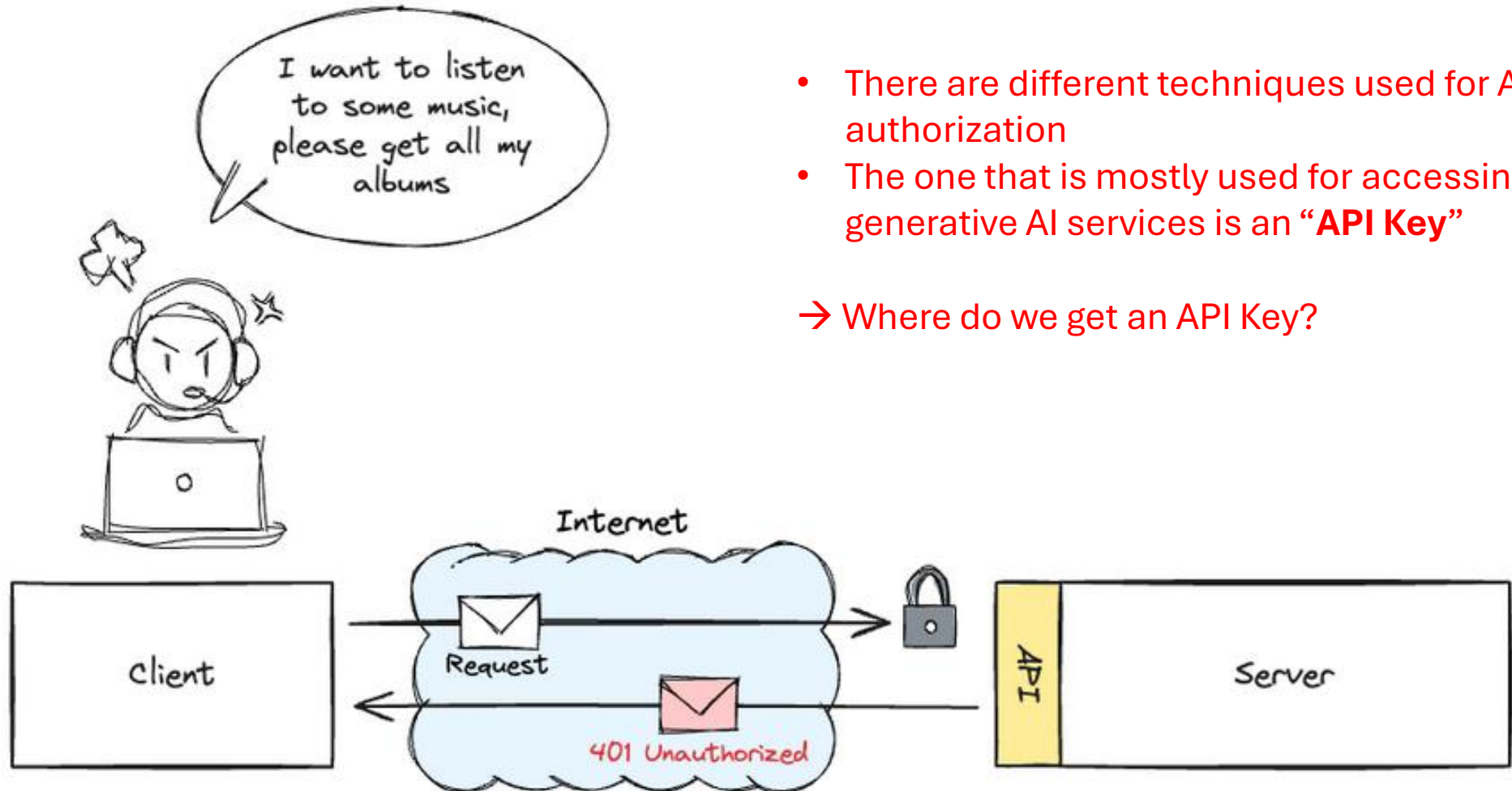
### How does this interaction work?

- Web services communicate via APIs (Application Programming Interface)



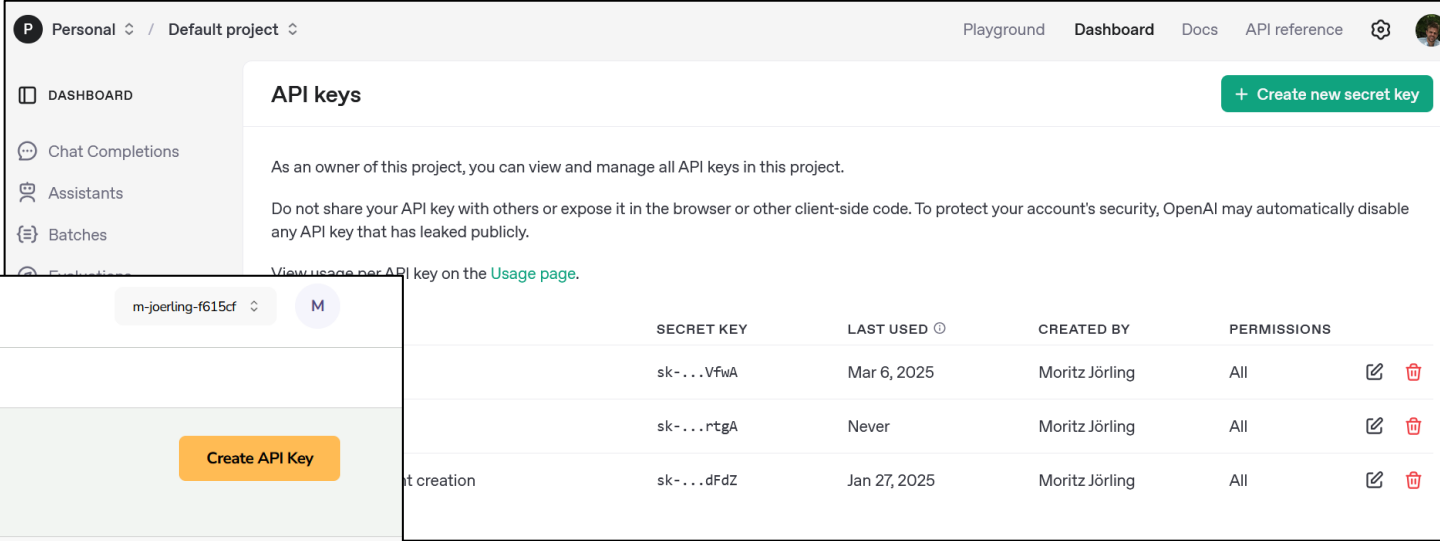
Source: Datacamp

# We need to have authorization to use the API



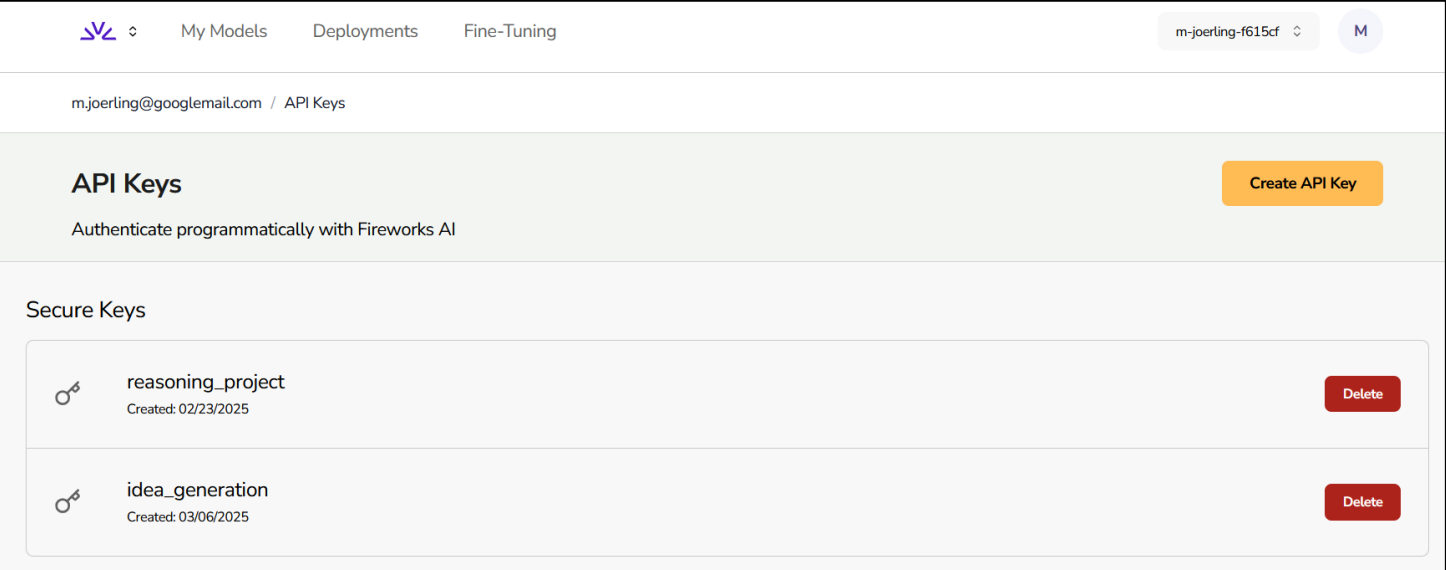
# Getting an API Key

1. Go to the service you want to use (e.g., Open AI, Anthropic, Mistral AI etc.)
  2. Setup an account
  3. Create an API Key and save it where no one else can access it
- Do not share your API Key with anyone!



The screenshot shows the OpenAI API Key management interface. The top navigation bar includes 'Personal', 'Default project', 'Playground', 'Dashboard', 'Docs', 'API reference', and a user profile icon. The left sidebar lists 'DASHBOARD', 'Chat Completions', 'Assistants', 'Batches', and 'Evaluations'. The main content area is titled 'API keys' and includes a '+ Create new secret key' button. Below this, there is a warning: 'As an owner of this project, you can view and manage all API keys in this project. Do not share your API key with others or expose it in the browser or other client-side code. To protect your account's security, OpenAI may automatically disable any API key that has leaked publicly. View usage per API key on the Usage page.'

SECRET KEY	LAST USED	CREATED BY	PERMISSIONS
sk-...VfwA	Mar 6, 2025	Moritz Jörling	All
sk-...rtgA	Never	Moritz Jörling	All
sk-...dFdZ	Jan 27, 2025	Moritz Jörling	All



The screenshot shows the Fireworks AI API Keys interface. The top navigation bar includes 'My Models', 'Deployments', and 'Fine-Tuning'. The left sidebar lists 'API Keys'. The main content area is titled 'API Keys' and includes a 'Create API Key' button. Below this, there is a section titled 'Secure Keys' with two entries: 'reasoning\_project' (Created: 02/23/2025) and 'idea\_generation' (Created: 03/06/2025). Each entry has a 'Delete' button.

Secure Keys	Created	Action
reasoning_project	Created: 02/23/2025	Delete
idea_generation	Created: 03/06/2025	Delete

# Anatomy of a REST API request

## 1. Endpoint (URL)

- The address where the API is hosted, defining the resource being accessed.
- Example: `https://api.example.com/users/123`

## 2. HTTP Method

- Specifies the type of action being performed.
- Common methods:
  - `GET` (Retrieve data)
  - `POST` (Create new data)
  - `PUT/PATCH` (Update data)
  - `DELETE` (Remove data)

## 3. Headers

- Provide metadata about the request.
- Examples:
  - `Content-Type: application/json` (Indicates request format)
  - `Authorization: Bearer <token>` (For authentication)

## 4. Query Parameters (Optional)

- Modify the request with key-value pairs.
- Example: `?page=2&limit=10`

## 5. Request Body (Optional)

- Contains data sent with `POST`, `PUT`, or `PATCH` requests.
- Typically formatted as JSON:

```
{  
  "name": "John Doe",  
  "email": "johndoe@example.com"  
}
```

## 6. Response

- The data returned by the API.
- Includes:
  - Status Code ( `200 OK`, `404 Not Found`, etc.)
  - Response Body (JSON, XML, etc.)
  - Headers (Metadata like rate limits)

## Status code categories

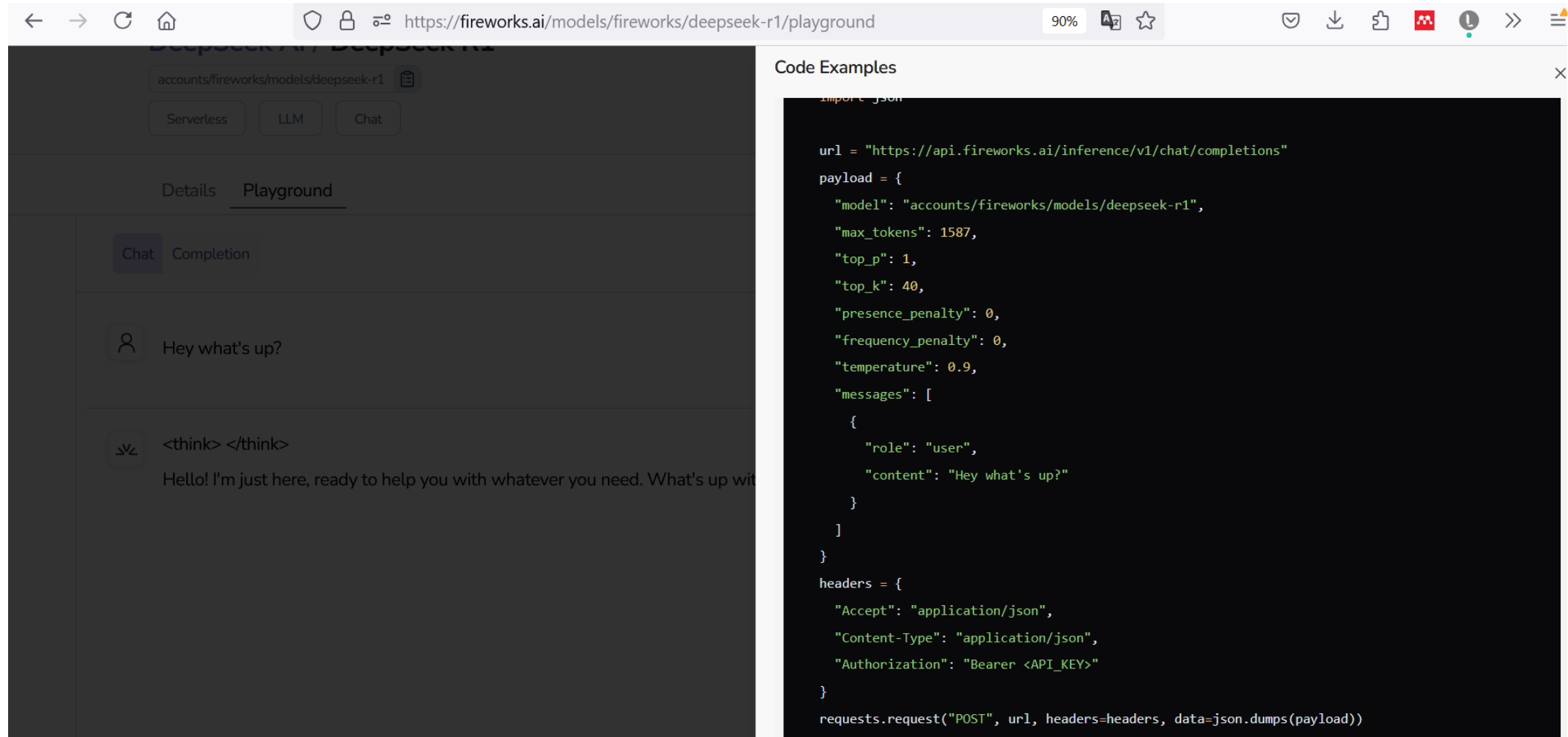
- 1XX : Informational responses
- 2XX : Successful responses
- 3XX : Redirection messages
- 4XX : Client error responses
- 5XX : Server error responses

## Frequently used status codes

- 200 : OK
- 404 : Not Found
- 500 : Internal Server Error

<sup>1</sup> For a full list of all response codes you can refer to the MDN page on status-codes via <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>

# Let's see it in action



The screenshot displays the Fireworks AI playground interface in a web browser. The browser's address bar shows the URL `https://fireworks.ai/models/fireworks/deepseek-r1/playground`. The page features a dark-themed chat interface with a sidebar on the left containing tabs for 'Serverless', 'LLM', and 'Chat'. The main chat area has tabs for 'Details' and 'Playground', with 'Playground' being the active tab. Within the 'Playground' tab, there are sub-tabs for 'Chat' and 'Completion', with 'Chat' selected. The chat history shows a user input 'Hey what's up?' and a model response 'Hello! I'm just here, ready to help you with whatever you need. What's up with'. Below the chat history, there is a text input field containing the prompt '<think> </think>'. To the right of the chat interface, a 'Code Examples' panel is open, displaying a Python code snippet for making an API call to the Fireworks AI inference endpoint. The code defines a URL, a JSON payload with various parameters like 'model', 'max\_tokens', 'top\_p', 'top\_k', 'presence\_penalty', 'frequency\_penalty', 'temperature', and 'messages', and then uses the 'requests' library to send a POST request.

```
import json

url = "https://api.fireworks.ai/inference/v1/chat/completions"

payload = {
    "model": "accounts/fireworks/models/deepseek-r1",
    "max_tokens": 1587,
    "top_p": 1,
    "top_k": 40,
    "presence_penalty": 0,
    "frequency_penalty": 0,
    "temperature": 0.9,
    "messages": [
        {
            "role": "user",
            "content": "Hey what's up?"
        }
    ]
}

headers = {
    "Accept": "application/json",
    "Content-Type": "application/json",
    "Authorization": "Bearer <API_KEY>"
}

requests.request("POST", url, headers=headers, data=json.dumps(payload))
```

<https://fireworks.ai/models>



# One thing before we try ourselves...

- **HTML** – What a website says (titles, body content, etc.)
- **CSS** – How a website looks (color, fonts, etc.)
- **JavaScript** – How it behaves (interactive, dynamic, etc.)

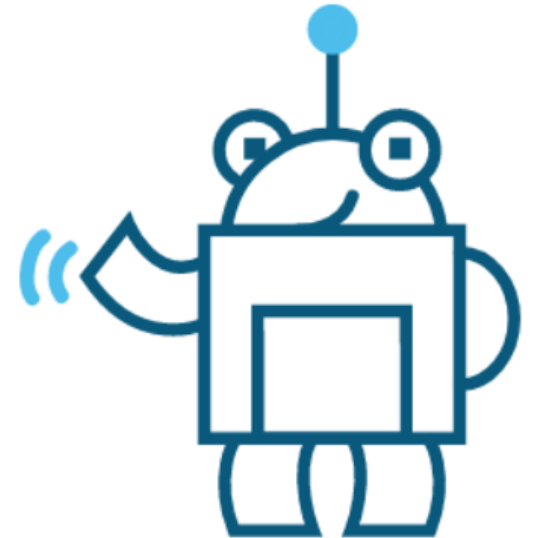
**HTML**  
Structure



**CSS**  
Appearance



**JavaScript**  
Action



# One thing before we try ourselves...

- **HTML** – What a website says (titles, body content, etc.)
- **CSS** – How a website looks (color, fonts, etc.)
- **JavaScript** – How it behaves (interactive, dynamic, etc.)

This is what we need for the interaction between Qualtrics and an API

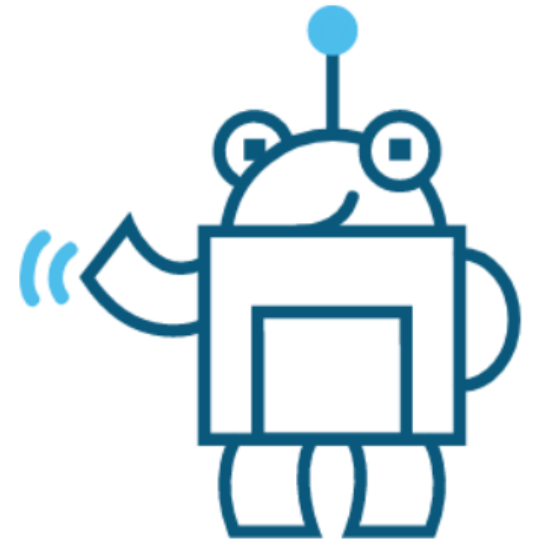
**HTML**  
Structure



**CSS**  
Appearance

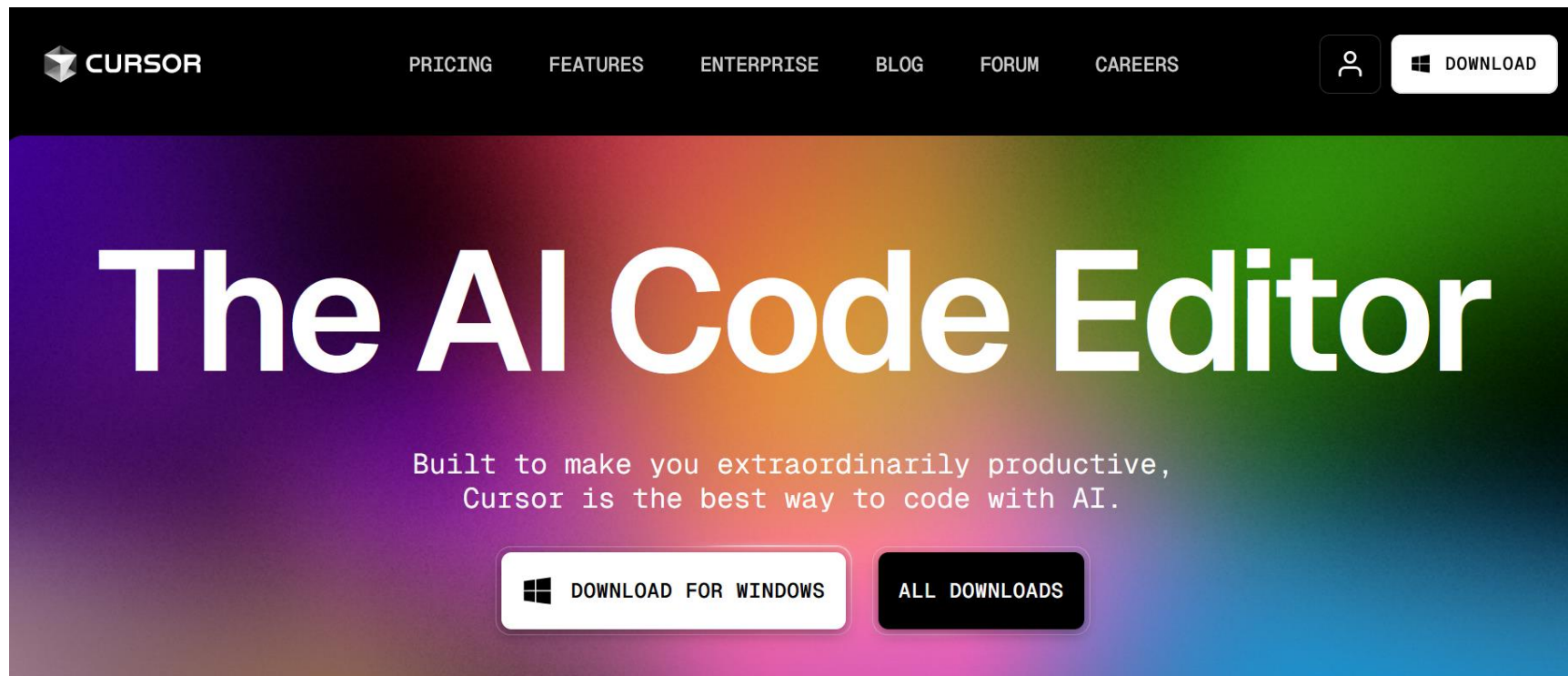


**JavaScript**  
Action

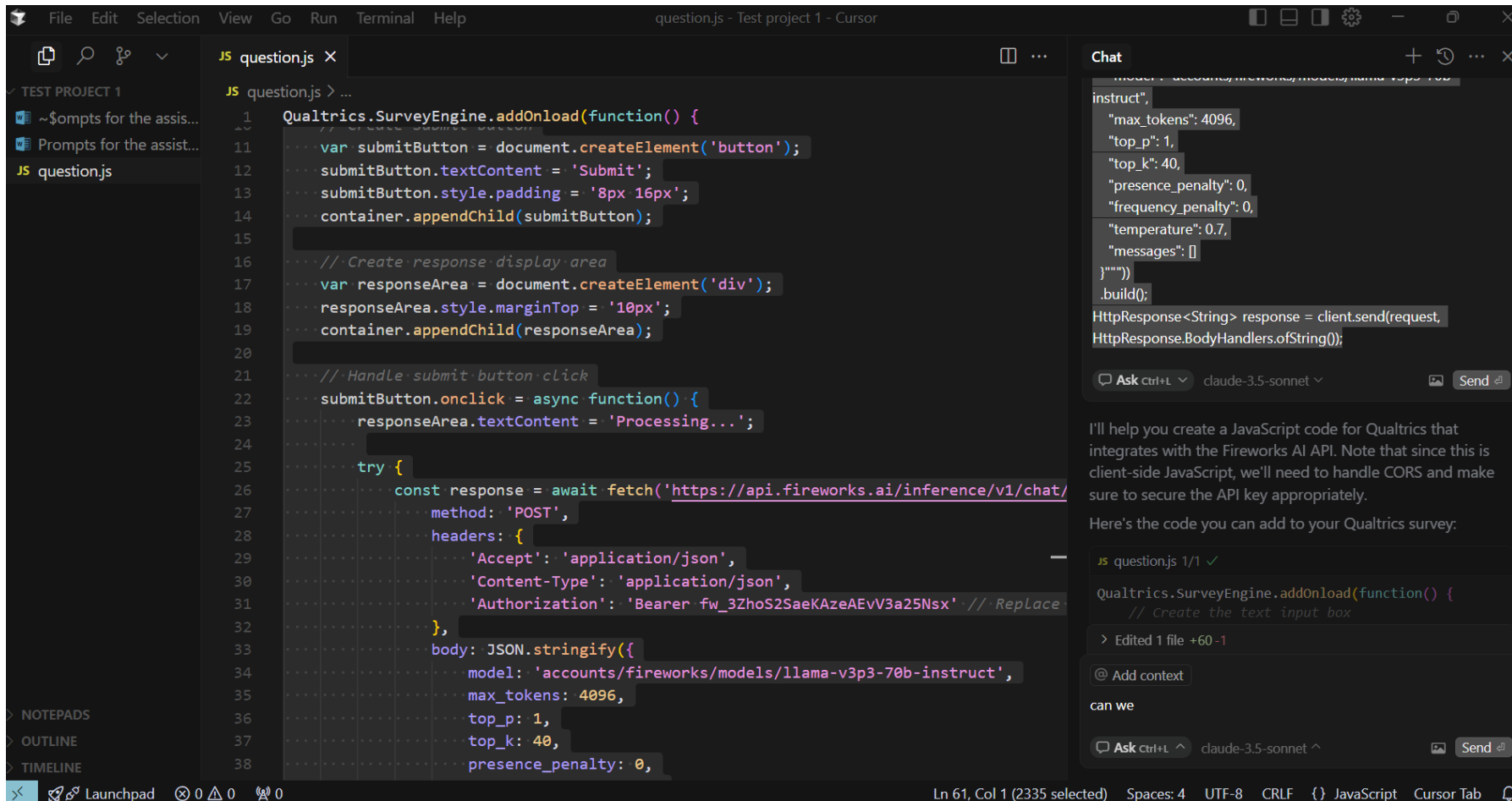


# Exercise

1. Go to Fireworks AI and get an API Key (Copy and save it)
2. Install Cursor AI



# What is cursor AI?



IDE = integrated development environment, a software application that provides comprehensive facilities for software development (source: Wikipedia)



Prompt: “Please create me a JavaScript code for qualtrics that provides a textbox for the user and sends this text input to the fireworks AI API. Here is the code snippet from fireworks how the api call should look like: [COPY AND PASTE THE API EXAMPLE FROM FIREWORKS AI]”

# Where to add the output from Cursor AI?

The screenshot displays a survey editor interface. On the left is a sidebar with the following sections:

- Question type**: A dropdown menu currently showing 'Text / Graphic'.
- Content type**: A dropdown menu currently showing 'Text'.
- Response requirements**: A toggle switch for 'Add validation' which is currently turned off.
- Question behavior**: Three options with icons: 'Display logic', 'Skip logic', and 'JavaScript'.

A red arrow points from the 'JavaScript' option in the 'Question behavior' section to the main survey content area. Overlaid on this area is the text: 'Mark the question and click on JavaScript'.

The main survey content area is titled 'Survey\_WS\_OSLO' and contains two blocks:

- Default Question Block**: Contains a question labeled 'Q1' with a checkbox icon. Below the label is the text 'Click to write the question text'.
- End of Survey**: Contains the text 'We thank you for your time spent taking this survey.' and 'Your response has been recorded.'

An 'Add Block' button is visible at the bottom right of the 'Default Question Block' section.

# Where to add the output from Cursor AI?

The image shows the Qualtrics survey editor interface. On the left, the 'Question type' dropdown is set to 'Text / Graphic'. Below it, the 'Content type' is 'Text'. Under 'Response requirements', there is a toggle for 'Add validation'. In the 'Question behavior' section, 'Display logic', 'Skip logic', and 'JavaScript' are listed. A red arrow points from the 'JavaScript' option to the 'Edit Question JavaScript' panel on the right. The main survey area shows a 'Default Question Block' with a question labeled 'Q1' and the text 'Click to write the question'. The 'Edit Question JavaScript' panel displays a code editor with a JavaScript snippet for fetching data from the OpenAI API. A red text overlay 'Add your API key' points to the 'Authorization' header in the code. The code includes a 'try' block for the fetch request and a 'catch' block for error handling. The 'Clear' button is at the bottom left of the code editor, and 'Discard changes' and 'Save' buttons are at the bottom right.

Question type

Text / Graphic

Content type

Text

Response requirements

Add validation

Question behavior

Display logic

Skip logic

JavaScript

Survey\_WS\_OSLO

Default Question Block

Q1

Click to write the question

End of Survey

Edit Question JavaScript

```
25 try {
26   const response = await fetch('https://api.fireworks.ai/inference/v1/chat/completions', {
27     method: 'POST',
28     headers: {
29       'Accept': 'application/json',
30       'Content-Type': 'application/json',
31       'Authorization': 'Bearer fw_3ZhoS2SaeKAZeAEVv3a25Nsx' // Replace with your actual API key
32     },
33     body: JSON.stringify({
34       model: 'accounts/fireworks/models/llama-v3p3-70b-instruct',
35       max_tokens: 4096,
36       top_p: 1,
37       top_k: 40,
38       presence_penalty: 0,
39       frequency_penalty: 0,
40       temperature: 0.7,
41       messages: [
42         {
43           role: 'user',
44           content: textBox.value
45         }
46       ]
47     })
48   });
49   const data = await response.json();
50   responseArea.textContent = data.choices[0].message.content;
51   // Store the response in the Qualtrics question
52   this.setChoiceValue(1, textBox.value + ' | Response: ' + data.choices[0].message.content);
53 } catch (error) {
54   responseArea.textContent = 'Error: ' + error.message;
55 }
56 }.bind(this);
57 }
58 }
59 }
60 }
61 }
```

Add your API key

Clear

Discard changes Save

# Where to add the output from Cursor AI?

The screenshot displays the SurveyJS editor interface. At the top, there's a tab labeled "Question type" and another tab labeled "Survey WS OSLO". Below these, a "Default Question Block" is visible. Inside this block, there's a checkbox labeled "Q1" which is checked. Below the checkbox, it says "Click to write the question text". To the right of this text area, there's a small icon of a code editor with the text "</>" inside it, which is circled in red. Below the question text area, there are two buttons: "Import from library" and "Add new question". At the bottom of the editor, there's a "Clear" button and a "Save" button. The text "Read more about the JS Question API" is also visible at the bottom right.



# Where to add the output from Cursor AI?

The screenshot displays the JS Question API editor interface. On the left, a sidebar titled "Question type" shows a "Default Question Block" with a checked item "Q1". The main editor area contains a question block with the text "Hello - who am I talking to?" and a "Submit" button. Below the submit button, the AI-generated response is visible: "Hello. I'm an artificial intelligence model known as Llama. Llama stands for 'Large Language Model Meta AI.'". A red circle highlights the code icon in the top right corner of the question block, indicating where to click to expand the model. The bottom right corner of the editor features a blue button with a right arrow. The right sidebar includes an "Expand model" link, an "Add new question" button, a link to "Read more about the JS Question API", and "Discard changes" and "Save" buttons.

Question type

Default Question Block

☒ Q1

Click to write the question text

Submit

Hello - who am I talking to?

Hello. I'm an artificial intelligence model known as Llama. Llama stands for "Large Language Model Meta AI."

Expand model

Add new question

Read more about the JS Question API

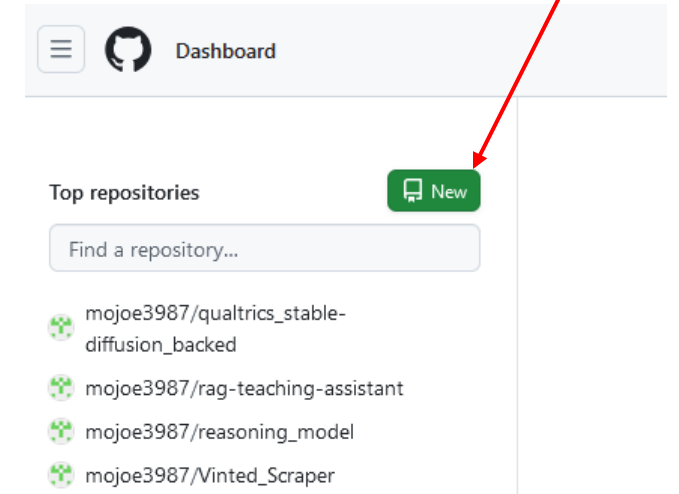
Discard changes Save

# Connect to GitHub



- Developer platform that allows developers to create, store, manage, and share their code
- Commonly used to host open source software development projects.
- As of January 2023, GitHub reported having over 100 million developers and more than 420 million repositories, including at least 28 million public repositories. It is the world's largest source code host as of June 2023. Over five billion developer contributions were made to more than 500 million open source projects in 2024.

- Please go to GitHub and create an account
- Then create a new repository



# Connect to GitHub



You need to give it a name


## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Required fields are marked with an asterisk (\*).

Owner \*

 mojoe3987

Repository name \*

/ wo\_oslo

✓ wo\_oslo is available.

Great repository names are short and memorable. Need inspiration? How about [friendly-potato](#) ?

Description (optional)

☐

Public

Anyone on the internet can see this repository. You choose who can commit.

☒

Private

You choose who can see and commit to this repository.

New

You can keep it private

 mojoe3987/rag-teaching-assistant

 mojoe3987/reasoning\_model

 mojoe3987/Vinted\_Scraper

# Connect to GitHub

You need to give it a name

## Create a new repository


A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository](#)

are

Copy the repository URL, return to Cursor AI and tell it to connect to this repository and to upload your local code to this GitHub

### Quick setup — if you've done this kind of thing before

 Set up in Desktop

or

HTTPS

SSH

[https://github.com/mojoe3987/wo\\_oslo.git](https://github.com/mojoe3987/wo_oslo.git)



Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

### ...or create a new repository on the command line

```
echo "# wo_oslo" >> README.md
git init
git add README.md
```



Private

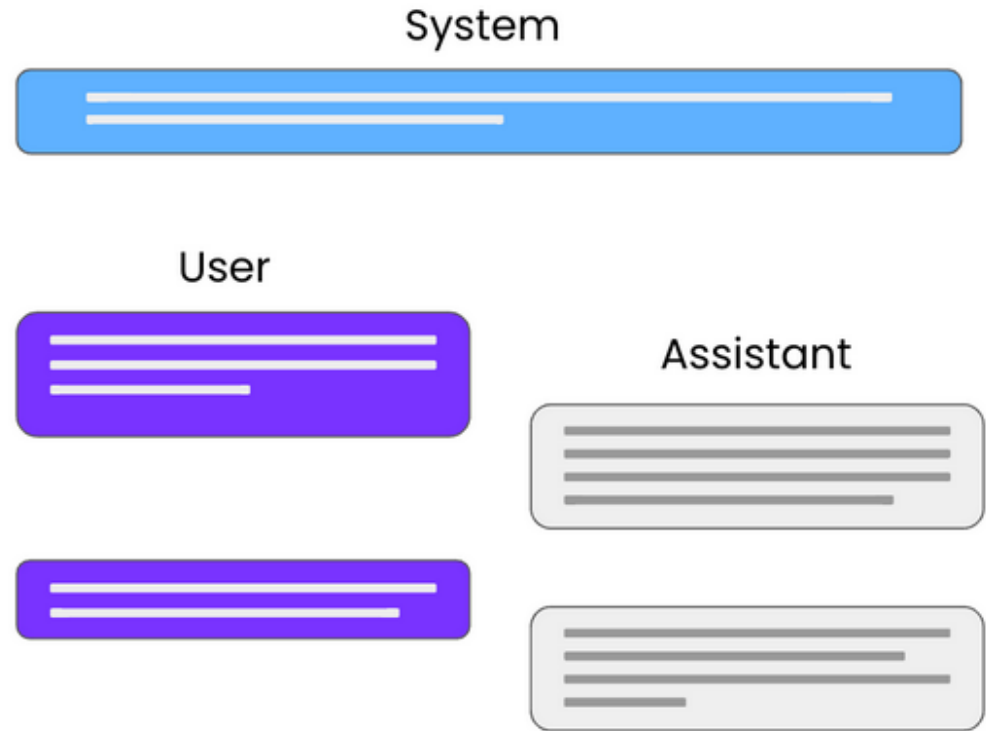
You choose who can see and commit to this repository.

You can keep it private

mojoe3987/rag-teaching-assistant  
mojoe3987/reasoning\_model  
mojoe3987/Vinted\_Scraper

# The different roles in an LLM conversation

- **System:** controls assistant's *behavior*
- **User:** *instruct* the assistant
- **Assistant:** response to user instruction
  - Can also be written by the user to provide examples



Source: Datacamp



Prompt: "Go to Cursor AI and tell it to add a system prompt that tells the LLM to answer in Shakespeare's language?"

# After the system prompt...

Hey there|

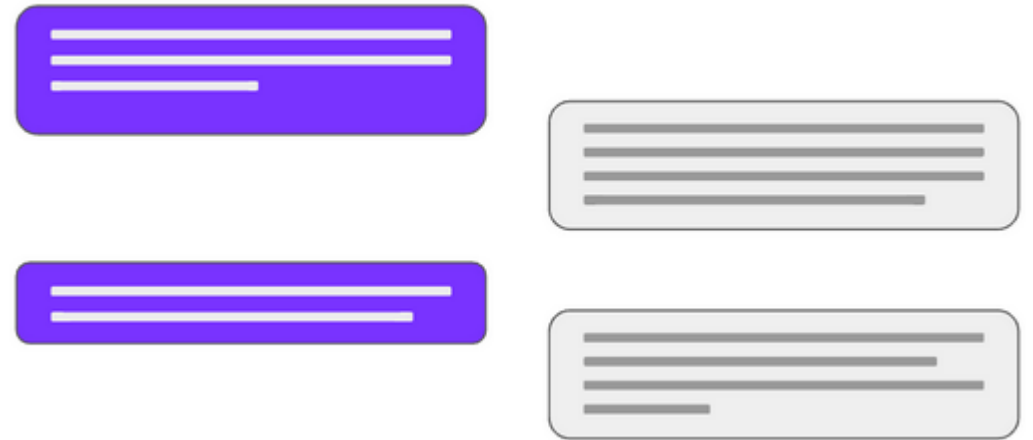
Submit

## **AI Response:**

Ho, thou dost address me with a greeting most brief and plain! Prithee, good fellow, how doth thy day fare? Doth the sun shine brightly upon thy countenance, and doth thy heart sing with joy and merriment? By thine informal salutation, I do perceive thou art a man of haste, with little time for elaborate courtesies. Yet, I shall respond in kind, with a warmth and courtesy befitting our chance encounter. Tell me, what dost thou here, and what dost thou seek in this, our fleeting conversation?

# Creating a conversation history

- Create conversation history
- Create back-and-forth conversations

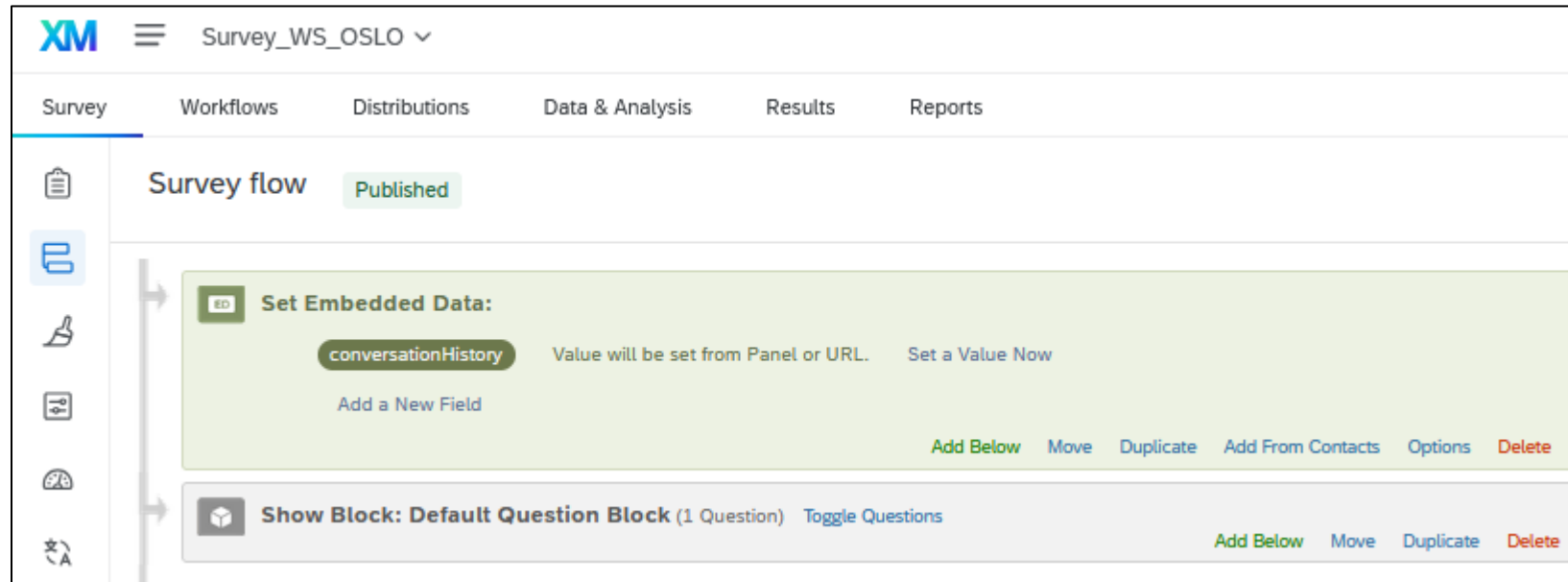






Prompt: “I have this working script, but I want that the conversation history is stored so that the LLM is aware of previous messages. Please save the conversation in an embedded data field “conversationHistory””

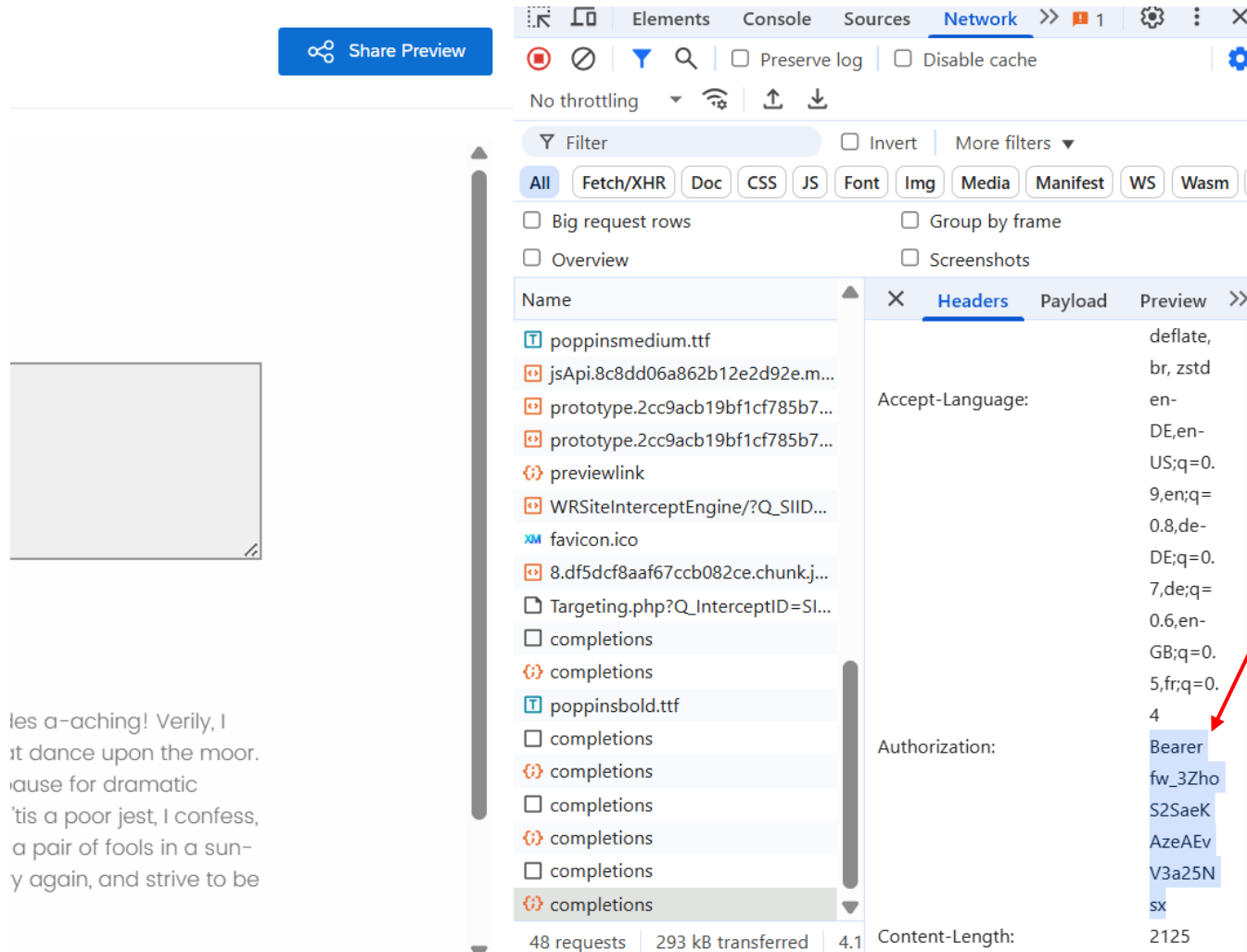
# Save the conversation in an embedded data field



# Save the conversation in an embedded data field

[illegible]

# The disadvantage of calling APIs directly from the Javascript



## Exposure of your API key:

- People can use your resources (especially dangerous when you have auto recharge)
- People can block API usage by overusing it

# You need to add backend to the app

User interface  
(also 'front-end')

Backend

Generative AI

qualtrics<sup>XM</sup>

 OpenAI

ANTHROPIC



 deepseek

You need to install Python on your system

Python

PSF

Docs

PyPI

Jobs

Community



Donate



Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

## Download the latest version for Windows

Download Python 3.13.2

Looking for Python with a different OS? Python for [Windows](#),  
[Linux/UNIX](#), [macOS](#), [Other](#)

Want to help test development versions of Python 3.14? [Pre-releases](#),  
[Docker images](#)

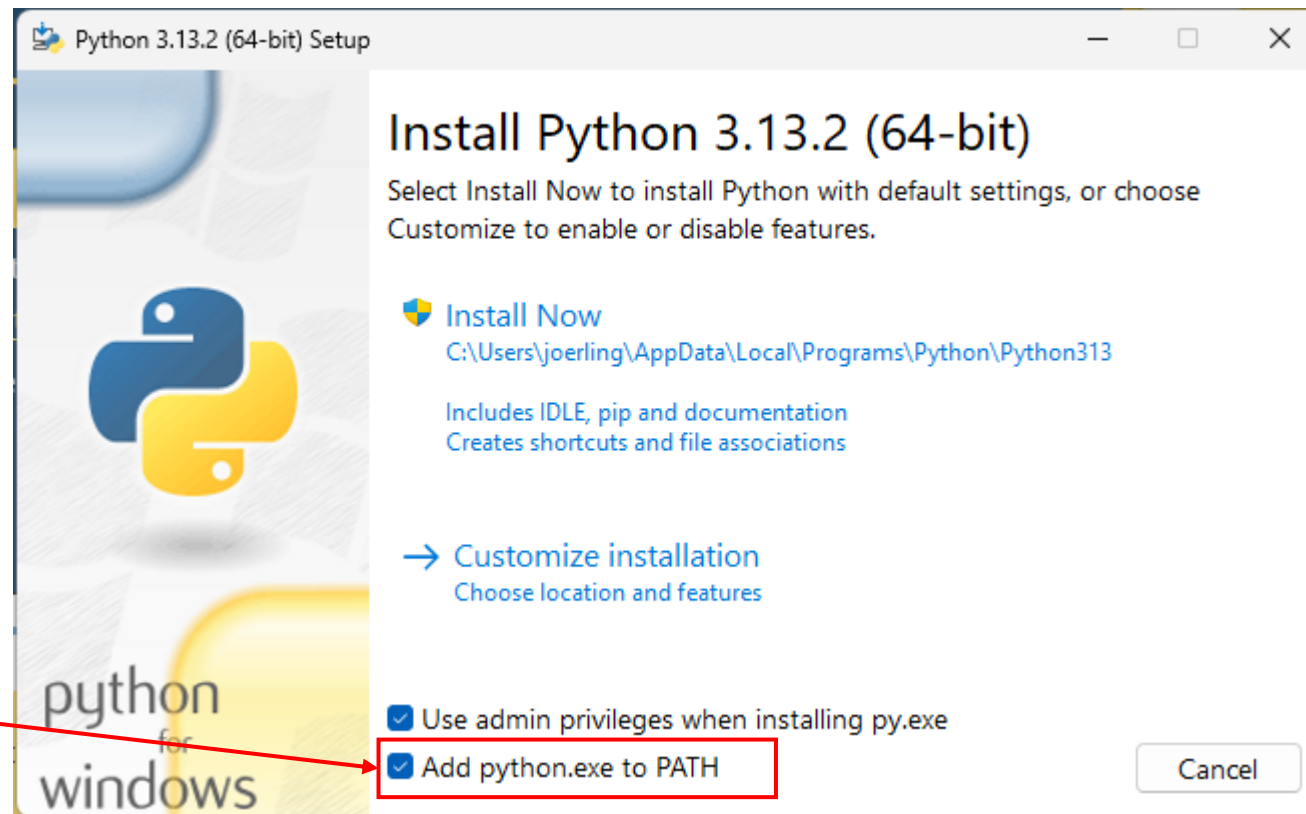


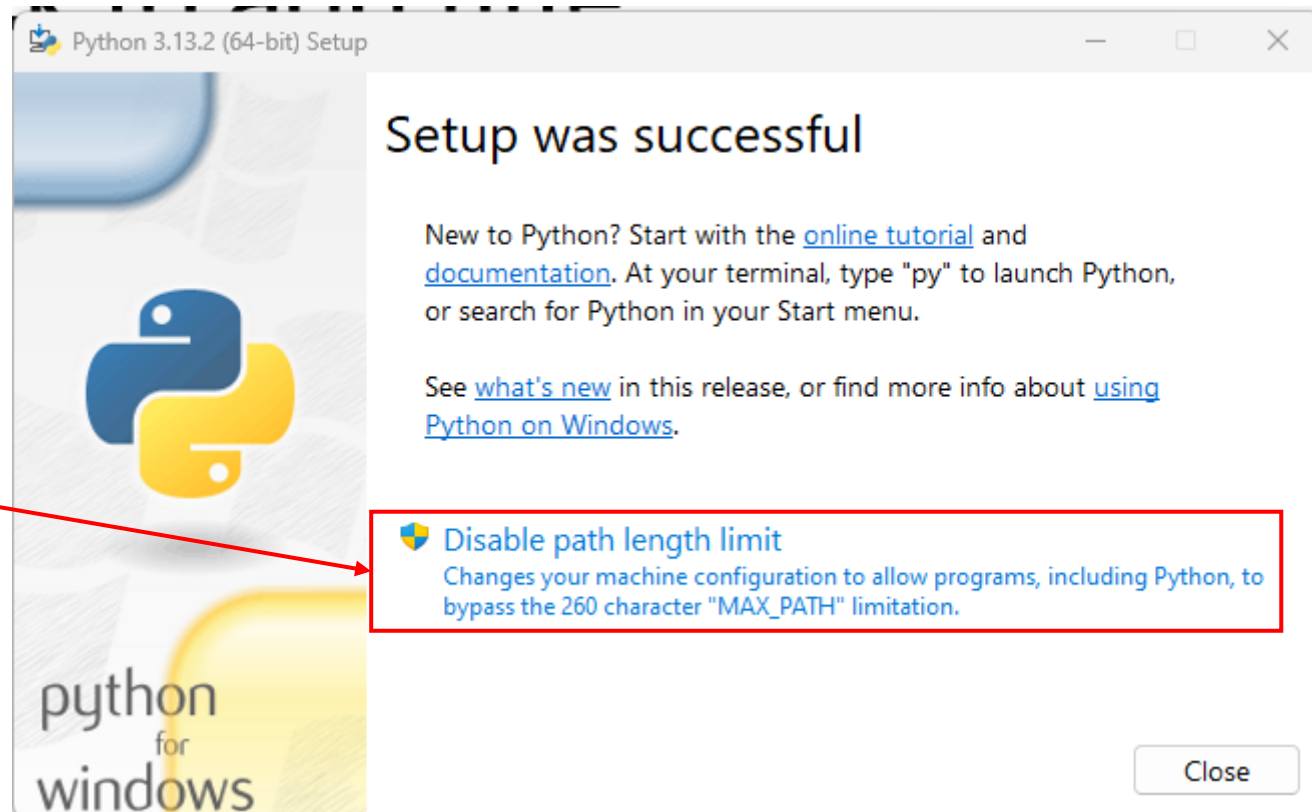
python-3.13.2-amd64.exe

27.3 MB • Done




Don't show when downloads finish  
You can change this anytime in [settings](#)









## MARKETPLACE

 175ms

Extension: Python X



# Python

ms-python

160,000,000



## Python language support with extension access ...

✓ *Installed*

Disable Uninstall | ☒ Auto Update

## DETAILS

## FEATURES

## CHANGELOG

EXTENSION PACK

# Python extension for Visual Studio Code

## Marketplace

Identifier

```
ms-  
python.python
```

# Commands for powershell terminal



Prompt: "I want to install pip."

Install pip (terminal):

```
PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 1> python -m ensurepip  
-upgrade
```



Prompt: "I want to create and activate a virtual environment."

Creating and activating a virtual environment (terminal):

```
PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 1> python -m venv venv  
PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 1> .\venv\Scripts\Activate.ps1
```



Prompt: "I have this script that works perfectly. It calls the Fireworks AI API with a JavaScript directly from Qualtrics. Can you please transform this Javascript into a Python script that I will use as a backend. The script should operate between the Qualtrics JS that calls the python script, which in turn calls the API and forwards the answer back to the JS in Qualtrics."

Prompt: "I want to install the requirements."

Install requirements (terminal):

```
(venv) PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 1> pip install -r requirements.txt
```

Prompt: "I want to set the API key as environment variable."

Install requirements (terminal):

```
(venv) PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 1  
> $env:FIREWORKS_API_KEY = "████████████████████████████████████████"
```

Prompt: "I want to run the app locally."

Run the app (terminal):

```
(venv) PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 1> python app.py
```

# The app should be running locally



```
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production
deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 738-891-640
127.0.0.1 - - [28/Mar/2025 14:17:21] "POST /chat HTTP/1.1" 200 -
```

# You can test the app via the terminal



Prompt: "I want to test the app locally with curl method."

```
joerling@PRT27069 MINGW64 ~/Dropbox/Workshop Oslo/Test project 1 (main)
$ curl -X POST http://localhost:5000/chat \
-H "Content-Type: application/json" \
-d '{
  "conversation_history": [
    {"role": "user", "content": "Hello, how are you?"}
  ]
}'
{
  "response": "Hello! I'm doing well, thanks for asking. I'm a large language
model, so I don't have feelings or emotions like humans do, but I'm always h
appy to chat and help with any questions or topics you'd like to discuss. How
about you? How's your day going so far?"
}
```

To communicate with  
Qualtrics it needs to  
be hosted online

# There are different options where to host an app

## Easier options:

- Heroku
- Render

## More scalable options but more complex:

- Google cloud services
- Microsoft Azure
- Amazon Web Services
- Digital Ocean



### Create an account

GitHub

GitLab

Bitbucket

Google

or

Email

your@email.com

Password

correct horse battery staple

By signing up you agree to our [terms of service](#) and [privacy policy](#).

Create Account

# Choose Web Service

## **Static Sites**

Static content served over a global CDN. Ideal for frontend, blogs, and content sites.

[New Static Site →](#)

## **Web Services**

Dynamic web app. Ideal for full-stack apps, API servers, and mobile backends.

[New Web Service →](#)

## **Private Services**

Web app hosted on a private network, accessible only from your other Render services.

[New Private Service →](#)

## **Background Workers**

Long-lived services that process async tasks, usually from a job queue.

[New Worker →](#)

## **Cron Jobs**

Short-lived tasks that run on a periodic schedule.

[New Cron Job →](#)

## **Postgres**

Relational data storage. Supports point-in-time recovery, read replicas, and high availability.

[New Postgres →](#)

## **Key Value**

Managed Redis®-compatible storage. Ideal for use as a shared cache, message broker, or job queue.

[New Key Value Instance →](#)



# Connect your GitHub

Git Provider

Public Git Repository

Existing Image

## Connect Git provider

Connect your Git provider to deploy from your existing repositories.



GitHub



GitLab



Bitbucket

# Install Render in your GitHub and choose the repository



Install [Render](#)

Install on your personal account mojoe3987



for these repositories:

☒ **All repositories**

This applies to all current and future repositories owned by the resource owner.  
Also includes public repositories (read-only).

☐ **Only select repositories**

Select at least one repository. Also includes public repositories (read-only).

with these permissions:

## You are deploying a Web Service

You seem to be using **Flask**, so we've autofilled some fields accordingly. Make sure the values lo

**Source Code**

 mojoe3987 / wo\_oslo • 1h ago

**Name**

A unique name for your web service.

wo\_oslo

# Install Render in your GitHub and choose the repository



Prompt: "I want to deploy my app in render. what would I set as a start command?"

## Install [Render](#)

Install on your personal account mojoe3987



for these repositories:

☒ **All repositories**

This applies to all current and future repositories owned by the resource owner. Also includes public repositories (read-only).

☐ **Only select repositories**

Select at least one repository. Also includes public repositories (read-only).

with these permissions:

## You are deploying a Web Service

You seem to be using **Flask**, so we've autofilled some fields accordingly. Make sure the values lo

wo\_oslo • 1h ago

### Name

A unique name for your web service.

wo\_oslo|

### Start Command

Render runs this command to start your app with each deploy.

\$ gunicorn app:app

### Instance Type

#### For hobby projects

**Free**

**\$0 / month**

512 MB (RAM)

0.1 CPU

# Set your API key as environment variable


## Environment Variables

Set environment-specific config and secrets (such as API keys), then read those values from your code. [Learn more.](#)

FIREWORKS\_API\_KEY

.....

+ Add Environment Variable

 Add from .env

Deploy Web Service

All logs ▾

🔍 Search

⚡ Live tail ▾

GMT+1



uous-2.2.0 packaging-24.2 requests-2.32.3 urllib3-2.3.0

Mar 18 06:18:47 PM ⓘ

Mar 18 06:18:47 PM ⓘ [notice] A new release of pip is available: 24.0 -> 25.0.1

Mar 18 06:18:47 PM ⓘ [notice] To update, run: pip install --upgrade pip

Mar 18 06:18:48 PM ⓘ ==> **Uploading build...**

Mar 18 06:18:51 PM ⓘ ==> **Uploaded in 2.5s. Compression took 0.3s**

Mar 18 06:18:51 PM ⓘ ==> **Build successful 🎉**

Mar 18 06:18:53 PM ⓘ ==> **Deploying...**

Mar 18 06:19:17 PM ⓘ ==> **Running 'gunicorn app:app'**

Mar 18 06:19:21 PM ⓘ [2025-03-18 17:19:21 +0000] [92] [INFO] Starting gunicorn 23.0.0

Mar 18 06:19:21 PM ⓘ [2025-03-18 17:19:21 +0000] [92] [INFO] Listening at: <http://0.0.0.0:10000> (92)

Mar 18 06:19:21 PM ⓘ [2025-03-18 17:19:21 +0000] [92] [INFO] Using worker: sync

Mar 18 06:19:21 PM ⓘ [2025-03-18 17:19:21 +0000] [93] [INFO] Booting worker with pid: 93

Mar 18 06:19:22 PM ⓘ 127.0.0.1 - - [18/Mar/2025:17:19:22 +0000] "HEAD / HTTP/1.1" 404 0 "-" "Go-http-client/1.1"

Mar 18 06:19:24 PM ⓘ ==> **Your service is live 🎉**

Mar 18 06:19:25 PM ⓘ 127.0.0.1 - - [18/Mar/2025:17:19:25 +0000] "GET / HTTP/1.1" 404 207 "-" "Go-http-client/2.0"

WEB SERVICE

wo\_oslo

Python 3

Free

Upgrade your instance →

Connect ▾

mojoe3987 / wo\_oslo 🔗 main

<https://wo-oslo.onrender.com> 📄

With this URL you can  
access your app

ⓘ Your free instance will spin down with inactivity, which can delay requests by 50 seconds or more.

March 18, 2025 at 6:18 PM

🔄 Building

[760238d](#) Changed Qualtrics script to use hosted backend

All logs ▾

🔍 Search

⚡ Live tail ▾

GM

CURSOR

```
Prompt: "It works. Now I need to update my JavaScript that it can call the  
backend from Qualtrics. This is the URL where the app is hosted  
[https://wo-oslo.onrender.com]"
```

# Retrieval-Augmented Generation (RAG)

## Traditional LLM Limitations ⚠️

- 📅 Static knowledge cutoff dates
- 💭 Prone to hallucinations
- 🔒 Limited access to specialized knowledge
- 🌫️ Lack of source transparency

## User Question

For which club does  
Leonel Messi play?

## Prompt

```
[{"role":  
  "system", "content" :  
    "Answer the following question the  
best you can."},  
{"role":  
  "user", "content" :  
    "For which club does Leonel Messi  
play?"}]
```

## LLM Response

As of my latest update in 2021,  
Lionel Messi  
plays for  
Paris Saint-Germain Football Club.



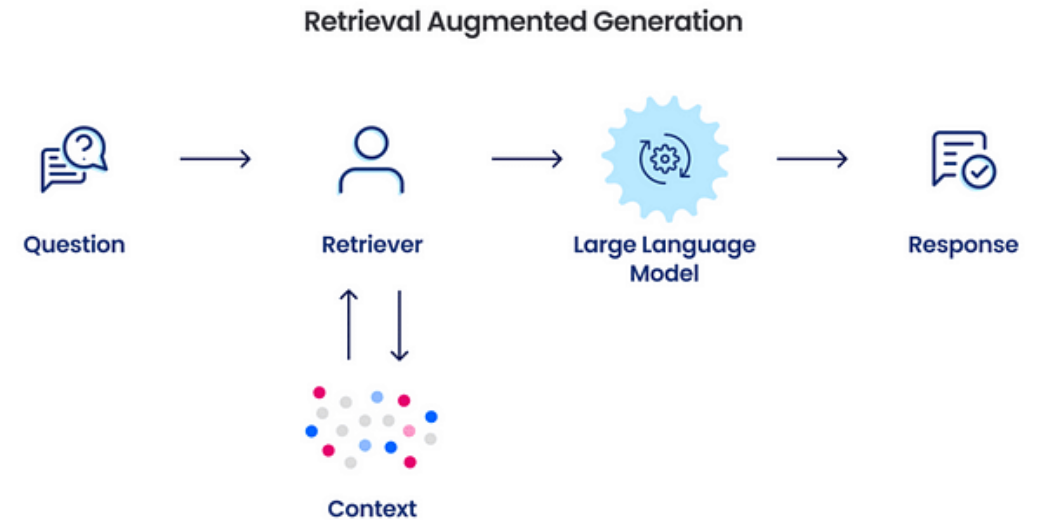
# RAG process (simplified)

## Knowledge Enhancement with RAG 📄

- 🔄 Extends knowledge beyond training cutoff
- 🌐 Accesses current information
- ⚓ Grounds responses in factual data
- 🔍 Taps into domain-specific knowledge
- 🧠 Combines model intelligence with external facts

## Source Integrity Benefits 🛡️

- 📄 Provides verifiable citations
- ⚖️ Reduces misinformation risk
- 🔗 Links claims to original sources



## User Question

For which club does  
Leonel Messi play?

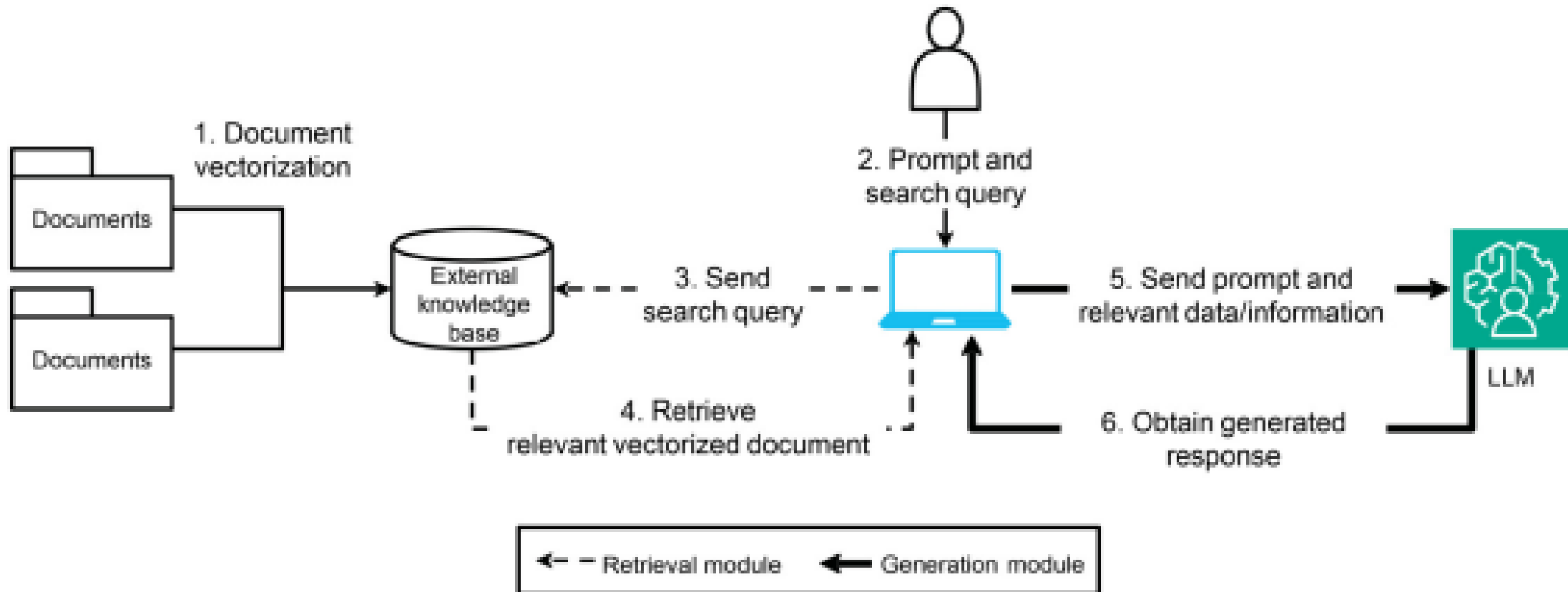
## Prompt

```
[{"role":  
  "system", "content" :  
    "Answer the following question the  
    best you can. Lionel Messi has signed  
    with Inter Miami CF, the club has  
    announced. The Argentine superstar has  
    joined David Beckham's club, and his  
    contract will run until 2025. The  
    World Cup winner said in a statement  
    from the club: I'm very excited to  
    start this next step in my career with  
    Inter Miami and in the United  
    States."},  
{"role":  
  "user", "content" :  
    "For which club does Leonel Messi  
    play?"}]
```

## LLM Response

Leonel Messi plays for Inter Miami  
CF.

# RAG architecture



Source: Arora, N., Chakraborty, I., & Nishimura, Y. (2025). AI-Human Hybrids for Marketing Research: Leveraging Large Language Models (LLMs) as Collaborators. *Journal of Marketing*, 89(2), 43-70. <https://doi.org/10.1177/00222429241276529> (Original work published 2025)

LOAD



SPLIT



EMBED



STORE

[ 0.3, 0.4, 0.1, 1.8, 1.1... ]

[ 0.7, 1.4, 2.1, 4.8, 4.1... ]

[ 1.2, 0.3, 1.2, 4.1, 1.8... ]



# Visualizing Chunking/Splitting

Splitter: Character Splitter  

Chunk Size: 90 

Chunk Overlap: 0 

Total Characters: 2658

Number of chunks: 30

Average chunk size: 88.6

[Upload .txt](#)

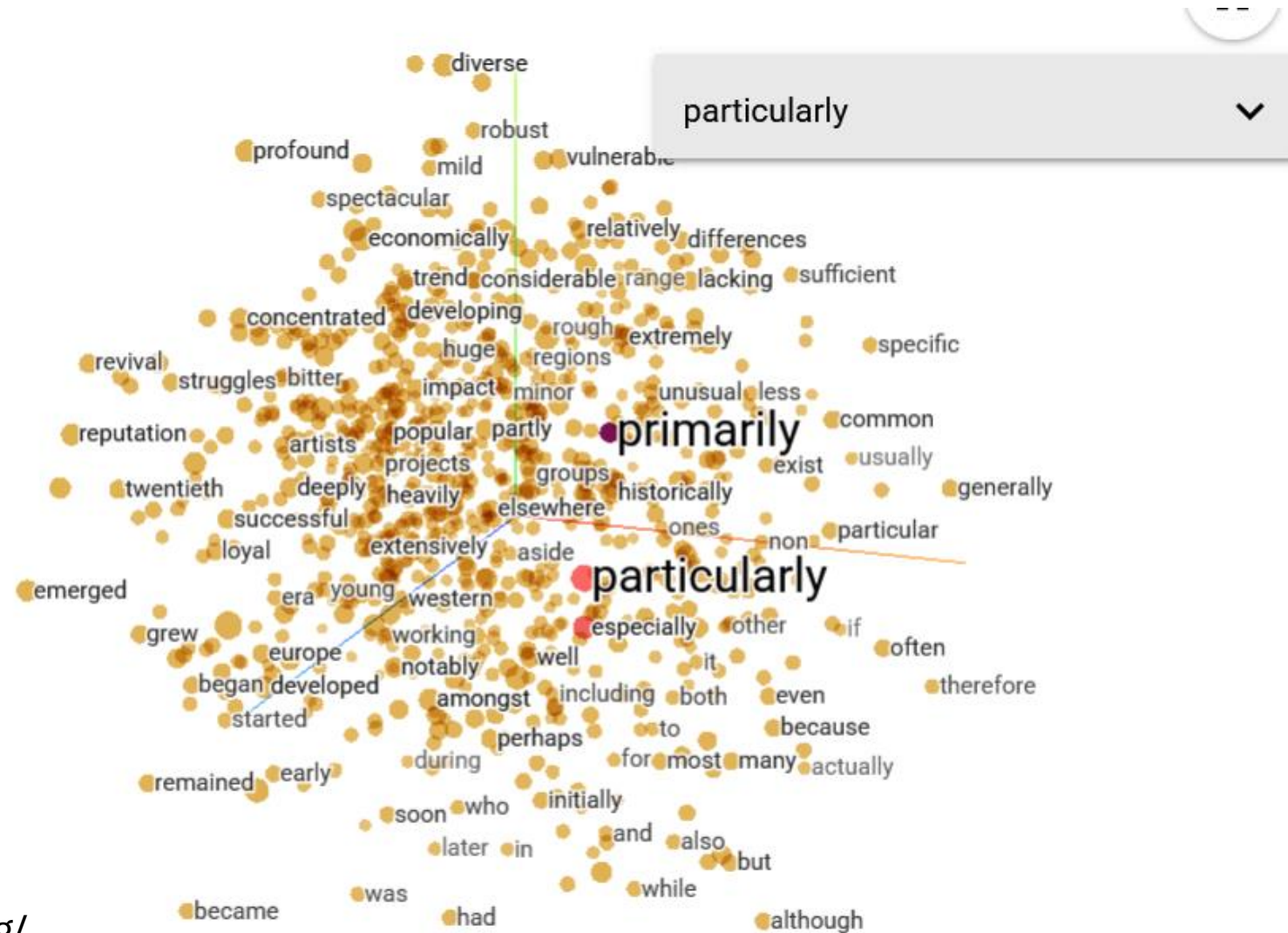
One of the most important things I didn't understand about the world when I was a child is the degree to which the returns for performance are superlinear.

Teachers and coaches implicitly told us the returns were linear. "You get out," I heard a thousand times, "what you put in." They meant well, but this is rarely true. If your product is only half as good as your competitor's, you don't get half as many customers. You get no customers, and you go out of business.

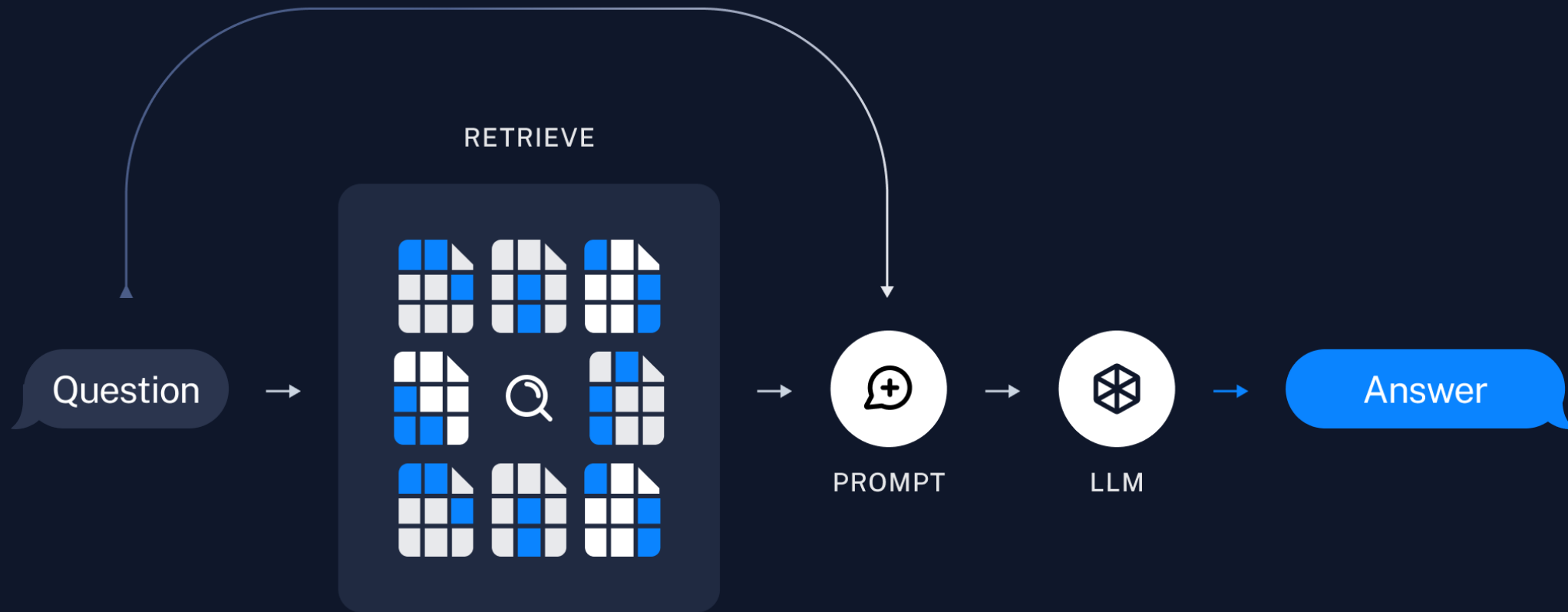
It's obviously true that the returns for performance are superlinear in business. Some think this is a flaw of capitalism, and that if we changed the rules it would stop being true. But superlinear returns for performance are a feature of the world, not an artifact of rules we've invented. We see the same pattern in fame, power, military victories, knowledge, and even benefit to humanity. In all of these, the rich get richer. [1]

<https://chunkviz.up.railway.app/>

# Visualizing Embeddings



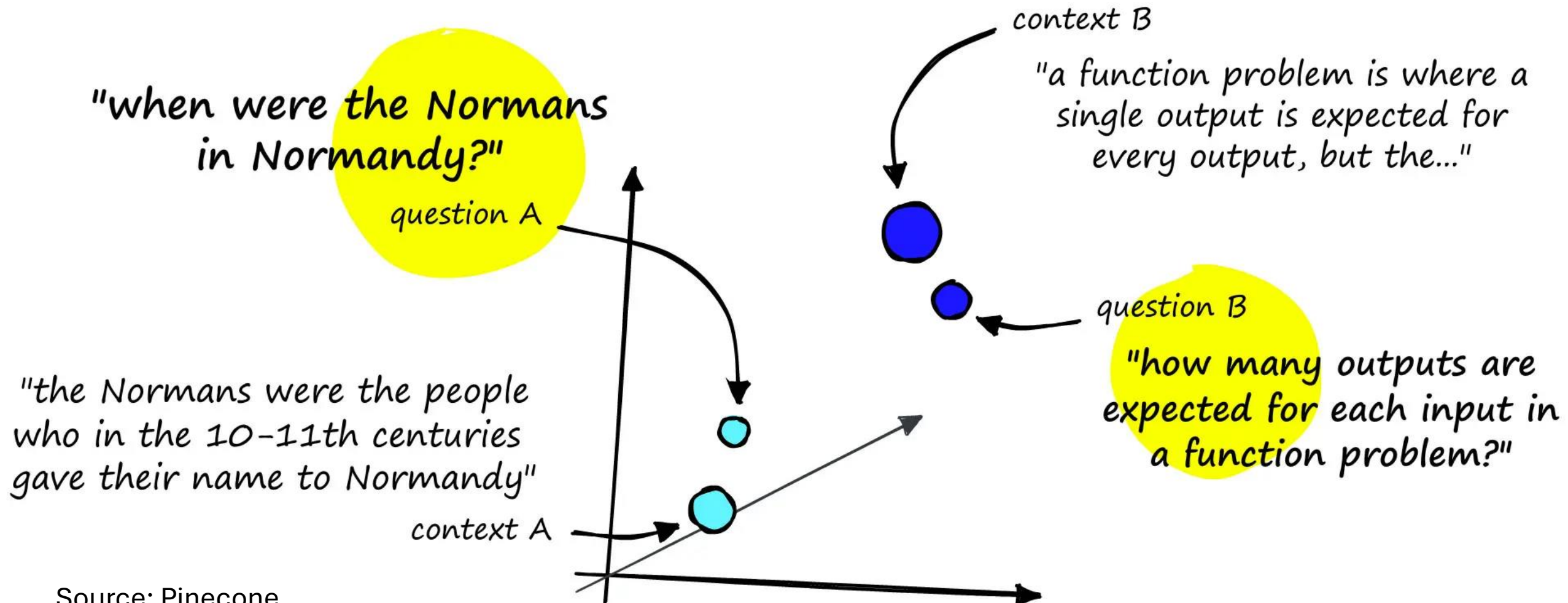
<https://projector.tensorflow.org/>



A retriever does also vectorize the question/prompt to find the relevant context in the vector database



# Retriever model





# Anything LLM (build your personal RAG)

 Anything LLM



Can you tell me something about a recent information leak scandal on signal?



There hasn't been a recent major "information leak scandal" involving Signal breaching its own encryption or servers to expose user messages. Signal's design with end-to-end encryption means they don't have access to message content.



However, there was an incident in **August 2022** that affected a small number of Signal users:







01. **Twilio Breach:** Signal uses a third-party service called Twilio to handle phone number verification texts (the codes you get when registering). Twilio suffered a phishing attack where attackers gained access to some of their internal systems.
02. **Impact on Signal Users:** For about 1,900 Signal users (a tiny fraction of the total user base), the attackers could potentially have accessed their phone numbers via Twilio's compromised systems. This could have allowed the attackers to attempt to re-register the user's phone number to a *different* device.
03. **What Was NOT Exposed:** Crucially, this incident **did not** expose any message history, contact lists, profile information, blocked lists, or other personal data. Message history is stored only on the user's device, not on Signal's servers.
04. **Signal's Response:** Signal notified the affected users directly and advised them to re-register Signal on their legitimate devices. They also encouraged all users to enable the "Registration Lock" feature, which requires your Signal PIN to register your phone number on a new device, mitigating this type of attack.

So, while an incident related to Signal account verification occurred via a third party, it wasn't a leak of Signal's encrypted messages or a breach of Signal's core infrastructure itself.

## My Documents

 New Folder

Name

- ☐  custom-documents
- ☐  Leavitt\_ Waltz 'took responsibility' for Signal ...ICO.pdf
- ☐  Signal downloads spike in the US and Yemen amid ...nch.pdf
- ☐  The Atlantic publishes full Signal chat messages...men.pdf
- ☐  Trump Officials Leak Yemen War Plans to the Atla...hat.pdf
- ☐  US national security scandal deepens as new Sign...ews.pdf



Click to upload or drag and drop

supports text files, csv's, spreadsheets, audio files, and more!

or submit a link

Fetch website

These files will be uploaded to the document processor running on this AnythingLLM instance.  
These files are not sent or shared with a third party.

## My Workspace

Name

No Documents



## My Documents

+ New Folder

Name

☐ 

Click to upload or drag and drop

supports text files, csv's, spreadsheets, audio files, and more!

or submit a link

Fetch website

These files will be uploaded to the document processor running on this AnythingLLM instance.  
These files are not sent or shared with a third party.

## My Workspace

Name

- 📄 Leavitt\_ Waltz 'took responsibility' for Si...ICO.pdf
- 📄 Signal downloads spike in the US and Yemen ...nch.pdf
- 📄 The Atlantic publishes full Signal chat mes...men.pdf
- 📄 Trump Officials Leak Yemen War Plans to the...hat.pdf
- 📄 US national security scandal deepens as new...ews.pdf



Save and Embed



Can you tell me something about a recent information leak scandal on signal?



Based on the provided context, there has been a recent scandal involving senior officials in the Trump administration who used the encrypted messaging app Signal for a group chat.

Key points about the scandal include:

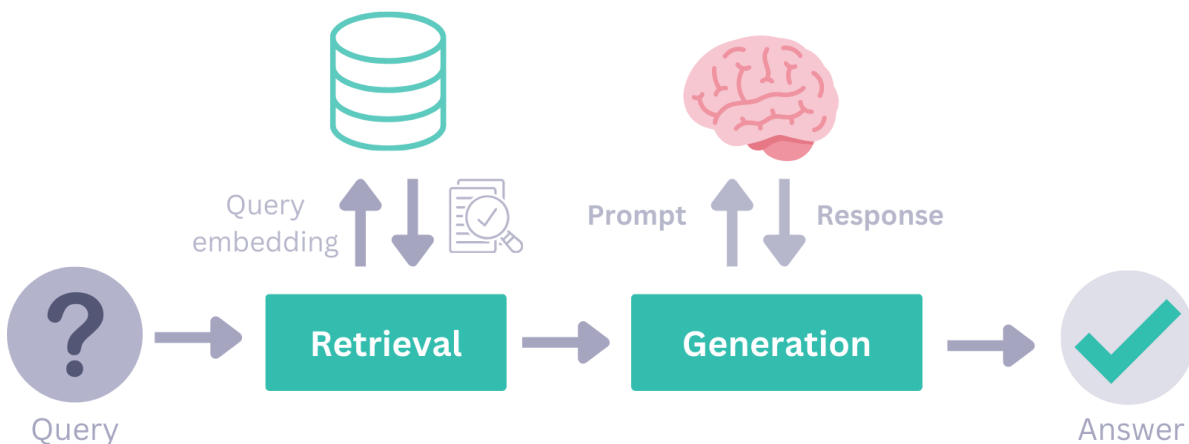


01. **Participants:** Senior national security officials, including Defense Secretary Pete Hegseth and National Security Advisor Michael Waltz, were involved.
02. **Information Shared:** Sensitive details regarding planned military strikes against Houthi targets in Yemen were discussed in the chat. This allegedly included the timing of attacks, weapons, and aircraft to be used.
03. **The Leak:** The information was leaked when journalist Jeffrey Goldberg of The Atlantic was added to the Signal group chat, reportedly by Waltz. Goldberg subsequently published details from the chat.
04. **Controversy:** There is debate over whether legally classified information was shared. The administration denies it, focusing blame on the journalist, while Democrats like Rep. Jim Himes called it a "brazen violation of laws" protecting national security. Security experts have questioned the use of a commercial app like Signal for such sensitive government communications.
05. **Reactions & Consequences:** The incident has sparked outrage, led to calls for the removal of the officials involved, prompted a government investigation into the security failure, and reportedly caused a spike in Signal downloads. The Defense Department had apparently warned staff about Signal's potential vulnerabilities shortly before the leak.

# Building a RAG model for others (e.g., company chatbot, teaching assistant)



Open source framework for building production-ready *LLM applications, retrieval-augmented generative pipelines* and *state-of-the-art search systems* that work intelligently over large document collections



- Retrievers
  - Document stores
  - Generators
- } Pipeline



Prompt: "I want to build a RAG model with Haystack. Let's do it step-by-step. I want to create a very basic RAG that uses as less dependencies as possible. It is for illustration purposes. First, we need to create the document store where the documents for the RAG are stored and retrieved from. The documents for the RAG are in the folder rag\_docs in this directory. Here is the documentation from Haystack: [COPY AND PASTE THE DOCUMENTATION]"



Prompt: "Before we continue with the retriever, please lets setup a requirements document."



Prompt: "Let's setup a virtual environment and activate it."

Then install requirements:

```
(rag_env) PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 2> pip install  
-r requirements.txt
```

 CURSOR

Prompt: "Let's continue with a basic retriever now."

 CURSOR

Prompt: "Let's continue with a basic generator now."

 CURSOR

Prompt: "I have this rag pipeline that I would like to test. I haven't run it yet. Please let me know how to run and test it with questions via the powershell terminal."

```
(rag_env) PS C:\Users\joerling\Dropbox\Workshop Oslo\Test project 2> python rag_app.py
```

# Some best practices and lessons learned...

- Use a consistent model version throughout your study to maintain reliability
- Track potential model updates and changes during the research
- Report all model settings (temperature, max. tokens etc.)
- Document the exact prompts, parameters, and system messages used
- Implement logging of all interactions for analysis
- Use the browser terminal for error logging (helps a lot when debugging the app)
- Think about the user and potentially adapt the interface (e.g., loading information)



# Other personal recommendations

- Going to developer meetups (Eventbrite, AI camp etc.)
- Interesting sources:
  - Great youtube content on LLMs: @AndrejKarpathy
  - Very extensive newsletter on AI developments: <https://buttondown.com/ainews>
  - Cursor AI community: <https://forum.cursor.com/>