

Group 5

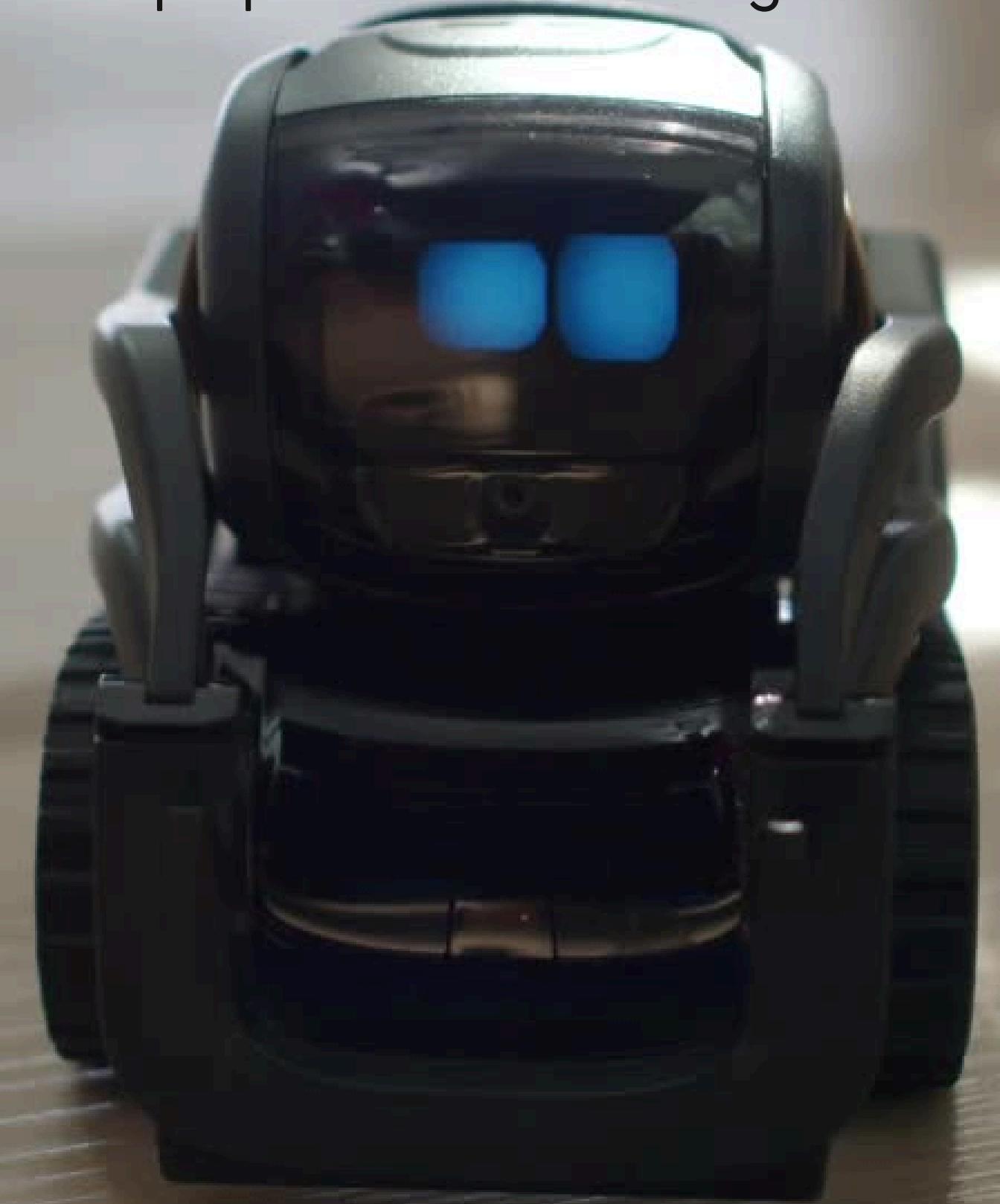
HI! I'M  
BAEMAX

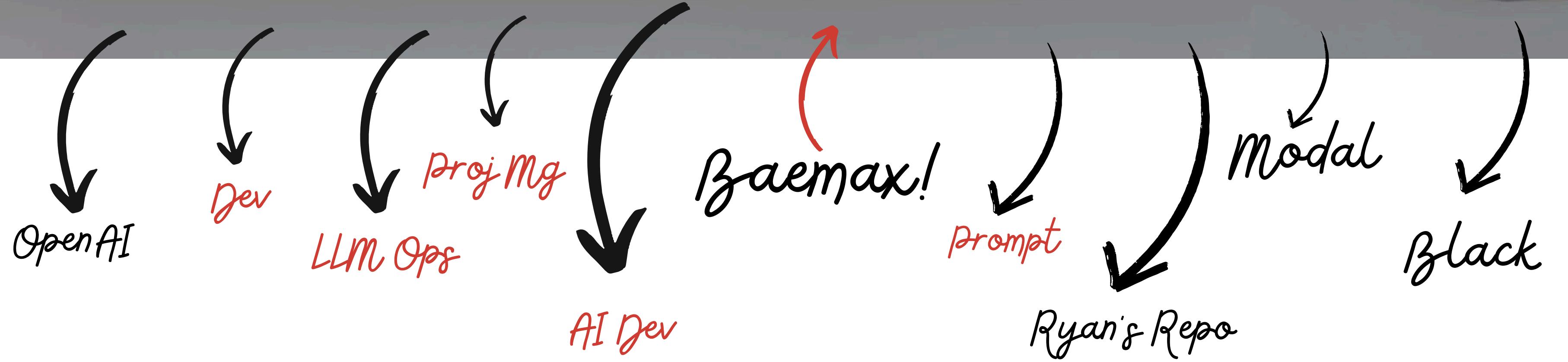
Your personal assistant :)



Disclaimer - Baymax is a Disney character from the movie Big Hero 6.  
These images are only for demonstration purposes and are being used for the presentation.

No harm was done to Baymax :)





# WHAT IS BAEMAX?.

Baemax is...

an LLM application designed to simplify complex Terms of Service (ToS) agreements. We use Retrieval Augmented Generation (RAG) and large language models (LLMs) to provide clear and concise explanations of legal jargon, making ToS documents easier to understand.

- Customer Value: User friendly, reliable and customizable
- Target User: Anyone who would like to learn about the terms and conditions they are agreeing to. Students, lawyers, business professionals, or even your neighbor!



# CUSTOMER VALUE

## Ease of Understanding

translates complex legal jargon into plain English

## Time Saving

summarizes lengthy ToS documents, saving users the time and effort of reading

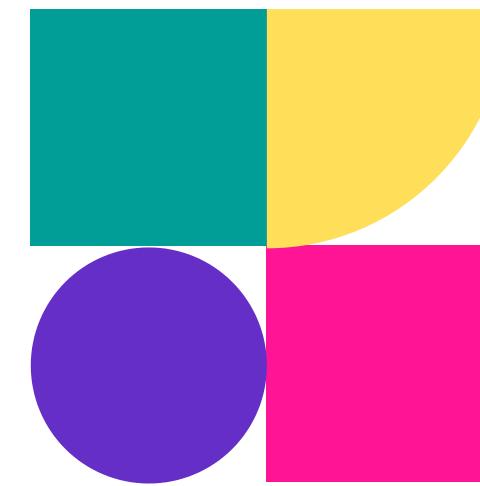
## Increased Transparency

by giving clear explanation, the app allows users to make informed decisions about the services they use

## Empower

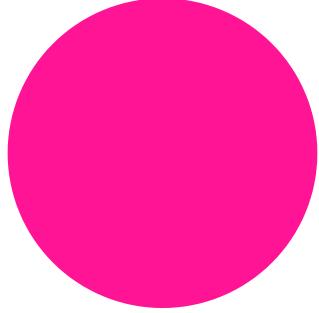
empowers users to confidently navigate the legal aspects of using online platforms and applications



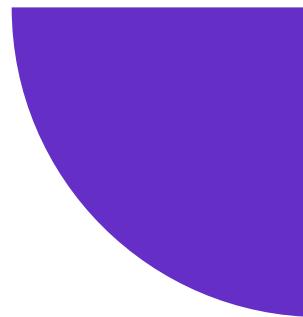


# techies assemble!

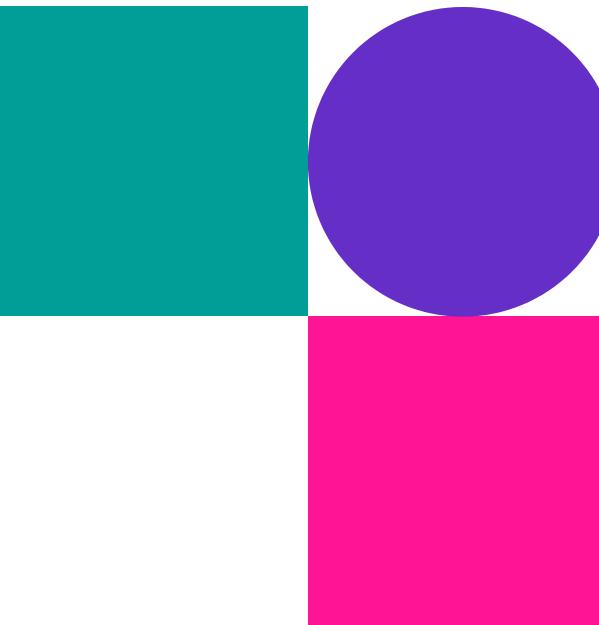
Loading....



**DATA INGESTION**



**RAG & EVALUATION**



**UI & DEPLOYMENT**



**USER TESTING**

**ENHANCEMENT**

# THE DATA

**WELLS FARGO**

---

- Terms of Service Agreements

[Personal](#)    [Privacy, Cookies, Security, and Legal](#)    [General Terms of Use](#)

## General Terms of Use

**PLEASE READ THESE TERMS OF USE CAREFULLY. BY ACCESSING THIS SITE OR USING THE SERVICES, YOU AGREE TO BE BOUND BY THESE TERMS OF USE. IF YOU DO NOT AGREE WITH THESE TERMS OF USE, PLEASE DO NOT ACCESS THIS SITE.**

You are currently viewing a page of the wellsfargo.com website or a related website or mobile application ("Site") belonging to Wells Fargo & Company or one of its direct or indirect subsidiaries ("Wells Fargo"). These General Terms of Use ("Terms of Use") govern your access to and use of the Site, including any content, functionality, and services offered on or through the Site ("Services"). "You" and "your" refer to any person who accesses or uses the Site or Services. The Site is intended for individuals who are at least 13 years old. If you are under the age of 13 years old, please do not access this Site.

Wells Fargo reserves the right to modify these Terms of Use at any time without notice. Any change to these Terms of Use will be effective upon posting of such updated Terms of Use on the Site. The most current version of the Terms of Use will be available to you by clicking the link at the bottom of the Site. By continuing to access or use the Services after the date of any change to these Terms of Use, you agree to be bound by such terms contained in the most recent version of these Terms of Use. **Wells Fargo reserves the right to modify or terminate the Services or to terminate your access to the Site, in whole or in part, at any time.**

# DATA INGESTION

---

≡ Google Cloud

- Web Scraping
  - BeautifulSoup
  - Google - API
  - Custom Search Engine
- Manual Addition

API APIs & Services

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
import pandas_datareader as pdr
import os
from urllib.parse import urlparse

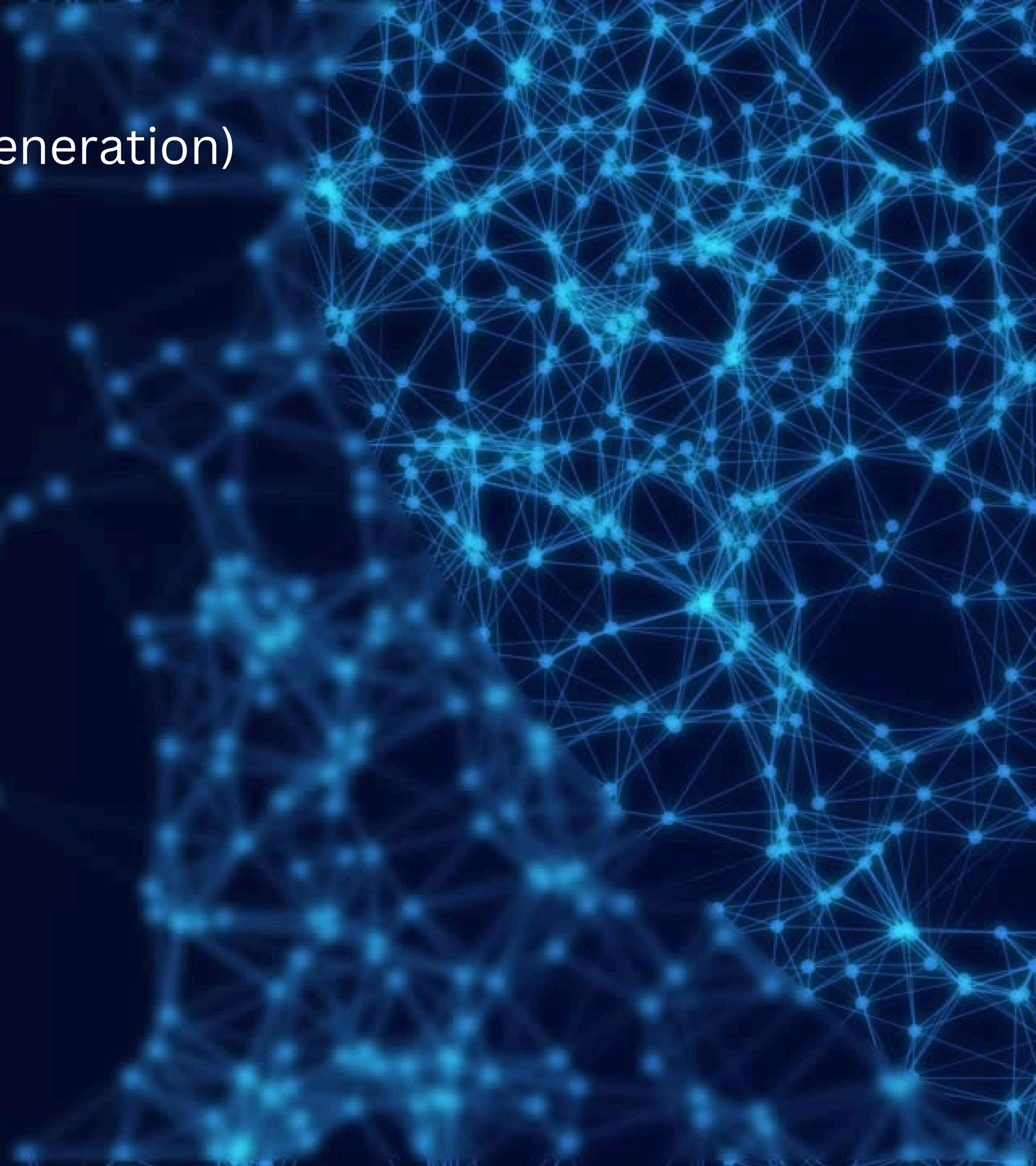
from googleapiclient.discovery import build
from googleapiclient.errors import HttpError
```

# RAG

(Retrieval-Augmented Generation)

# FLOW

- Utilizing a pre-existing vector store of embedded document chunks
- Retrieving the most relevant chunks based on the query.
- Using the retrieved context to generate augmented answers with OpenAI.



# Steps to create the vector store

## Code Breakdown

Here's the relevant code flow for vectorizing documents:

### 1. `initialize_vectorstore_with_metadata`:

- Loads metadata and processes documents.

```
python Copy code  
  
metadata_list = load_metadata(metadata_file)  
documents = create_documents_with_metadata(metadata_list, data_folder)  
embeddings = OpenAIEMBEDDINGS()  
return FAISS.from_documents(documents, embeddings)
```

### 2. `create_documents_with_metadata`:

- Reads files, splits content, and attaches metadata.

```
python Copy code  
  
for meta in metadata_list:  
    file_path = os.path.join(data_folder, meta["filename"])  
    if os.path.exists(file_path):  
        with open(file_path, "r", encoding="utf-8") as file:  
            text = file.read()  
            chunks = text_splitter.create_documents([text])  
            for chunk in chunks:  
                chunk.metadata = meta # Attach metadata  
            documents.extend(chunks)  
    else:  
        raise FileNotFoundError(f"File {file_path} not found.")  
return documents ↓
```

## Summary of RAG Initialization

- Metadata is loaded from `metadata.json`.
- Files in the `data` folder are processed:
  - Content is chunked.
  - Metadata is attached.
- Chunks are embedded into vector representations.
- The vector store is created using `FAISS`.
- A retriever is initialized from the vector store for answering queries.

# EVALUATING RAG USING DEEPEVAL

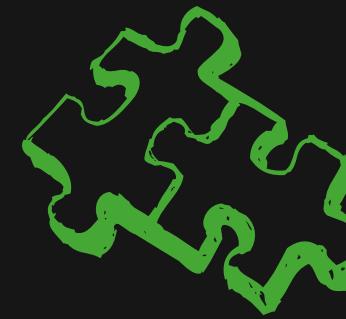
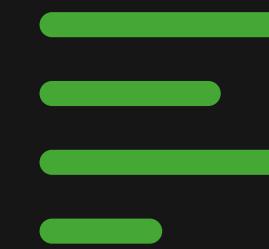
python

## Defining LLMTTestCase

```
LLMTestCase(  
    input=question,  
    expected_output=gt_answer,  
    actual_output=generated_answer,  
    retrieval_context=retrieved_document,  
)
```

## Evaluation Metrics

- **Correctness:** Evaluates whether the generated answer matches the expected output.
- **Faithfulness:** Checks if the generated answer aligns with the retrieved context.
- **Answer Relevancy:** Measures how closely the generated answer aligns with the user's query.



### Metrics Summary

- Correctness (GEval) (score: 0.6474614551196836, threshold: 0.5, strict: False, evaluation model: gpt-4o, reason: The actual output mentions location access, which aligns with the expected output, but includes additional permissions not specified in the expected output., error: None)
- Faithfulness (score: 1.0, threshold: 0.7, strict: False, evaluation model: gpt-4, reason: None, error: None)
- Answer Relevancy (score: 1.0, threshold: 0.7, strict: False, evaluation model: gpt-4o, reason: The score is 1.00 because the response was perfectly relevant and addressed the question without any irrelevant information. Great job!, error: None)

### For test case:

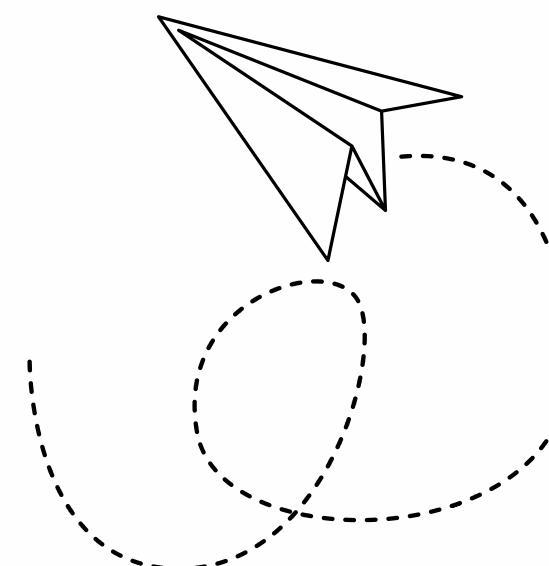
- input: What permissions might you grant when using Amazon-created apps?
- actual output: When using Amazon-created apps, you might grant the following permissions:
  - Access to your device's location services.
  - Permission to access your contacts.
  - Other permissions that your mobile device will inform you about.
  - The ability to change these permissions in your device's settings menu.
- expected output: Device-specific permissions like location or camera access.
- context: None
- retrieval context: ['We generally do not charge your credit card until after your order has entered the shipping process or, for digital products, until we make the digital product available to you.\n\nAPP PERMISSIONS\nWhen you use apps created by Amazon, such as the Amazon App or Kindle App, you may grant certain permissions to us for you to use on your behalf. These permissions are used to provide you with a better shopping experience and to help us improve our products and services. We will never share your personal information with third parties without your permission. You can always change or withdraw your consent at any time by updating your account settings or by contacting us directly. If you have any questions or concerns about our privacy practices, please feel free to contact us at any time. Thank you for choosing Amazon!']



# UI & DEPLOYMENT

Tying it all together to make Baemax!

- Creating a Front end
  - Streamlit UI
  - app.py
- Connecting the dots

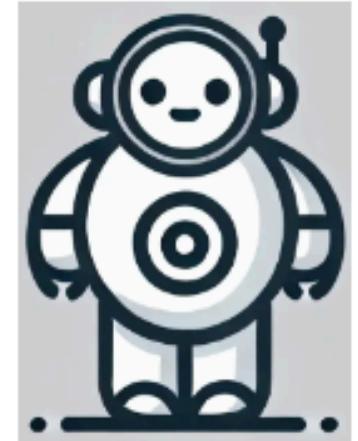


## Baemax T&C

An App to help demystify terms and conditions agreements.

Please upload a document or ask about the terms we have in our database.

What would you like to know? ➤



# UI & DEPLOYMENT

Tying it all together to make Baemax!

jupyter nbconvert --to python notebook.ipynb

This simple Jupyter  
command line function  
was the  
beginning of a beautiful  
transformation....



# UI & DEPLOYMENT

Tying it all together to make Baemax!

- Packaging it all up
  - `__init__.py`

The image shows a file explorer window on the left and a code editor window on the right.

**File Explorer:**

- frontend
  - streamlit
  - assets\images
- core
  - `__pycache__`
  - `__init__.py`
  - `helpers.py`
  - `rag.py`
- data
- app.py

**Code Editor:**

```
from .helpers import (
    QuestionAnswerFromContext,
    answer_question_from_context,
    bm25_retrieval,
    create_question_answer_from_context_chain,
    encode_from_string,
    encode_pdf,
    generate_document_summary,
    process_uploaded_tc,
    read_file_content,
    read_pdf_to_string,
    retrieve_context_per_question,
    show_context,
    text_wrap,
    retrieve_all_metadata
```

# WHAT'S NEW ?



- Front end app - Streamlit
- File upload feature
- Metadata
- rag.py, helpers.py and init.py
- We can interact and do user testing with Baemax 2.0 :D

What do you want to analyze?

Upload T&C document ?

Drag and drop file here  
Limit 200MB per file • PDF, DOCX, TXT

Browse files

Available Terms and Conditions

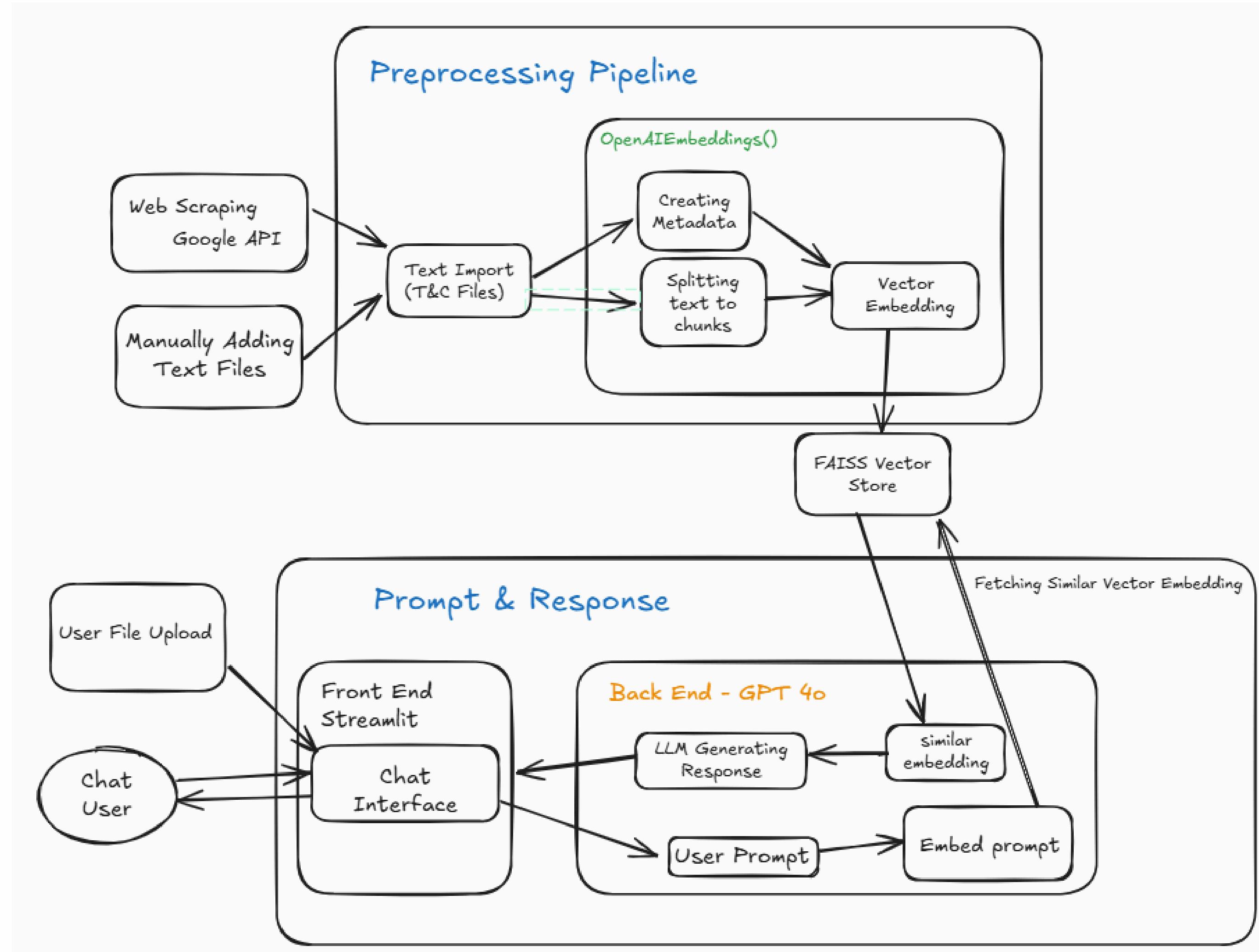
Qdoba

Select Model

gpt-4

Reset Chat

A screenshot of a Streamlit application interface. At the top, it asks "What do you want to analyze?". Below that is a file upload section with a placeholder "Drag and drop file here" and a note "Limit 200MB per file • PDF, DOCX, TXT", along with "Browse files" and "Available Terms and Conditions" buttons. A dropdown menu shows "Qdoba". Another dropdown menu shows "Select Model" with "gpt-4" selected. At the bottom is a "Reset Chat" button.



# USER TESTING & PROMPTING

baemax\_tc / frontend / app.py

Code Blame 384 lines (322 loc) · 12.4 KB

```
252     if "openai_model" not in st.session_state:
253         st.session_state["openai_model"] = "gpt-4"
254
255     # Initialize chat history
256     if "messages" not in st.session_state:
257         st.session_state.messages = []
258
259     # Display chat messages from history on app rerun
260     for message in st.session_state.messages:
261         with st.chat_message(message["role"]):
262             if message["role"] == "assistant":
263                 formatted_content = (
264                     message["content"].replace("\n*", "\n\n*").replace("\n-", "\n\n-")
265                 )
266                 st.markdown(formatted_content)
267             else:
268                 st.markdown(message["content"])
269
270     # System prompt
271 ~ system_prompt = """You are a helpful assistant with expertise in answering questions and providing insights.
272 When relevant context is provided, prioritize using it to answer the query.
273 If the question is unrelated to the context or no context is provided, respond based on your general knowledge.
274 If you do not know the answer, tell the user that their question is outside of your scope and do not lie.
275
276 You can:
277 - Answer questions using the provided context.
278 - Summarize information from context or general knowledge.
279 - Assist with general queries unrelated to the context.
280
281 Maintain a conversational and approachable tone. Always aim to provide clear, concise, and accurate answers."""
282
```

## SYSTEM PROMPT

# USER TESTING & PROMPTING

## FreeTaxUSA's Terms and Condition

 Tell me more about the terms and conditions of FreeTax USA.

### Available Terms and Conditions:

- <https://www.freetaxusa.com/terms> Terms of Use Last Updated: 10/31/2024 These Terms of Use (the "Terms of Use") are entered into by you and TaxHawk, Inc. ("TaxHawk", "we" or "us") concerning your access to and use of the [www.FreeTaxUSA.com](http://www.FreeTaxUSA.com) website. The following terms and conditions, together with any documents they expressly incorporate by reference (collectively, "Terms of Use"), govern your access to and use of the website, including any content, functionality, and services offered on or through, and any other media form, media channel, mobile application linked, or otherwise connected to [www.FreeTaxUSA.com](http://www.FreeTaxUSA.com) (collectively, the "Website"). By using the Website, you accept and agree to be bound and abide by these Terms of Use and our Privacy Policy, found at <https://www.FreeTaxUSA.com/privacy>, incorporated herein by reference.
- Privacy Policy All information we collect on this Website is subject to our Privacy Policy (<https://www.FreeTaxUSA.com/privacy>). By using the Website, you consent to all actions taken by us with respect to your information and agree to be bound by our Privacy Policy, which is incorporated into these Terms of Use. The Website is hosted in the United States and we make no claim that the Website or any of its content is accessible or appropriate outside of the United States. If you access the Website from outside the United States with laws or regulations governing personal data collection, use or disclosure that differ from applicable laws in the United States, you are transferring your data to the United States and you agree to have your data transferred to and processed in the United States. If you have any questions regarding privacy, please read our privacy policy: <https://www.FreeTaxUSA.com/privacy>.

## Comparison between burgerking and chick-fil-A

 comparsion between burgerking and chick-fil-A

 Based on the provided context, here are some comparisons between Chick-fil-A and Burger King:

#### Pricing:

- The prices for menu items displayed through the Chick-fil-A Online Services may differ from the prices offered or published by Operators for the same menu items at a physical Chick-fil-A restaurant location.
- Burger King also has variable pricing, with prices and participation varying across different U.S. locations.

#### Offers:

- Burger King has specific offers such as the "*5YourWayMeal*" and "*Whopper Jr. Duo for 5*". These offers are subject to terms and conditions, including geographic restrictions and the inability to make substitutions.
- The context does not provide specific information about Chick-fil-A's offers.

#### Refunds:

- Chick-fil-A has a strict no-refund policy for completed orders. While they have no obligation to provide refunds or credits, they may grant them at their sole discretion.
- The context does not provide specific information about Burger King's refund policy.

#### Delivery:

- Chick-fil-A reserves the right to change the prices displayed through the CFA Online Services and applicable fees or surcharges at any time. This likely includes delivery and service fees.
- Burger King mentions that prices may be higher for

# TIME TO WATCH A MOVIE!

The screenshot shows a Streamlit application window titled "Hi! I'm Baymax Paquet. - Presen...". The main content area is titled "baemax\_tc/frontend/data/link...". The application is "RUNNING..." and has "Stop" and "Fork" buttons.

**Powered by OpenAI**

Select Model: gpt-4

Reset Chat

OpenAI API Key: [REDACTED]

**What do you want to analyze?**

Upload T&C document: Drag and drop file here  
Limit 200MB per file • PDF, DOCX, TXT

Browse files

**Baemax T&C**

An App to help demystify terms and conditions agreements.

Please upload a document or ask about the terms we have in our database.

RAG system initialized successfully!

Icons at the bottom include a user profile, a crown, and a red button.



## OpenAI API Configuration

API key provided successfully!

Select Model

gpt-4

streamlitApp

Reset Chat



### Rate Limit Issue:

- ◆ You're running DeepEval's latest [Correctness \(GEval\) Metric](#)! (using gpt-4o, strict=False, async\_mode=True)
- ◆ You're running DeepEval's latest [Faithfulness Metric](#)! (using gpt-4, strict=False, async\_mode=True)
- ◆ You're running DeepEval's latest [Answer Relevancy Metric](#)! (using gpt-4o, strict=False, async\_mode=True)...

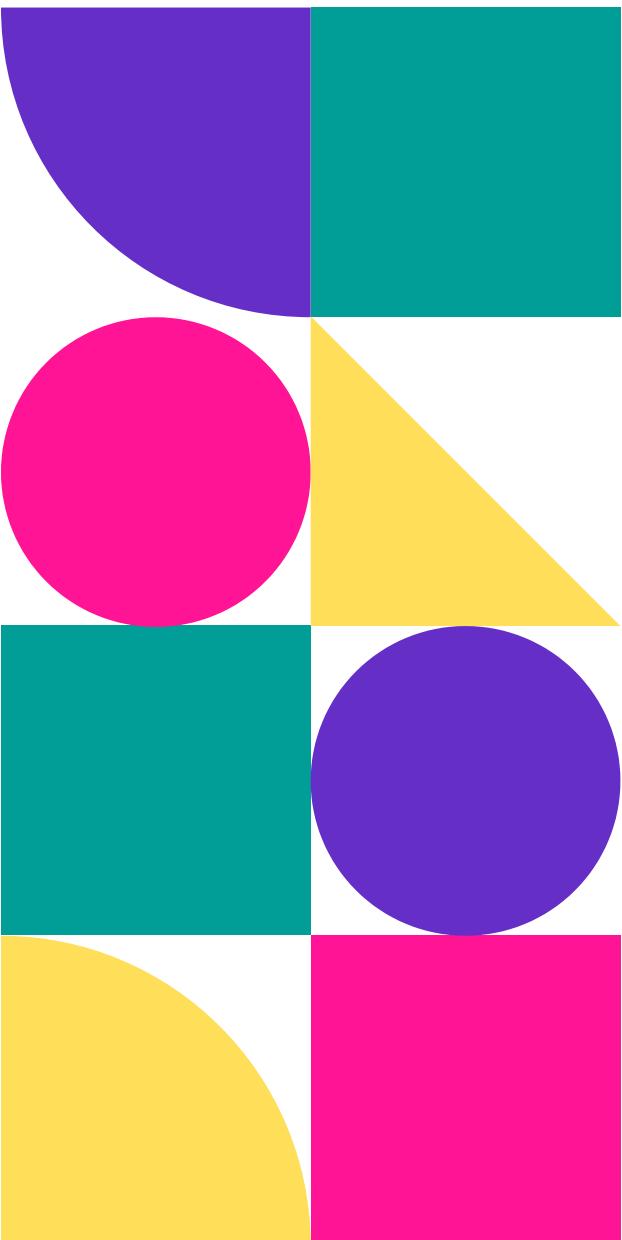
Event loop is already running. Applying nest\_asyncio patch to allow async execution...

```
Evaluating 10 test case(s) in parallel: |██████████| 10% (1/10) [Time Taken: 00:15, 15.32s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 1 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 1 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 20% (2/10) [Time Taken: 00:28, 14.02s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 2 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 1 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 1 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 30% (3/10) [Time Taken: 00:33, 10.04s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 2 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 2 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 2 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 40% (4/10) [Time Taken: 00:39, 8.48s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 2 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 2 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 50% (5/10) [Time Taken: 00:44, 7.15s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 3 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 3 time(s)...
ERROR:root:OpenAI rate limit exceeded. Retrying: 3 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 60% (6/10) [Time Taken: 00:54, 8.10s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 3 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 70% (7/10) [Time Taken: 00:57, 6.56s/test case]ERROR:root:OpenAI rate limit exceeded. Retrying: 4 time(s)...
Evaluating 10 test case(s) in parallel: |██████████| 100% (10/10) [Time Taken: 01:19, 7.96s/test case]
```

# BAEMAX CHALLENGES

- Rate Limiting Issue
- Session state of API Key
- Adding functionalities
- Data Leakage
- Relevant context
- Choice of LLM - Llama, Modal, Gemini
- RAG





How did we overcome these challenges?

**WE WORK  
TOGETHER!**



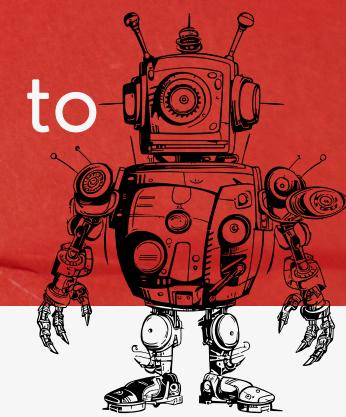
# HOW CAN WE IMPROVE BAEMAX?

- Extend our Vector DB
- Automate Data Ingestion
- Upload docs - keep it going!
  - Ask the user if they want us to keep their ToS

# DATA FLYWHEEL

**Personal Chatgpt but for  
ToS**

Automate our data collection to  
inlcude more data



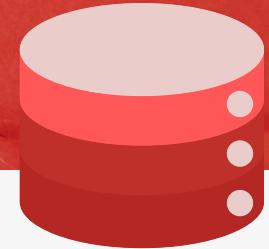
**Multilingual**

Data should not be limited to English  
but the app should take on any  
language



**Uploaded Doc Stays in DB**

The data being uploaded should  
have the option to load into the  
vector DB



**Better UX/UI?**

You tell us..what would make our  
app, your app?



How would you design a comprehensive human evaluation framework that balances efficiency with qualitative insight? What specific aspects of the responses would you prioritize for human review, and how would you structure this feedback loop to improve the system? Ideally think of how to tie this into unit tests and evaluation metrics.

- A/B testing - Try many RAG's
  - Re-ranking
- Thumbs up and down like feature
- Human evaluation
  - Prompting



**LET'S PLAY  
QUESTION &  
ANSWER**

**WE ARE SATISFIED  
WITH OUR CARE :)**

Happy Graduation and Happy Holidays!

