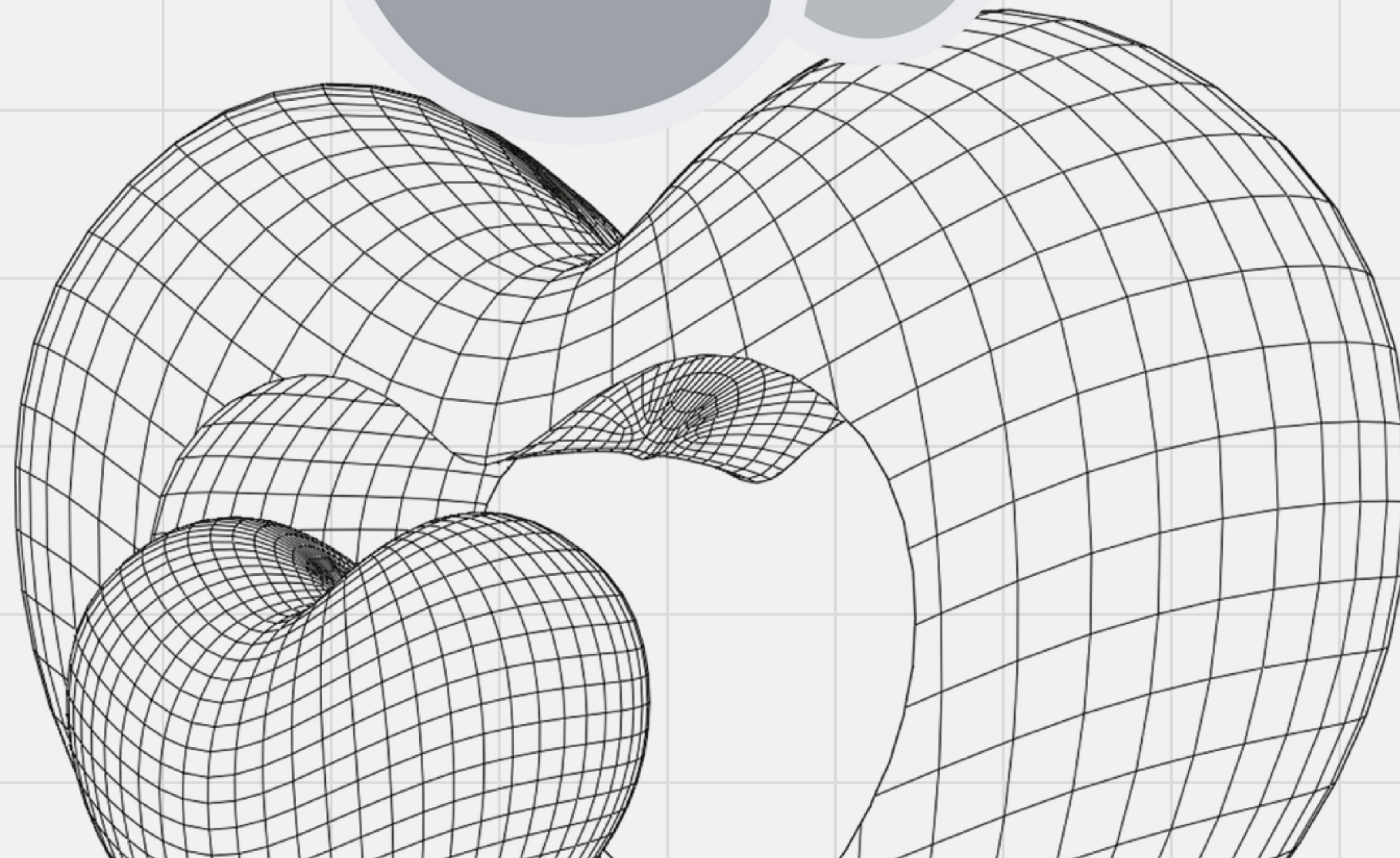


# ElasticRAG

HARNESSING AI-POWERED NLP FOR SEAMLESS ELASTICSEARCH DATA RETRIEVAL!





# TABLE OF CONTENT

1

OVERVIEW

2

RESOURCES

3

PROCESS

4

OUTCOME



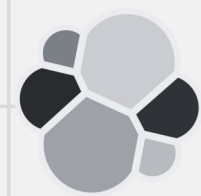
# OVERVIEW

IMAGINE QUERYING VAST AMOUNTS OF INDEXED DATA IN ELASTICSEARCH WITH THE SIMPLICITY OF ASKING A QUESTION! WITH THE LLAMA3 8B MODEL, YOU CAN NOW RETRIEVE DATA EFFORTLESSLY USING NATURAL LANGUAGE. NO MORE COMPLEX QUERIES, JUST ASK, AND THE INSIGHTS FLOW DIRECTLY FROM ELASTICSEARCH. DIVE INTO A SEAMLESS DATA INTERACTION EXPERIENCE AND WATCH YOUR PRODUCTIVITY SOAR. GET READY TO REVOLUTIONIZE THE WAY YOU INTERACT WITH DATA!

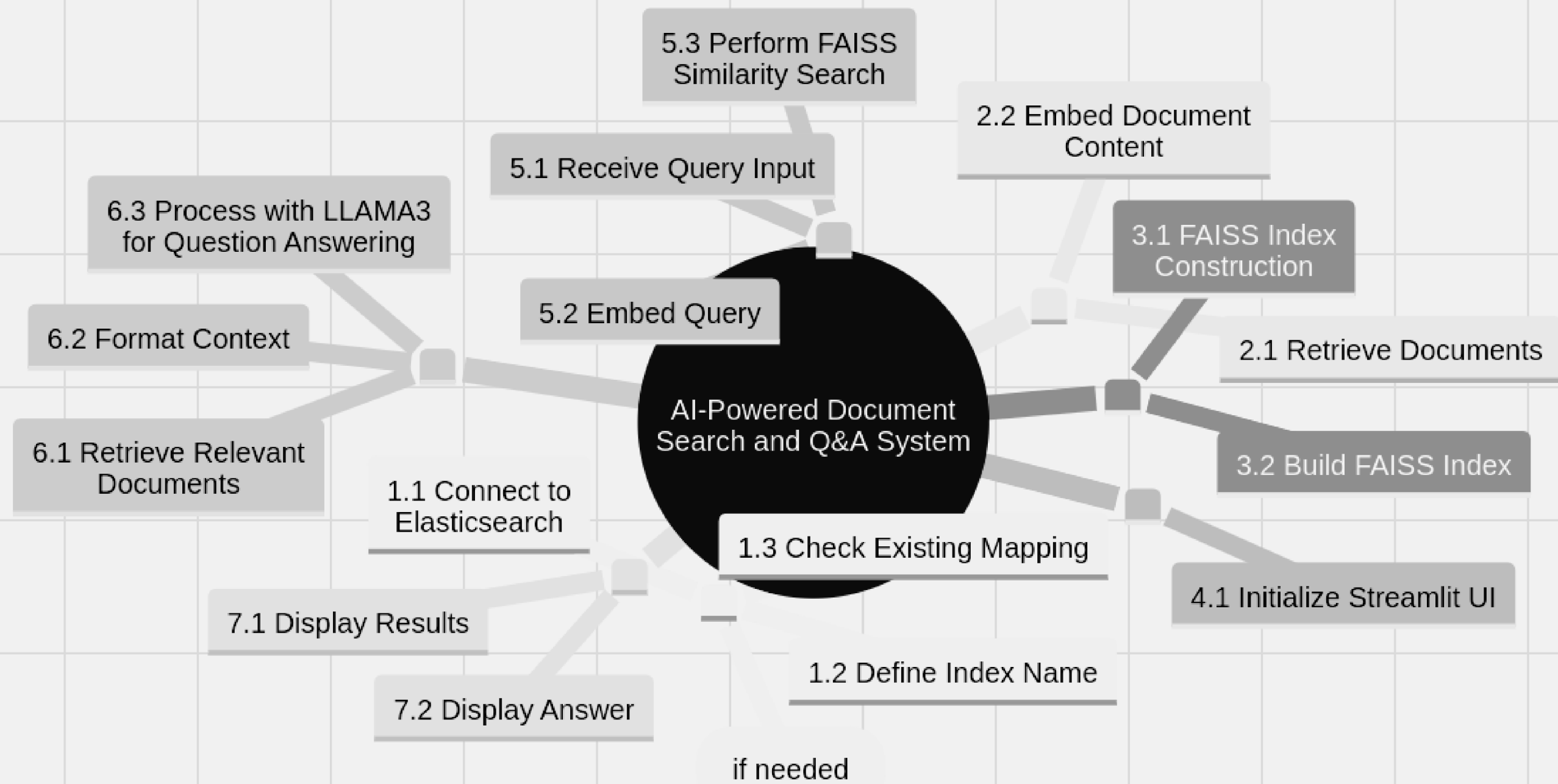


# RESOURCES

- OLLAMA LLAMA3(LLM)
- ELASTICSEARCH(ENGINE)
- KIBANA(UI FOR ELASTIC)
- LANGCHAIN
- OLLAMA NOMIC EMBED TEXT(EMBEDDER)
- FAISS
- PYTHON
- STREAMLIT(UI)



# PROCESS





# PROCESS MAP DESCRIPTION

## INITIALIZATION:

CONNECT TO ELASTICSEARCH: ESTABLISH CONNECTION TO ELASTICSEARCH.  
DEFINE INDEX NAME: SPECIFY THE ELASTICSEARCH INDEX NAME.

## MAPPING CREATION:

CHECK EXISTING MAPPING: DETERMINE IF THE INDEX MAPPING EXISTS IN ELASTICSEARCH.  
CREATE DYNAMIC MAPPING: IF NECESSARY, CREATE OR UPDATE THE MAPPING BASED ON SAMPLE DATA (CREATE\_DYNAMIC\_MAPPING).

## DOCUMENT RETRIEVAL AND EMBEDDING:

RETRIEVE DOCUMENTS: FETCH ALL DOCUMENTS FROM ELASTICSEARCH.  
EMBED DOCUMENT CONTENT: USE OLLAMA EMBEDDINGS TO EMBED DOCUMENT CONTENT.

## FAISS INDEX CONSTRUCTION:

BUILD FAISS INDEX: CONSTRUCT A FAISS INDEX USING DOCUMENT EMBEDDINGS.

## USER INTERFACE:

STREAMLIT UI: INTERACT WITH USERS THROUGH A STREAMLIT WEB INTERFACE.



# PROCESS MAP DESCRIPTION

## QUERY PROCESSING:

QUERY INPUT: RECEIVE USER QUERY INPUT.

EMBED QUERY: EMBED THE QUERY USING CACHED EMBEDDING FUNCTION.

## SIMILARITY SEARCH:

FAISS SEARCH: UTILIZE FAISS FOR SIMILARITY SEARCH BASED ON QUERY EMBEDDINGS (FAISS\_SIMILARITY\_SEARCH).

## ANSWER GENERATION:

RETRIEVE RELEVANT DOCUMENTS: RETRIEVE RELEVANT DOCUMENTS BASED ON SIMILARITY SCORES.

FORMAT CONTEXT: COMBINE RETRIEVED DOCUMENTS INTO A FORMATTED CONTEXT FOR LLAMA3 INPUT.

## LLAMA3 QUESTION ANSWERING:

LLAMA3 INTERACTION: INTERACT WITH LLAMA3 MODEL FOR QUESTION ANSWERING.

## DISPLAY RESULTS:

DISPLAY ANSWER: PRESENT THE GENERATED ANSWER OR RESPONSES TO THE USER VIA THE STREAMLIT INTERFACE.





# OUTCOME

## ElasticRAG

Ask me something...

UI WITH PROMPT  
HERE THE INDEX TO BE SEARCHED IS  
ALREADY LOADED USING THE PYTHON  
CODE, NOW ALL WE HAVE TO DO IS  
ASK QUESTIONS BASED ON THE DATA.

## ElasticRAG

Ask me something...

pull up june data

To pull up the June data, I'll extract the records with dates that fall within the m

```
1. **2024-06-10T00:00:00.000-04:00**
  * "filename": "report_june_10.pdf"
  * "filecontent": "This is the content of the report for June 10, 2024. It in
2. **2024-06-11T00:00:00.000-04:00**
  * "filename": "summary_june_11.txt"
  * "filecontent": "Summary of events and activities for June 11, 2024. Highli
3. **2024-06-12T00:00:00.000-04:00**
  * "filename": "meeting_minutes_june_12.docx"
  * "filecontent": "Minutes of the meeting held on June 12, 2024. Includes age
4. **2024-06-13T00:00:00.000-04:00 (twice)**
  * First occurrence:
    + "filename": "presentation_june_13.pptx"
    + "filecontent": "Presentation slides for the June 13, 2024 meeting.
  * Second occurrence:
    + Same file information as above
```

I ASKED THE RAG TO PULL JUNE DATA AND IT  
PULLED IT PERFECTLY!





# THANK YOU

THE PROJECT WAS DONE IN A PERIOD OF 2 WEEKS,  
FROM SCRATCH BY ME(AN AI & ML STUDENT), WITH  
NO PRIOR KNOWLEDGE, IT WAS FUN AND  
CHALLENGING TO ACCOMPLISH THIS.

I WILL LINK BELOW THE RESOURCES AND YOUTUBE  
CHANNELS THAT HELPED ME LEARN AND COMPLETE THE  
PROJECT.

