



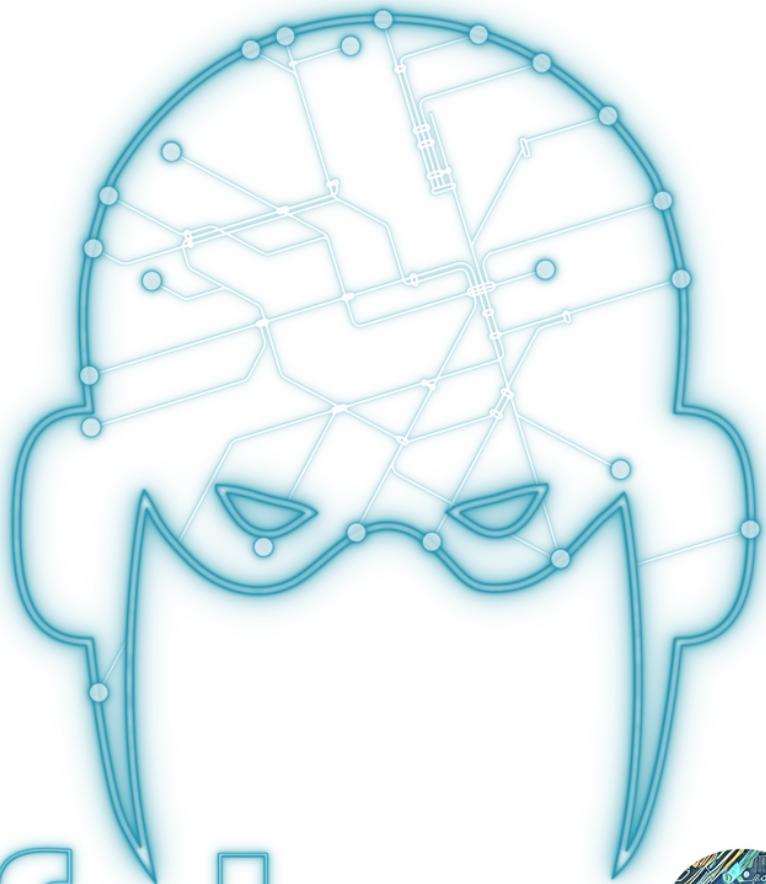
Creating a Semantic Search-Based Question-Answering System with LLMs



Jettro Coenradie



Daniël Spee



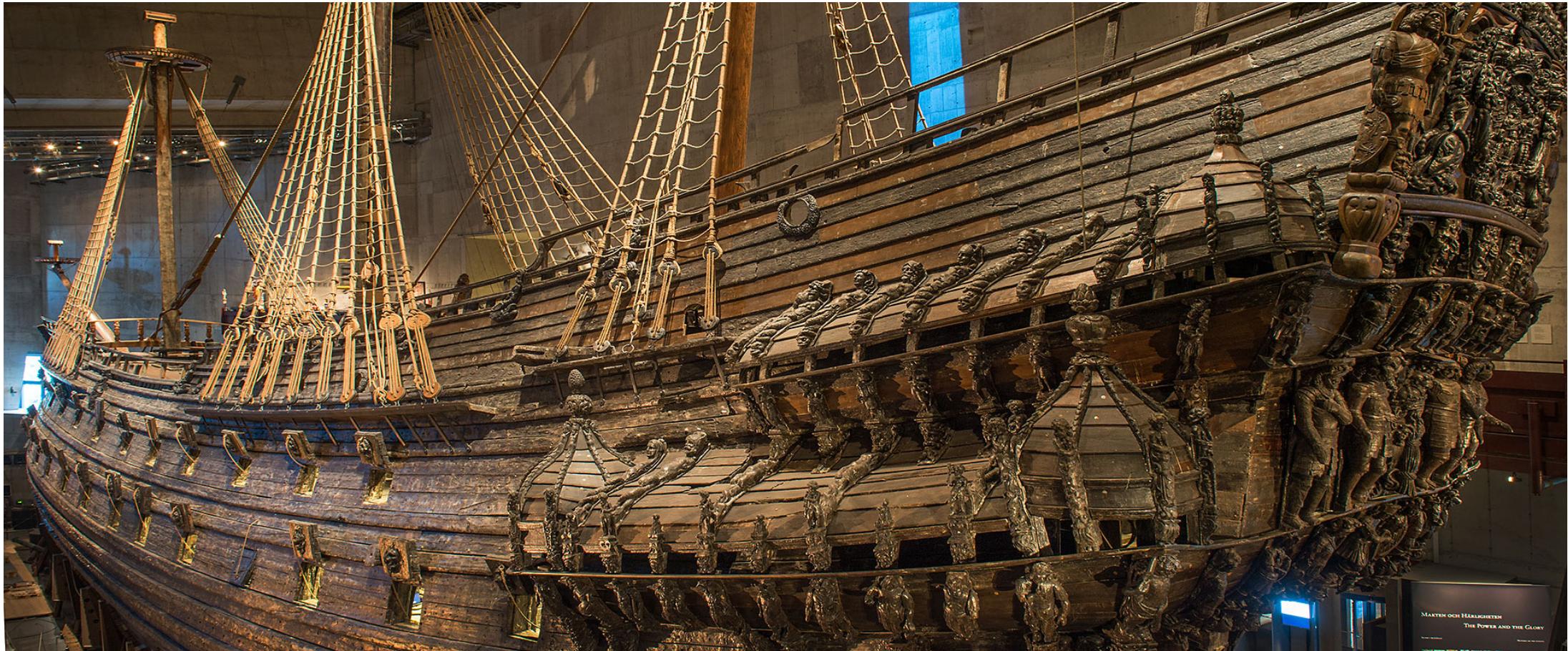
Jfokus 24

luminis. / part of
yuma



luminis. / part of
yuma

The Vasa Warship



<https://www.vasamuseet.se/en>

luminis. / part of
quma

Daniël Spee

Search Engineer @ Luminis



Jfokus 24

 <https://www.linkedin.com/in/dspee/>

 <https://github.com/Daantie>

 <https://www.luminis.eu/expert/daniel-spee/>

 daniel.spee@luminis.eu

luminis.

Jettro Coenradie

Data Evangelist @ Luminis



Jfokus 24

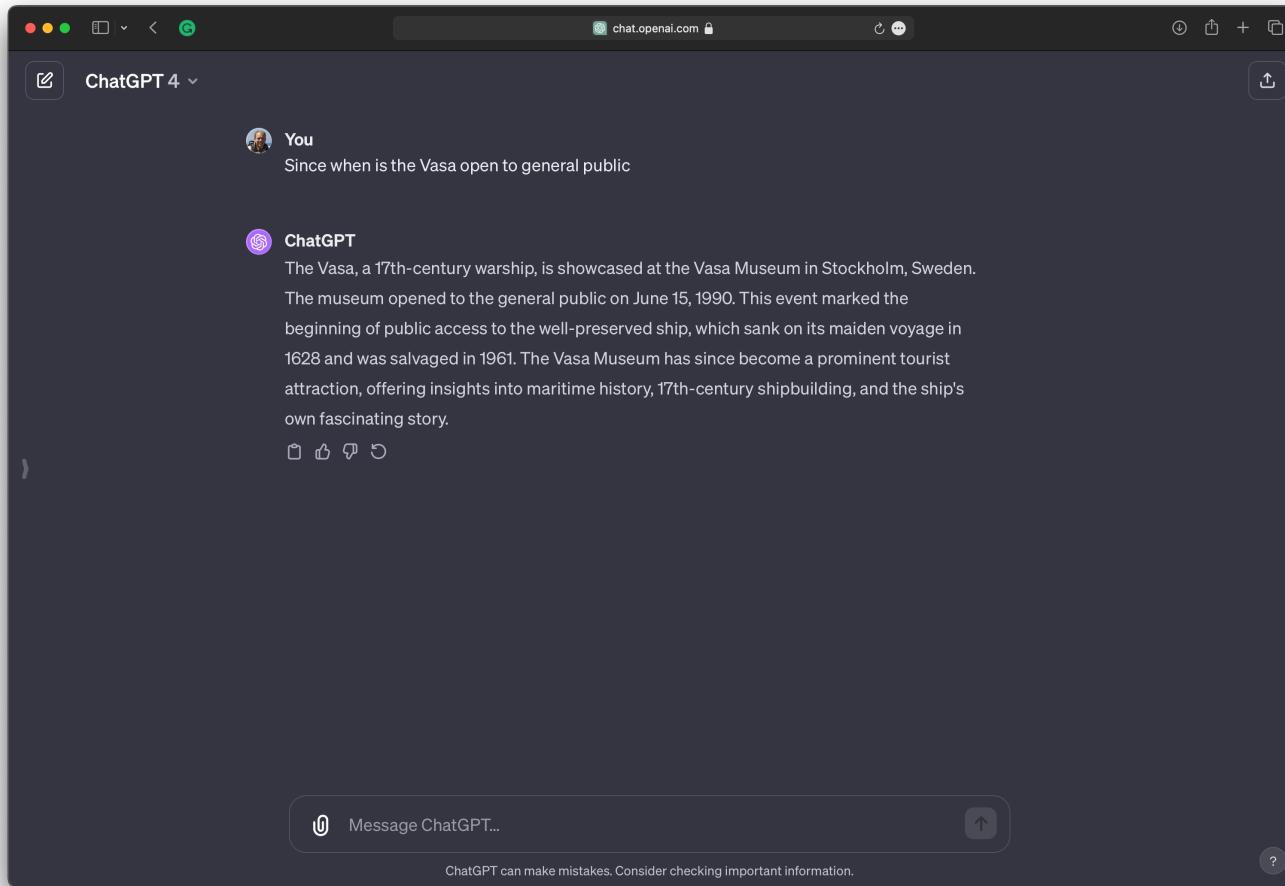
 <https://www.linkedin.com/in/jettro>

 <https://github.com/jettro>

 <https://www.luminis.eu/expert/jettro-coenradie/>

 Jettro.coenradie@luminis.eu

luminis.



luminis.

The screenshot shows the homepage of the Vasa Museum website (vasamuseet.se) displayed on a Mac OS X desktop. The page features a large banner image showing a crowd of people in dark coats and hats gathered around a wooden railing, likely at the opening ceremony in 1962. Overlaid on this image is a white rectangular box containing text and a downward-pointing arrow. At the top of the page, there is a navigation bar with links for 'Visit', 'Explore' (which is underlined), and 'Buy ticket'. There are also links for 'Press', 'Browsealoud', and 'Choose language'. The Vasa Museum logo is in the top left corner, and a 'MENU' icon is in the top right corner.

February 1962

Vasa meets the public

By Friday 16 February 1962, the ship is ready to be displayed to the general public at the newly-constructed Wasa Shipyard, where visitors can see Vasa while a team of conservators, carpenters and other technicians work to preserve the ship. The museum opens with a salute from two of Vasa's cannon. Public interest is enormous and success is immediate – in 1962, 439,300 buy a ticket to see the ship and its unique finds.

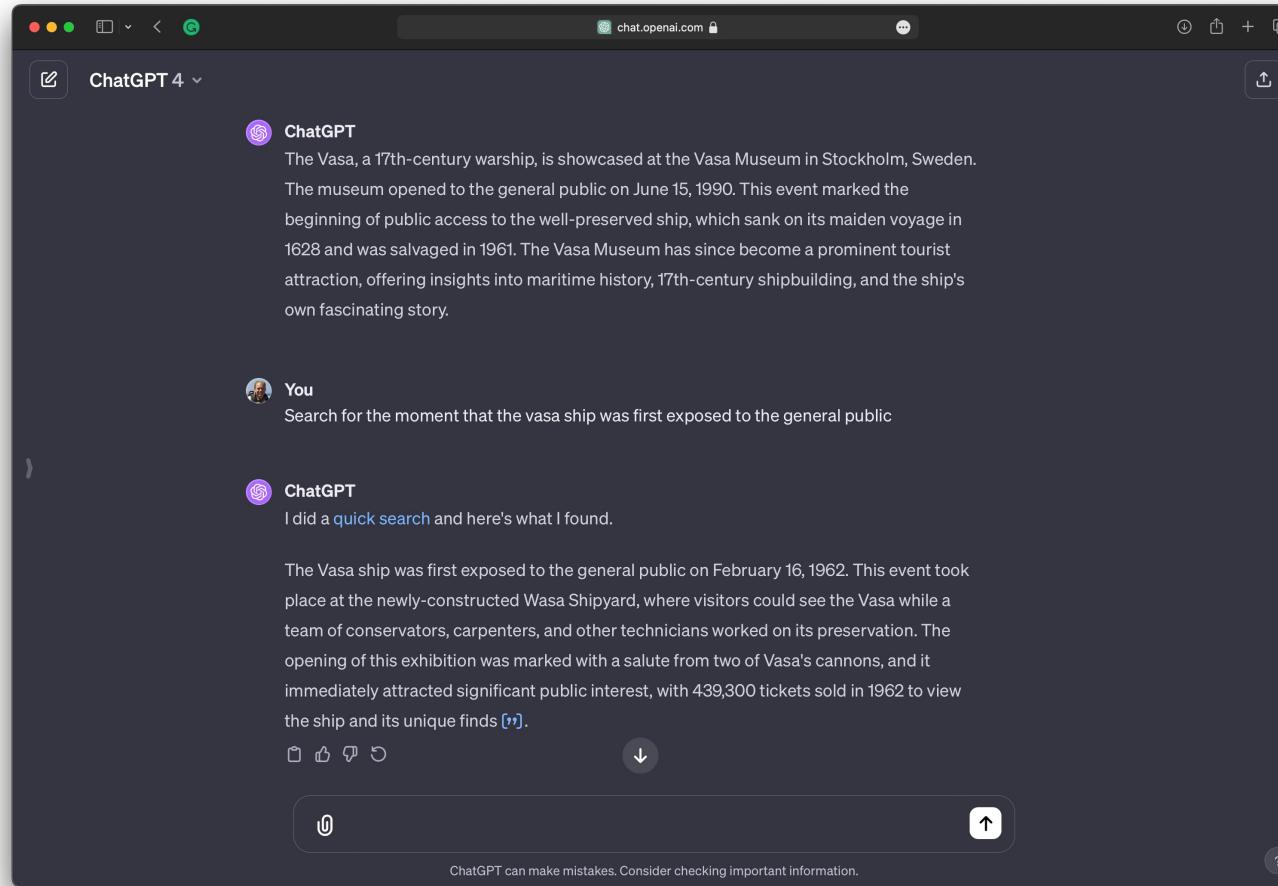
[Continue to April 1962](#)

luminis.

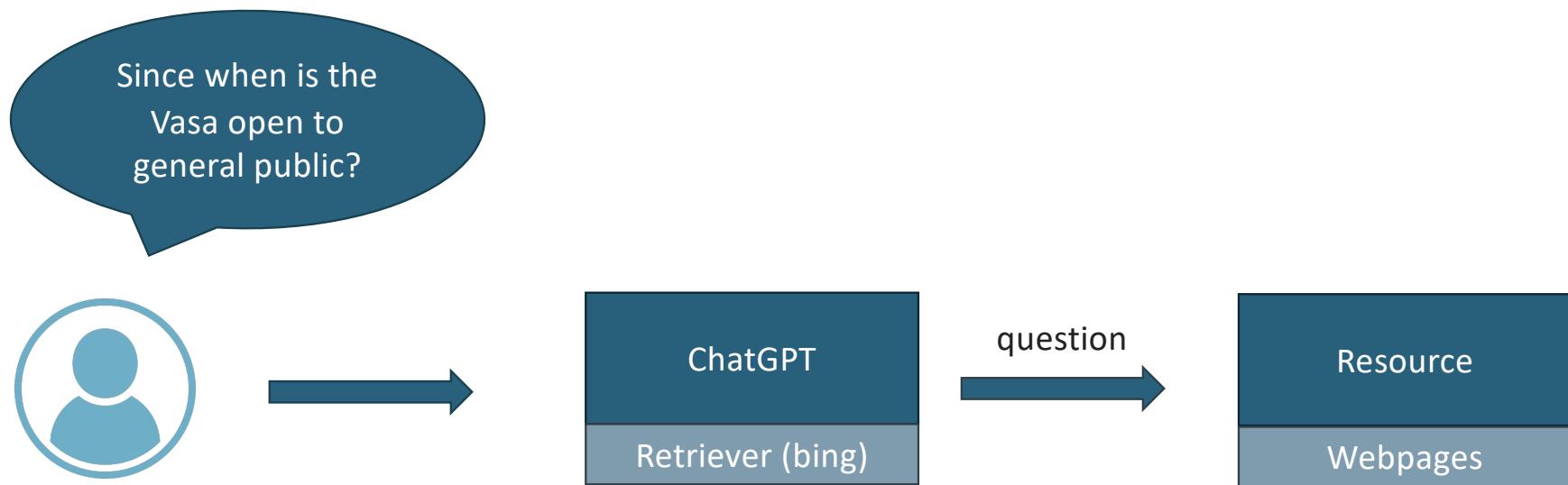
A screenshot of the ChatGPT 4 interface on a dark-themed web browser window. The window title is "ChatGPT 4". The URL bar shows "chat.openai.com". The main conversation area has the following messages:

- You**: Since when is the Vasa open to general public
- ChatGPT**: The Vasa, a 17th-century warship, is showcased at the Vasa Museum in Stockholm, Sweden. The museum opened to the general public on June 15, 1990. This event marked the beginning of public access to the well-preserved ship, which sank on its maiden voyage in 1628 and was salvaged in 1961. The Vasa Museum has since become a prominent tourist attraction, offering insights into maritime history, 17th-century shipbuilding, and the ship's own fascinating story.
- You**: Search for the moment that the vasa ship was first exposed to the general public
- ChatGPT**: (Thinking) Searching “Vasa ship first exposed to general public date”

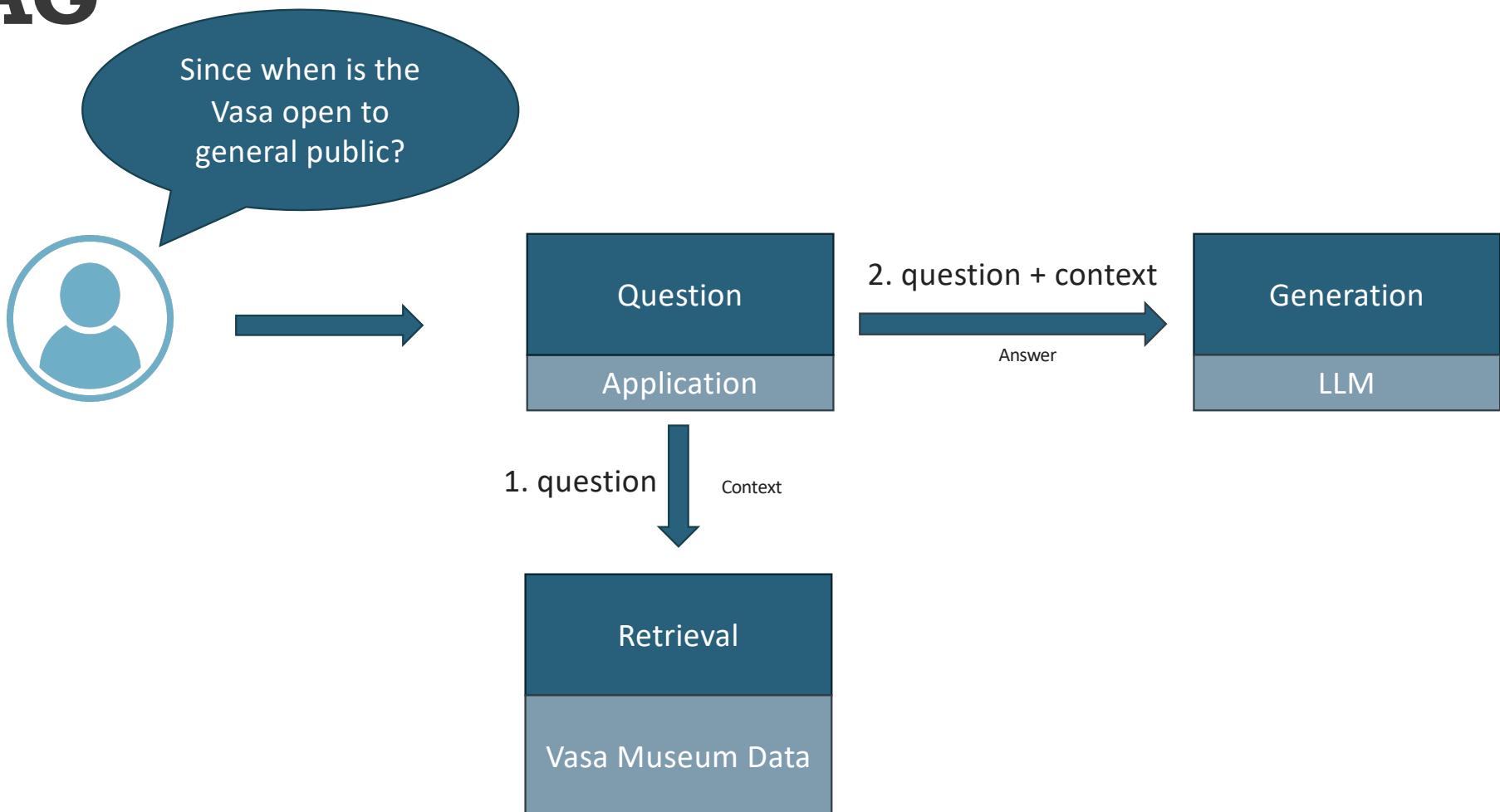
The bottom of the window features a footer bar with icons for microphone, camera, and question mark, and the text "ChatGPT can make mistakes. Consider checking important information."



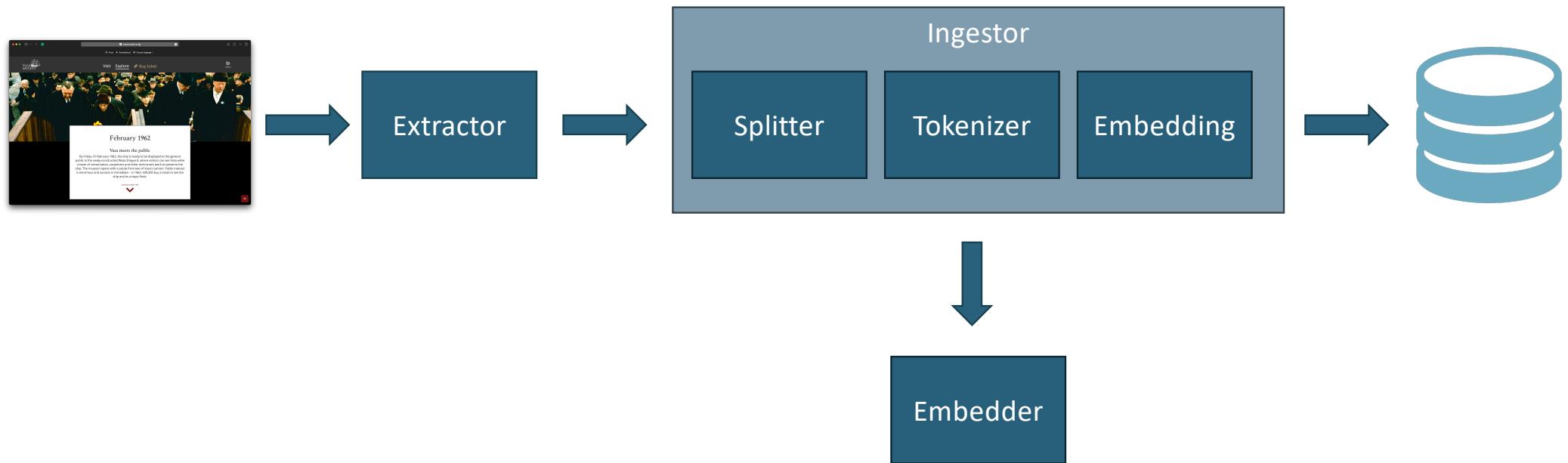
Retrieval Augmented Generation (RAG)



RAG



Vector Store ~ Ingestion



Workshop agenda



Setting up your environment



Splitting text into chunks



Embedding chunks



Storing embedding and data in a content store

Use synthetic questions to validate retrieval quality.



Generate an answer to a question from a context

Use LLM to validate the quality of the answer

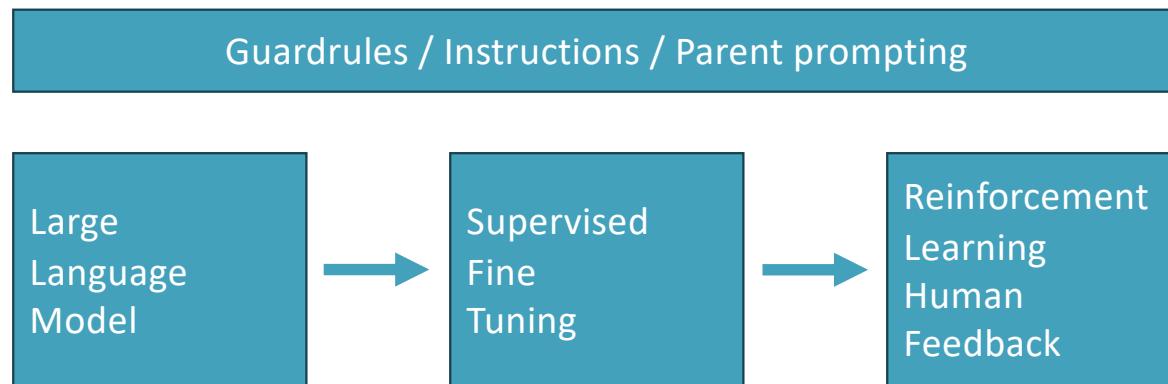


Improve retrieval strategy

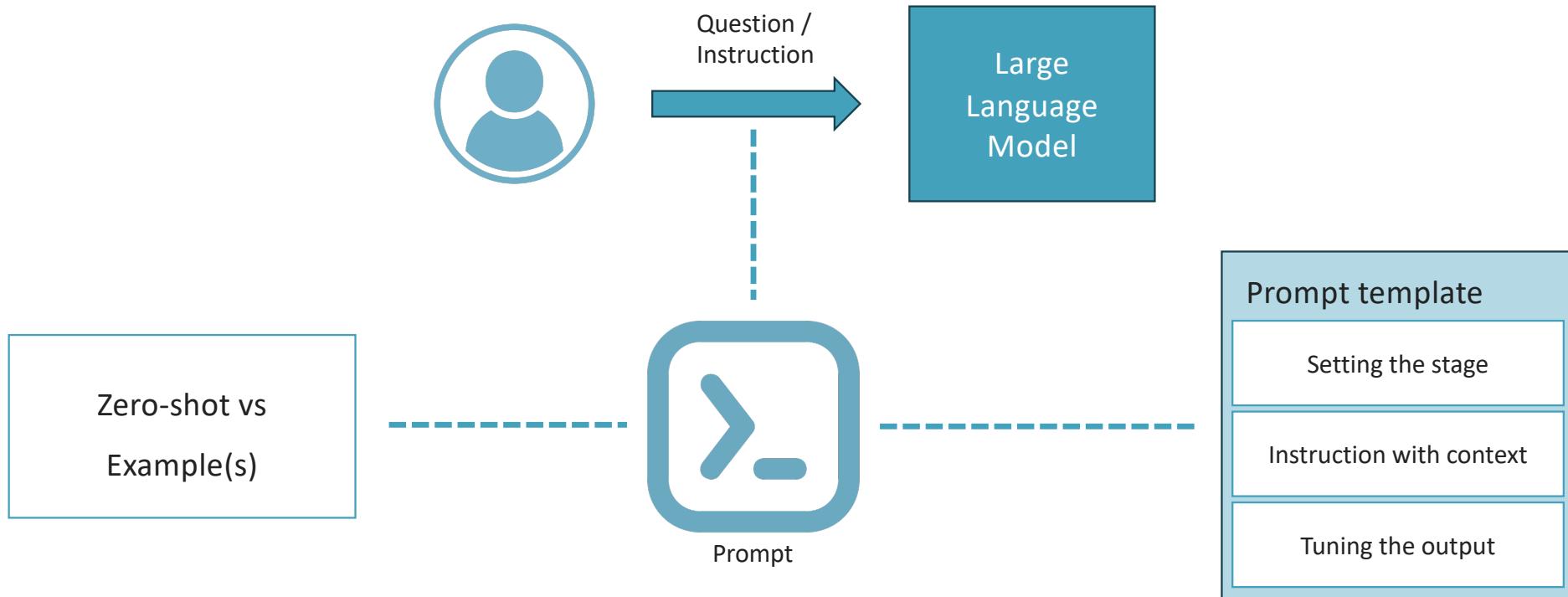


All together now

LLM – Large Language Model



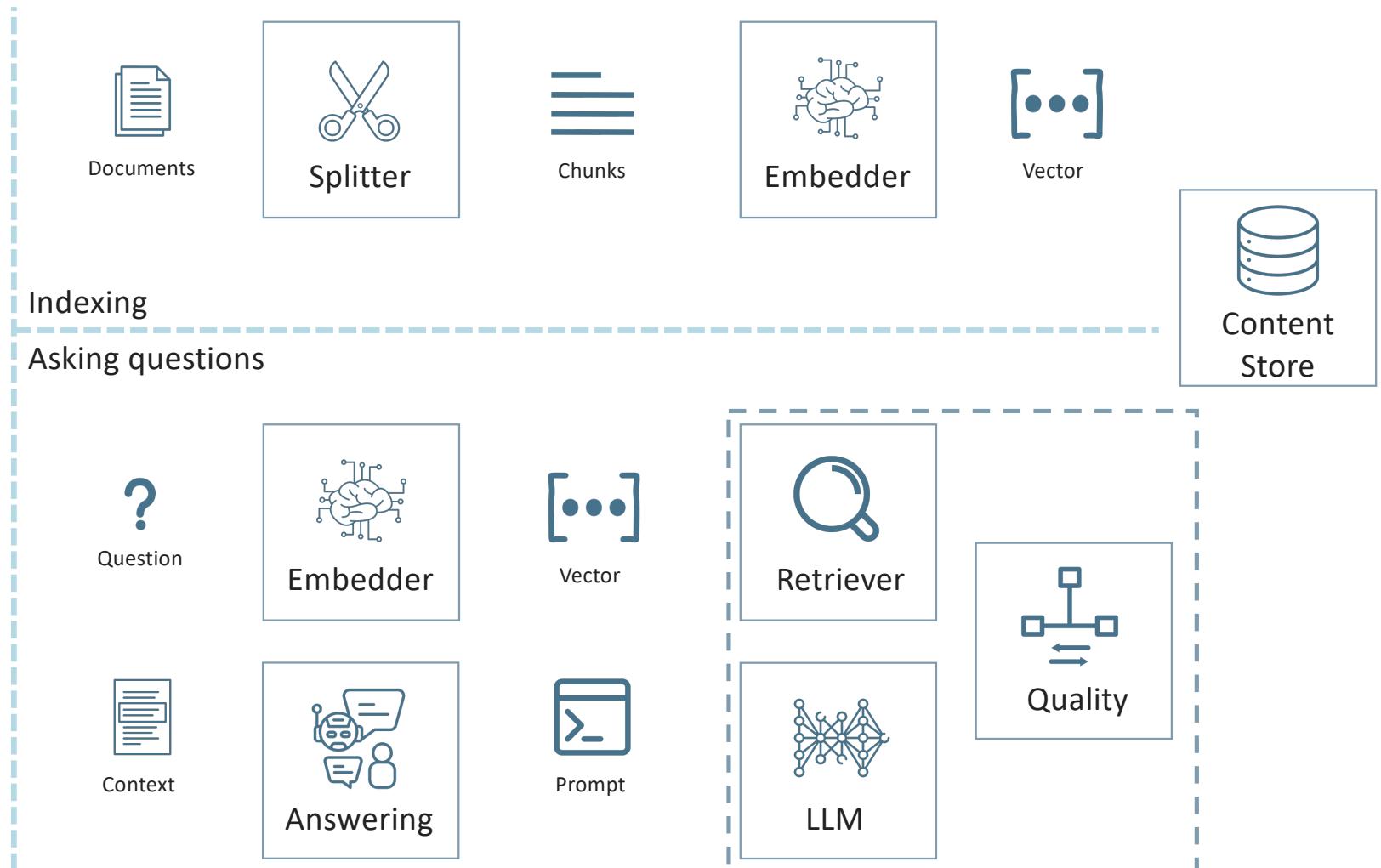
Prompting





Java: AppStepN...

Python: app_stepN_...



Chunking Strategy

Splitting text into chunks

Why is splitting necessary?

Impact of different splitters

Examples splitters

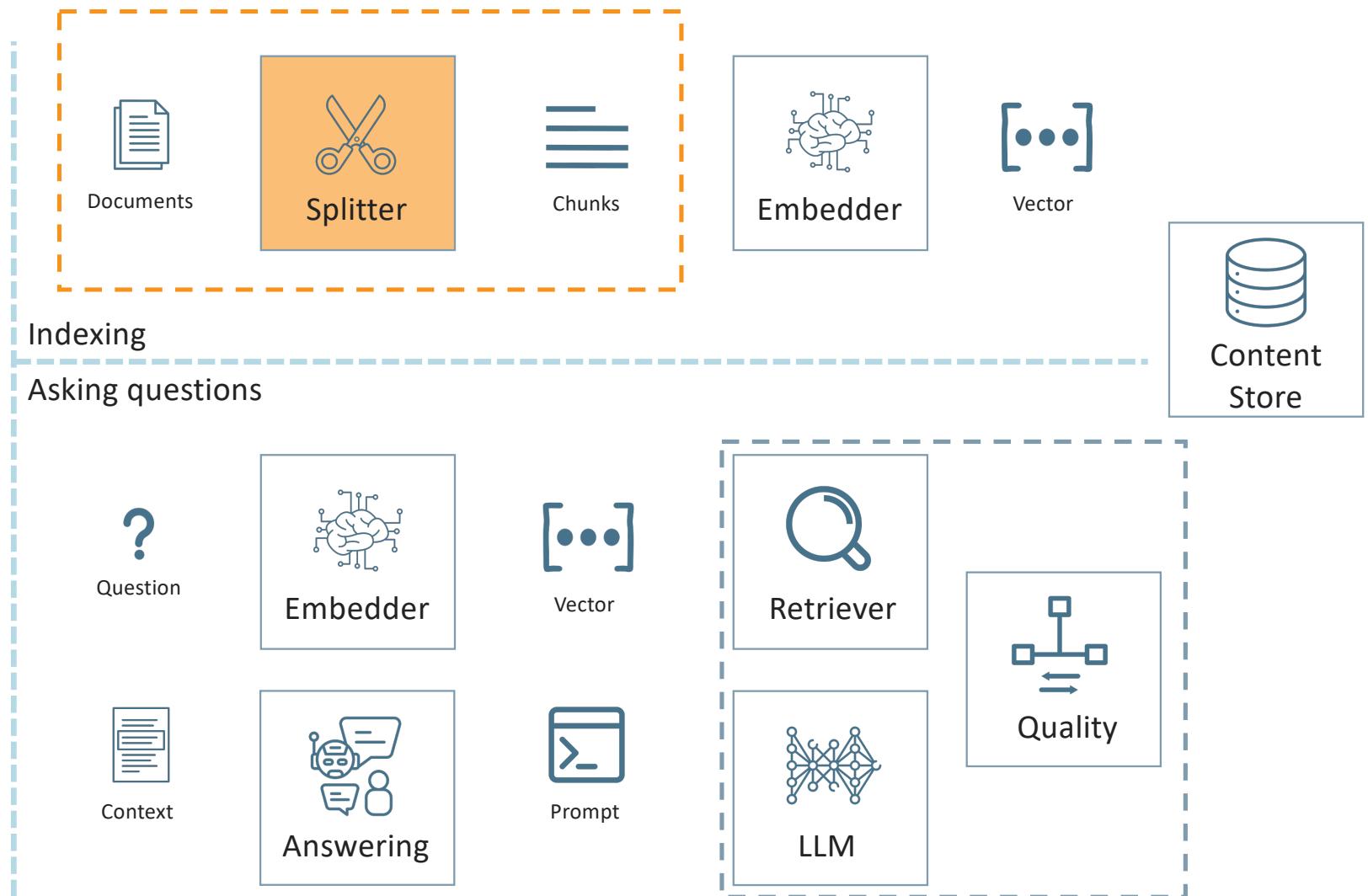


DALL E - visualize a splitter that splits text into chunks



Java: AppStepN...

Python: app_stepN_...





<https://rag4j.org>

Workshop jFokus 2024

<https://github.com/RAG4J/rag4j-jfokus>

<https://github.com/RAG4J/rag4p-jfokus>

secret_key=

J: AppStep1ChunkingStrategy

P: app_step1_chunking_strategy

Embeddings

Vectorisation

From text to numbers

- Dictionaries of tokens
- Sparse vectors
- One hot encoding

Dictionary	programming	1	0	0	0	0
search	0	1	0	0	0	0
data	0	0	1	0	0	0
language	0	0	0	1	0	0
pipeline	0	0	0	0	1	0

Document	programming	1	0	0	0	0
your	0	0	0	0	0	0
data	0	0	1	0	0	0
Pipeline	0	0	0	0	1	0

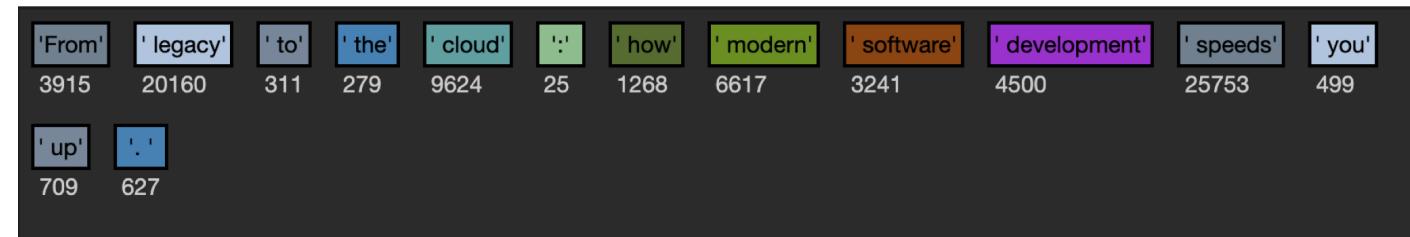
Query	programming	1	0	0	0	0
language	0	0	0	1	0	0

Dictionaries

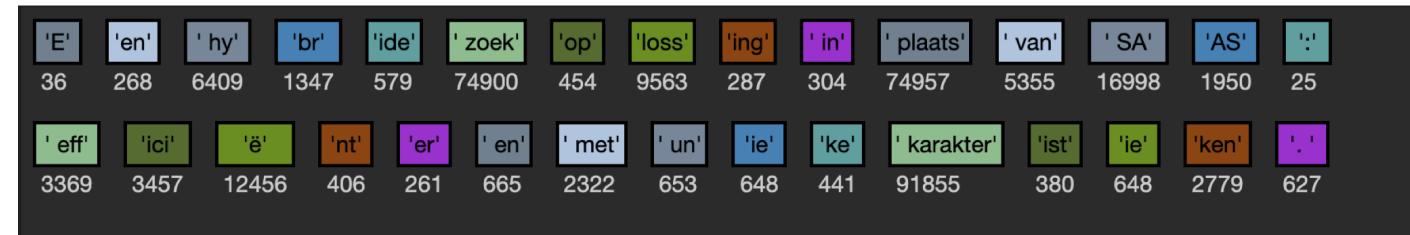
From text to numbers

- Tokenisation
- Wordparts
- GPT-4: 100277

English

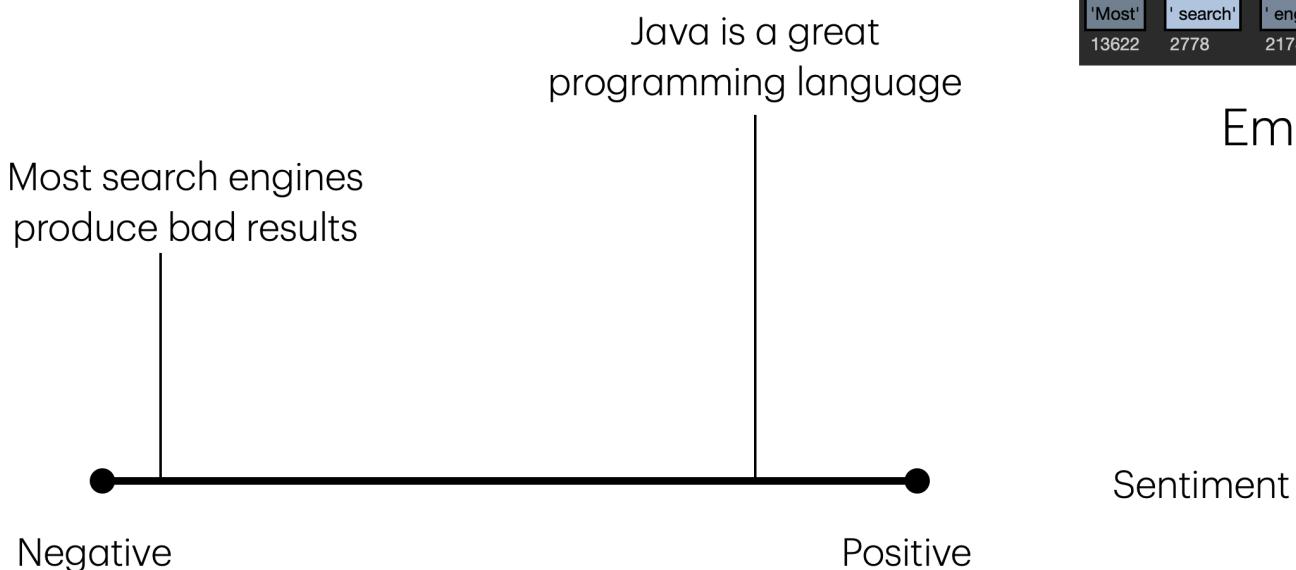


Nederlands



From vectors to concepts

How are vectors or embeddings created



'Java'	'is'	'a'	'great'	'programming'	'language'	''
15391	374	264	2294	15840	4221	198

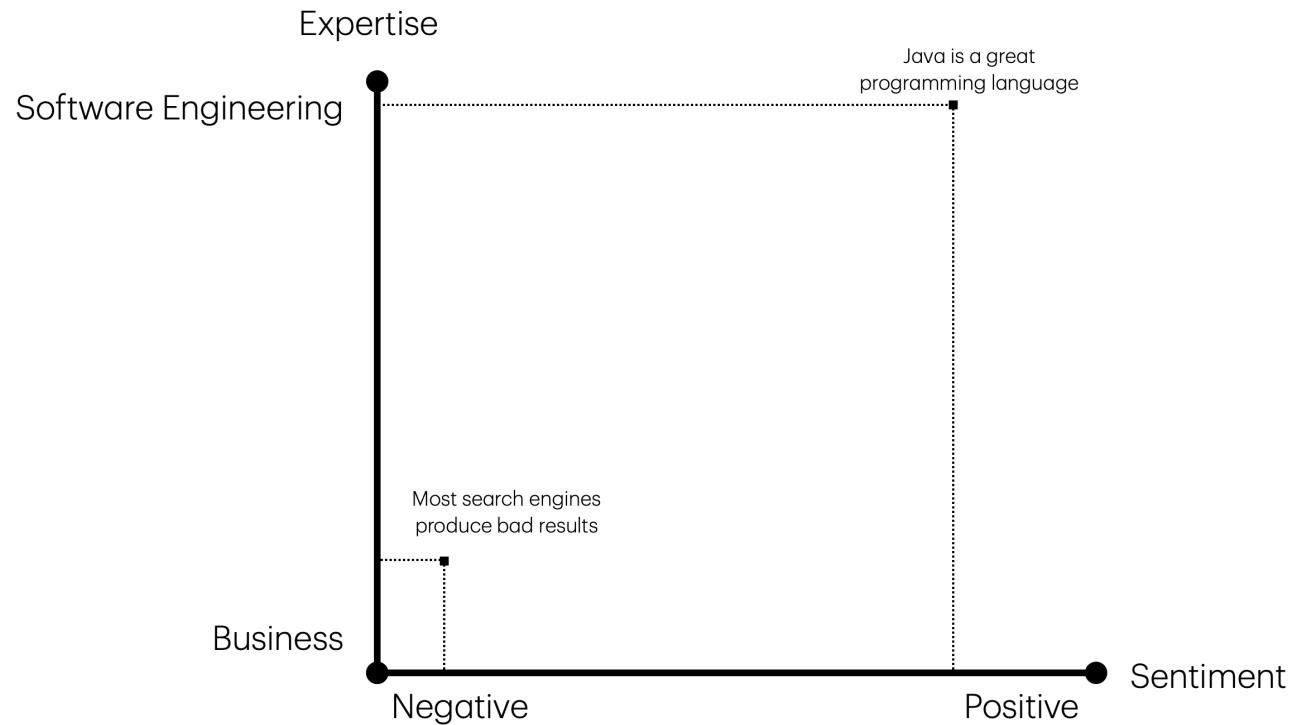
Embedding - [0.85]

'Most'	'search'	'engines'	'produce'	'bad'	'results'	''
13622	2778	21787	8356	3958	3135	198

Embedding - [0.09]

From vectors to concepts

How are vectors or embeddings created



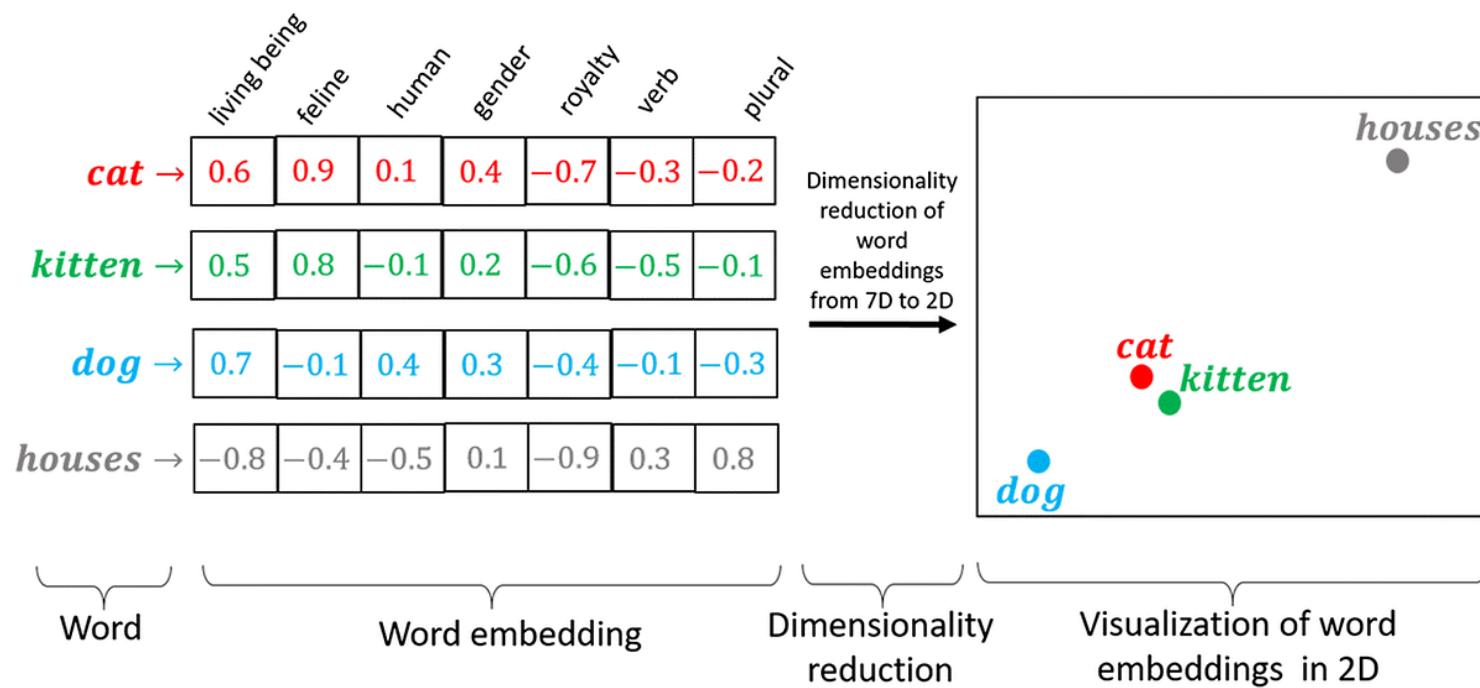
'Java'	'is'	'a'	'great'	'programming'	'language'	'.'
15391	374	264	2294	15840	4221	198

Embedding - [0.85, 1]

'Most'	'search'	'engines'	'produce'	'bad'	'results'	'.'
13622	2778	21787	8356	3958	3135	198

Embedding - [0.09, 1.3]

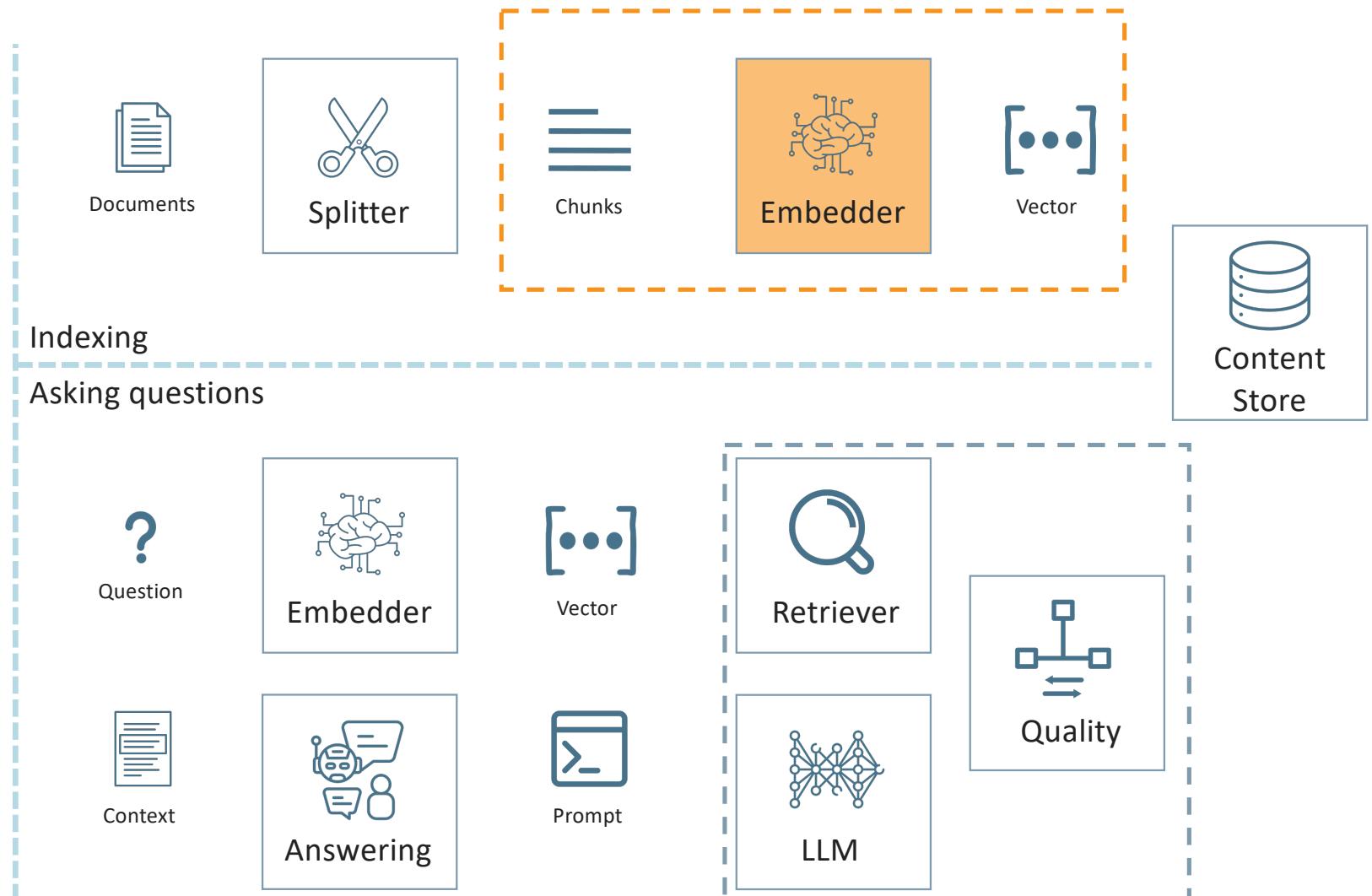
Embeddings

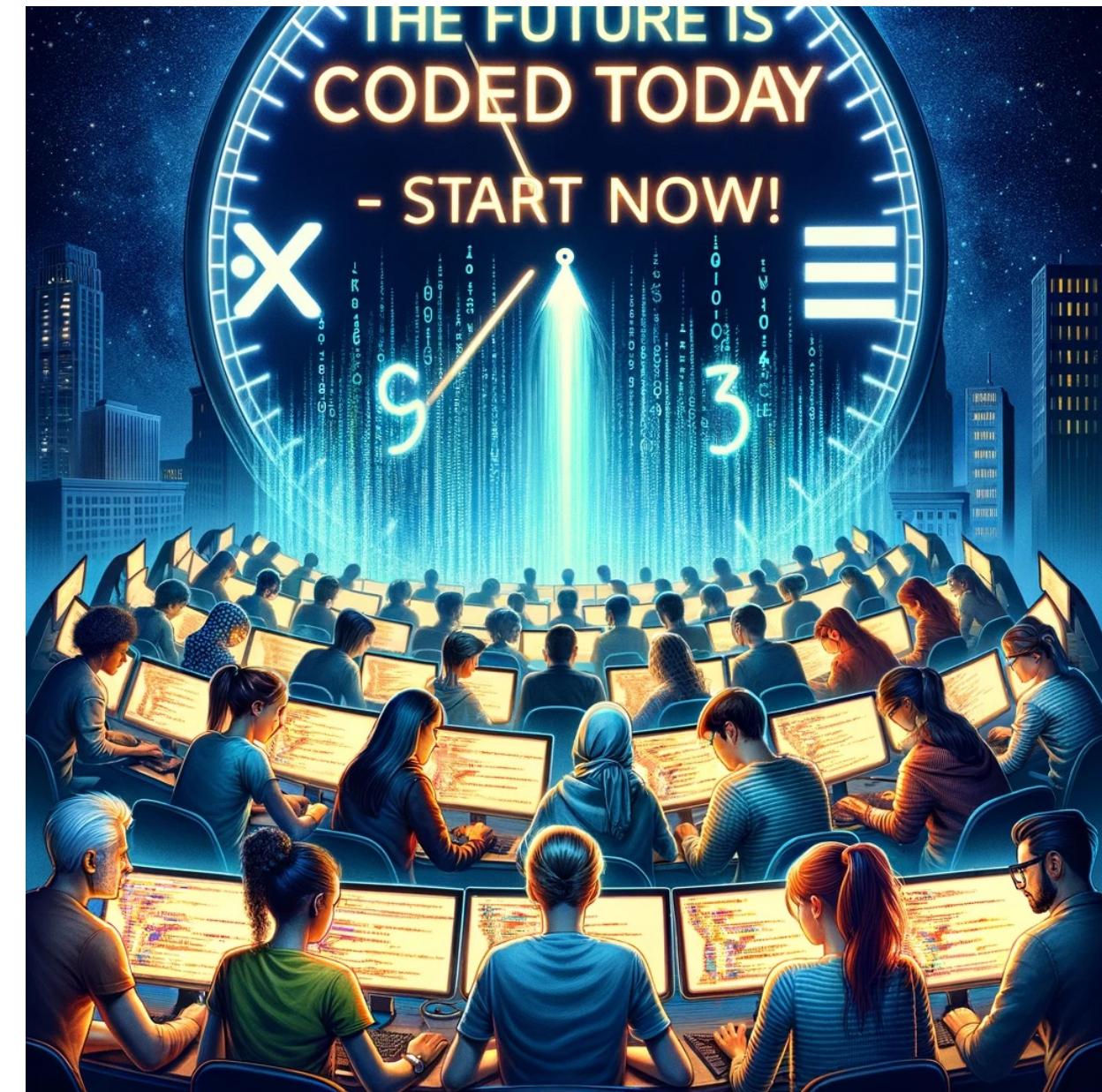




Java: AppStepN...

Python: app_stepN_...





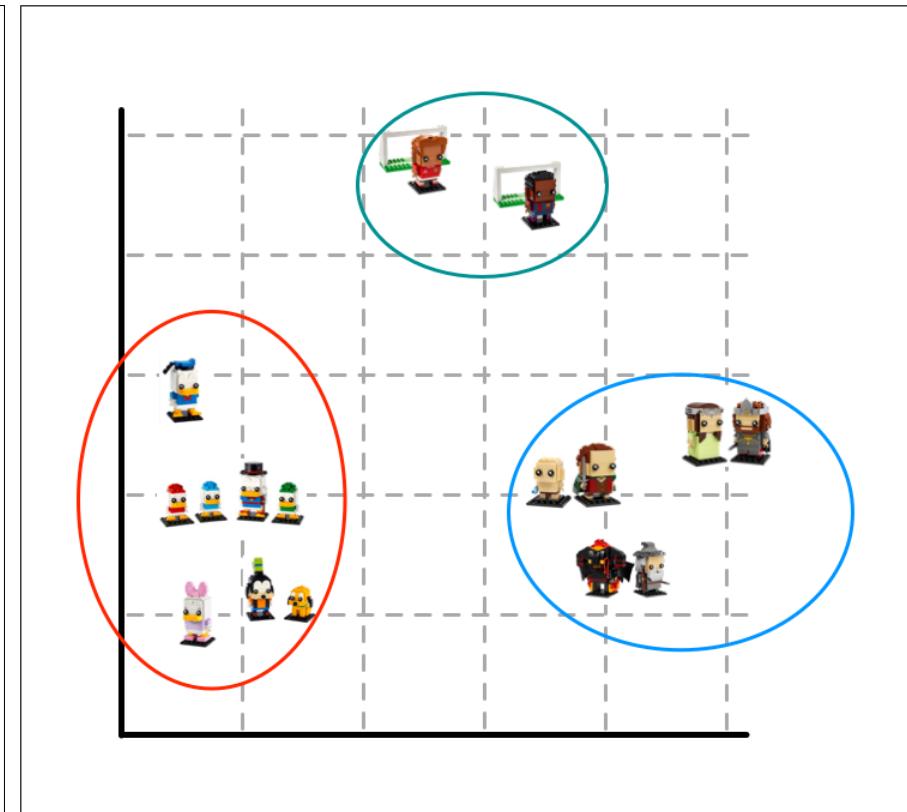
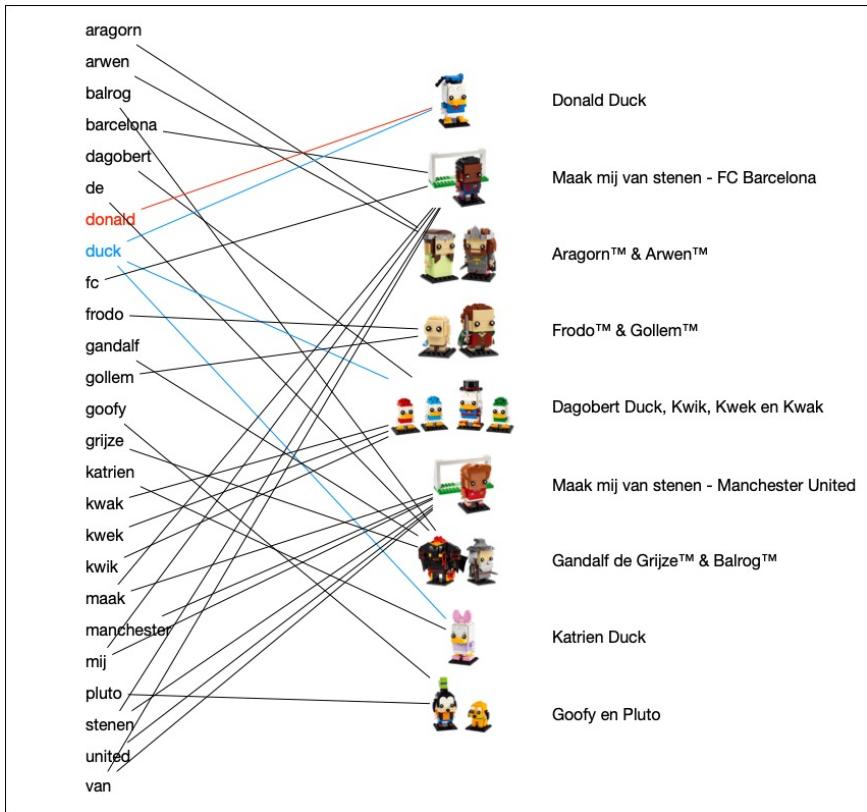
secret_key=

J: AppStep2Embedder

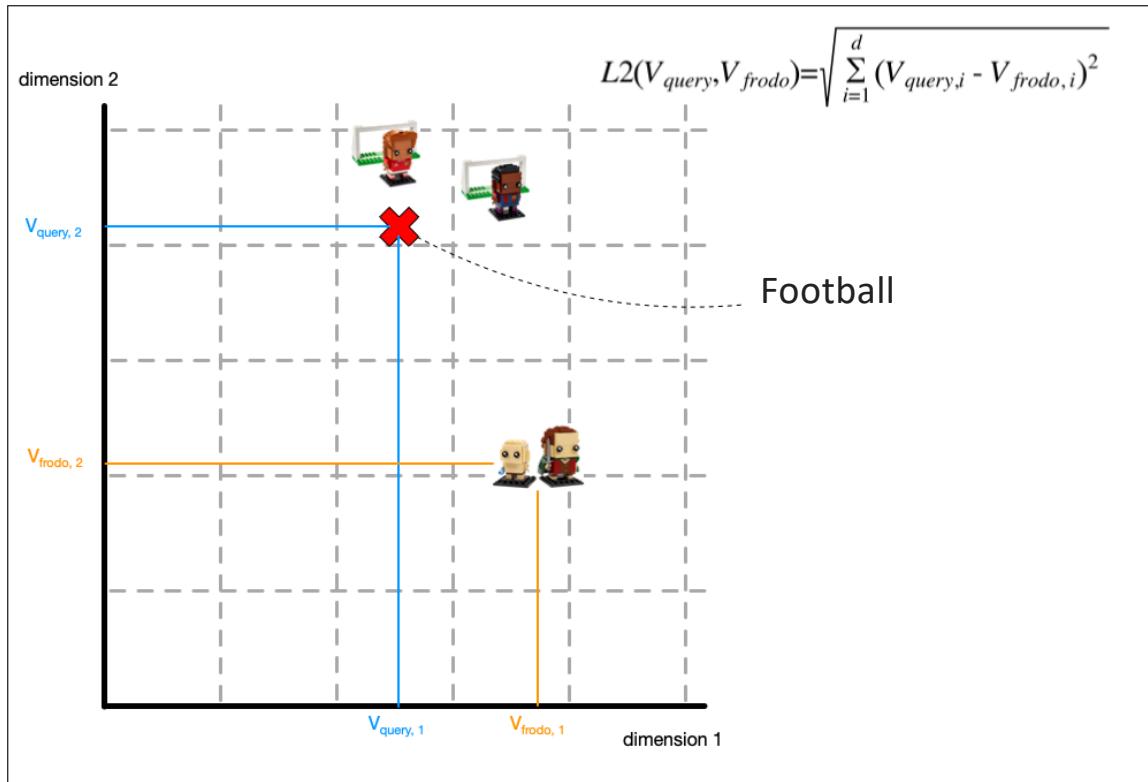
P: app_step2_embedder

Content Stores

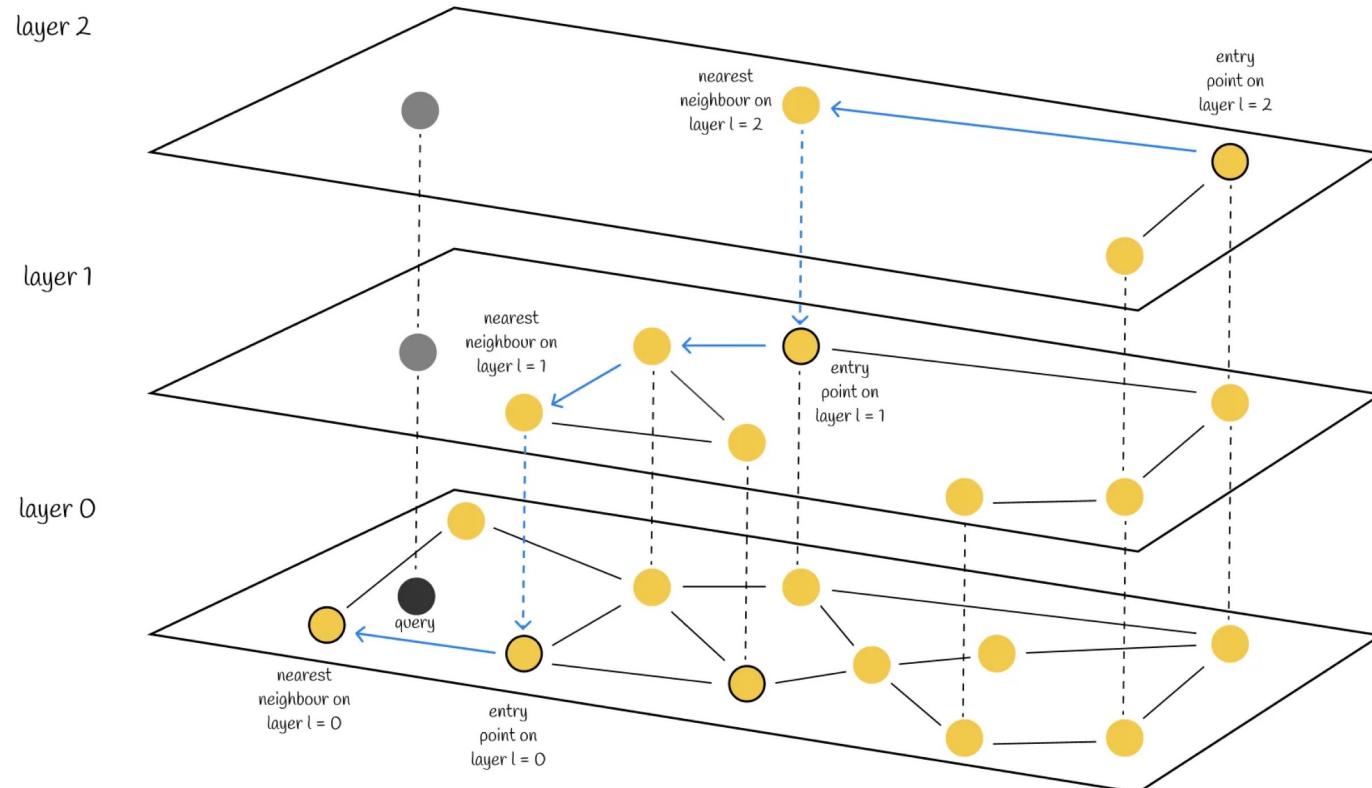
Lexical vs Semantic Search



Comparing vectors



Similarity / HNSW

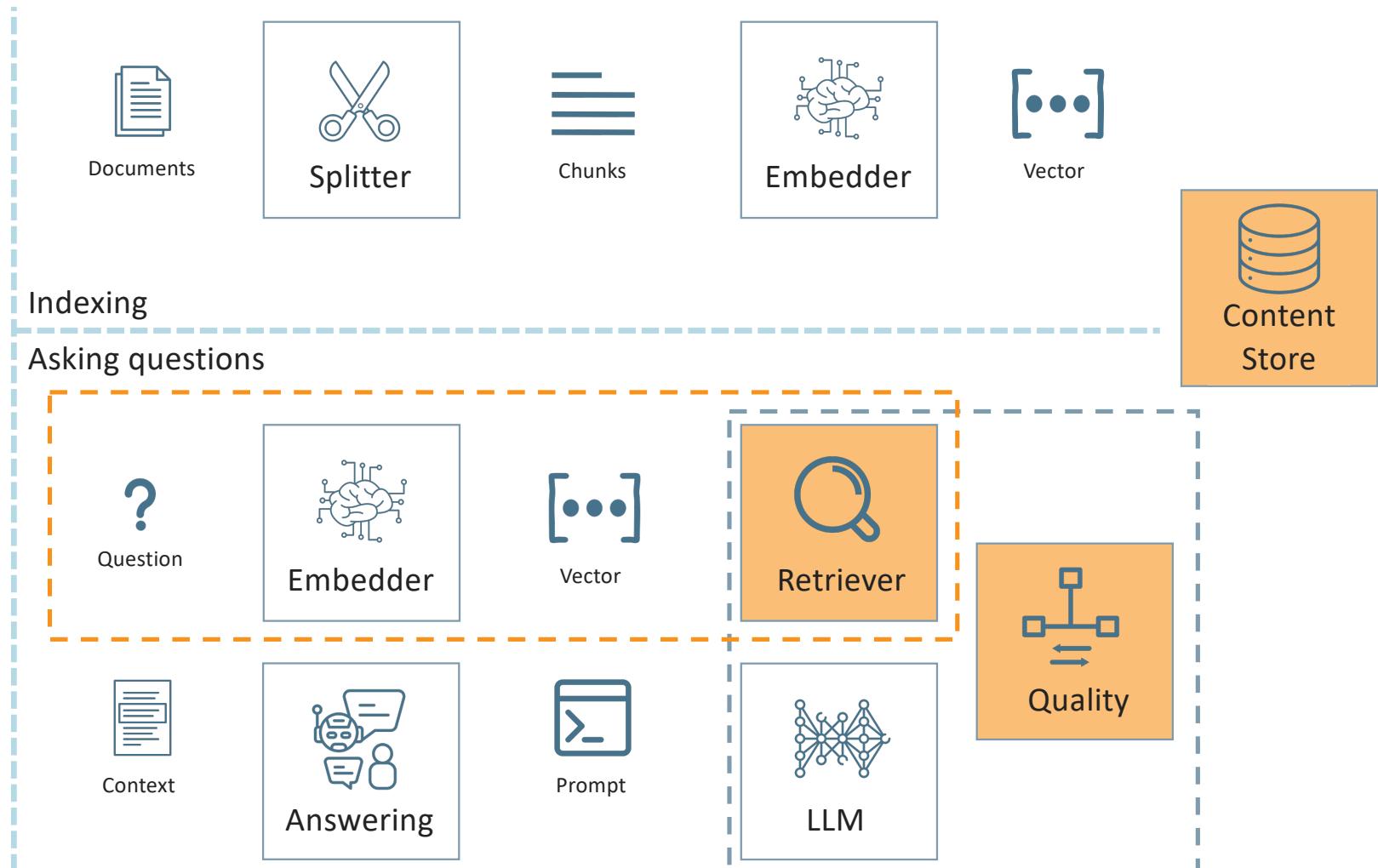


<https://towardsdatascience.com/similarity-search-part-4-hierarchical-navigable-small-world-hnsw-2aad4fe87d37>



Java: AppStepN...

Python: app_stepN_...





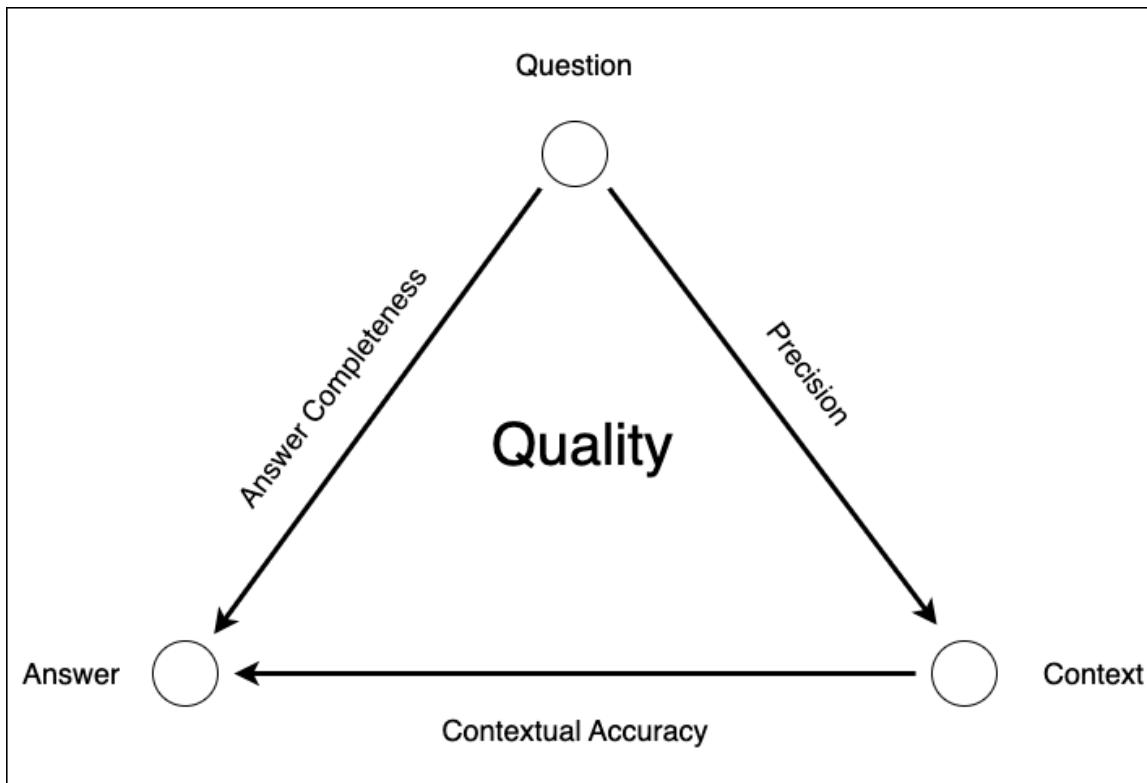
secret_key=

J: AppStep3ContentStore

P: app_step3_content_store

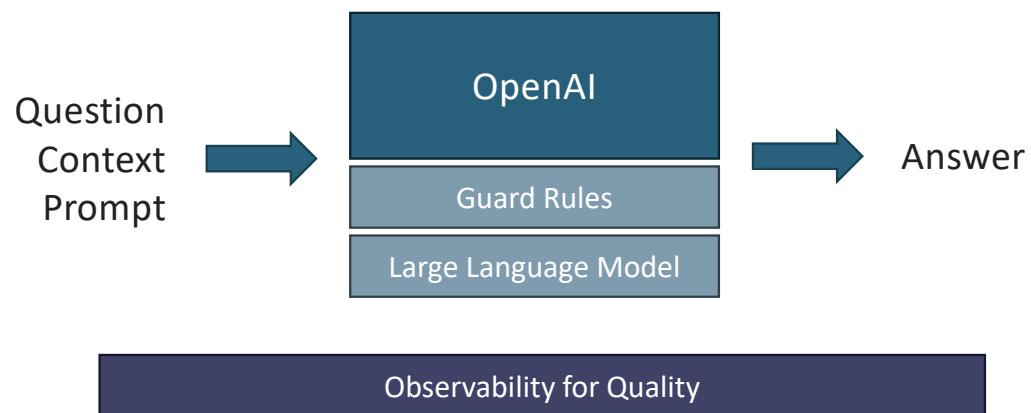
Evaluation

Quality of RAG



Answer Generation

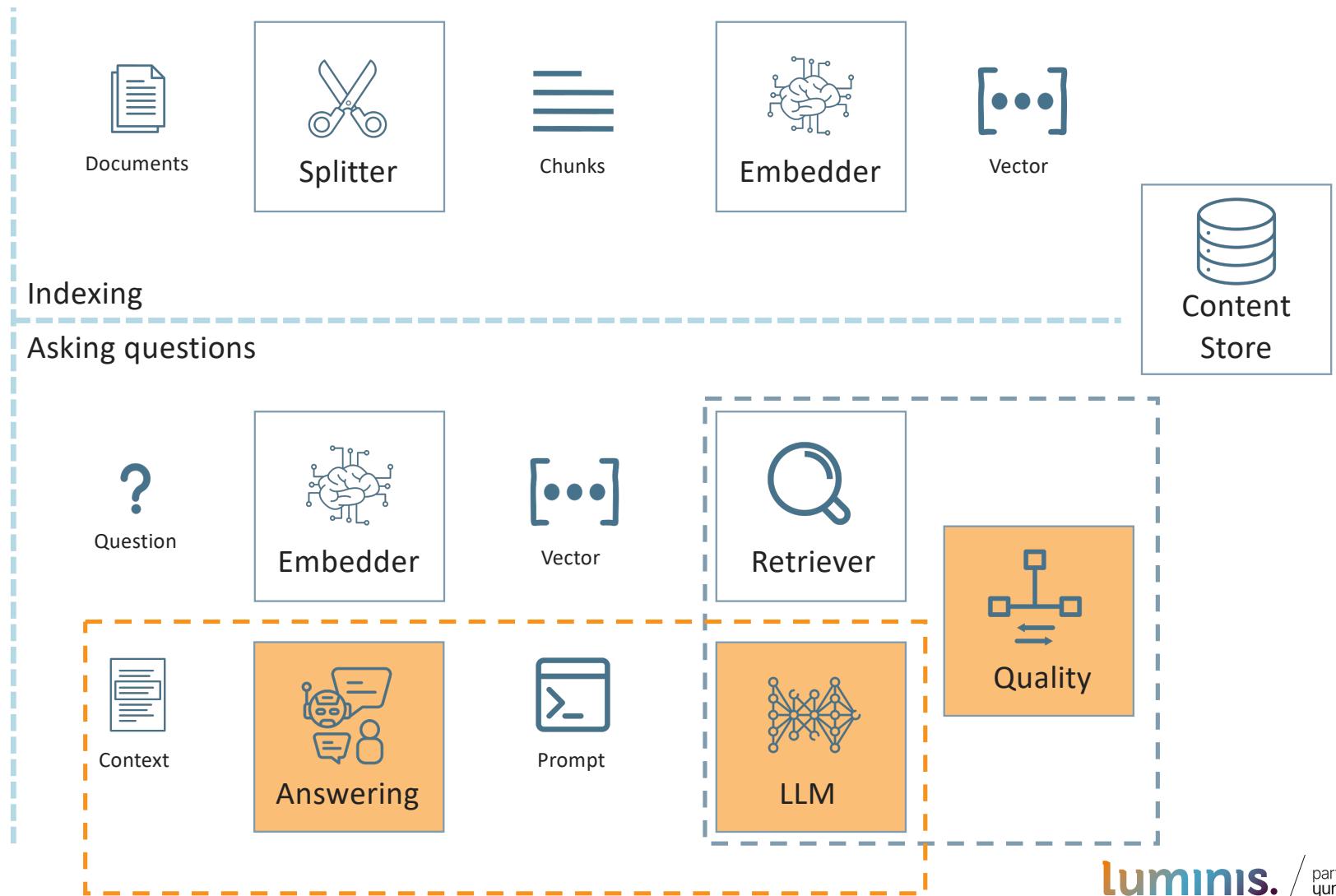
Generating Answers





Java: AppStepN...

Python: app_stepN_...





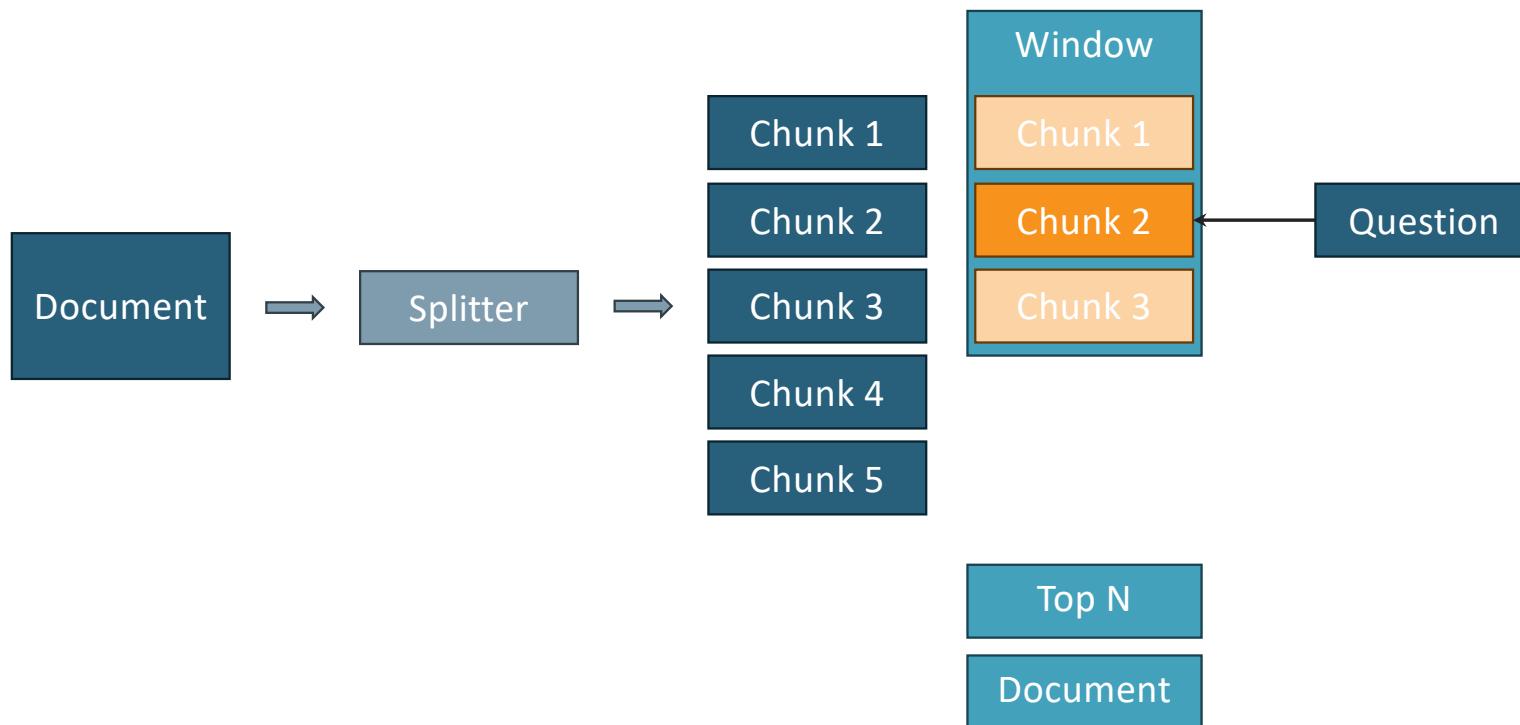
secret_key=

J: AppStep4Generator

P: app_step4_generator

Retrieval Strategy

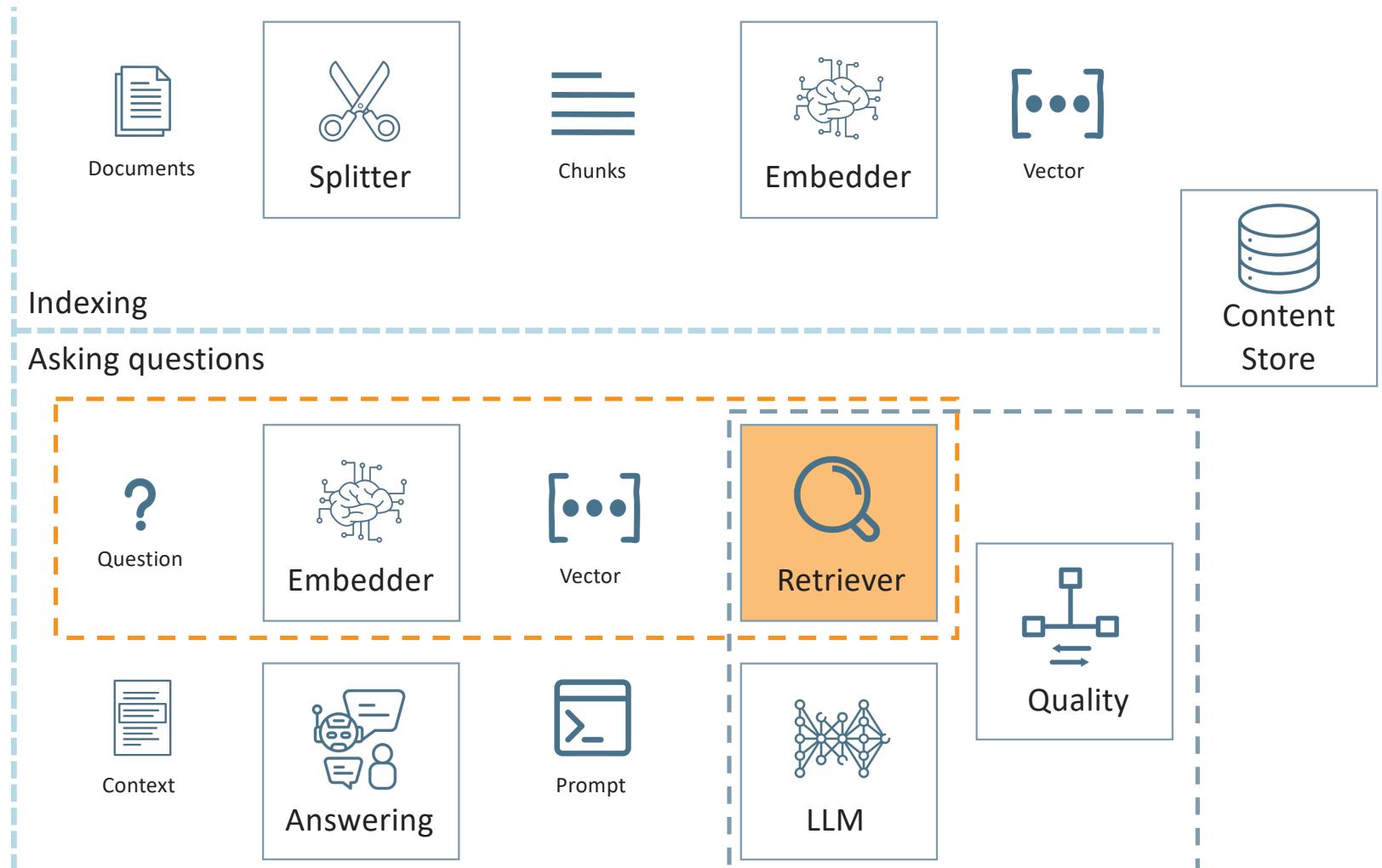
Retrieving the context





Java: AppStepN...

Python: app_stepN_...





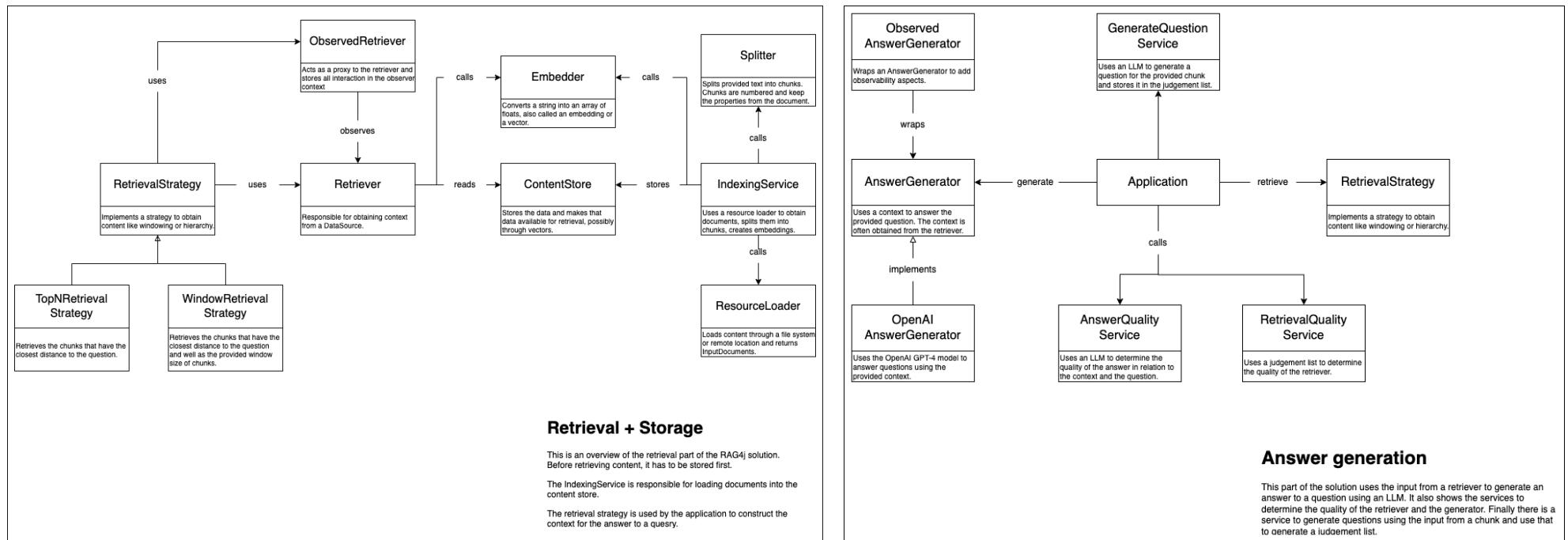
secret_key=

J: AppStep5RetrievalStrategy

P: app_step5_retrieval_strategy

Everything together

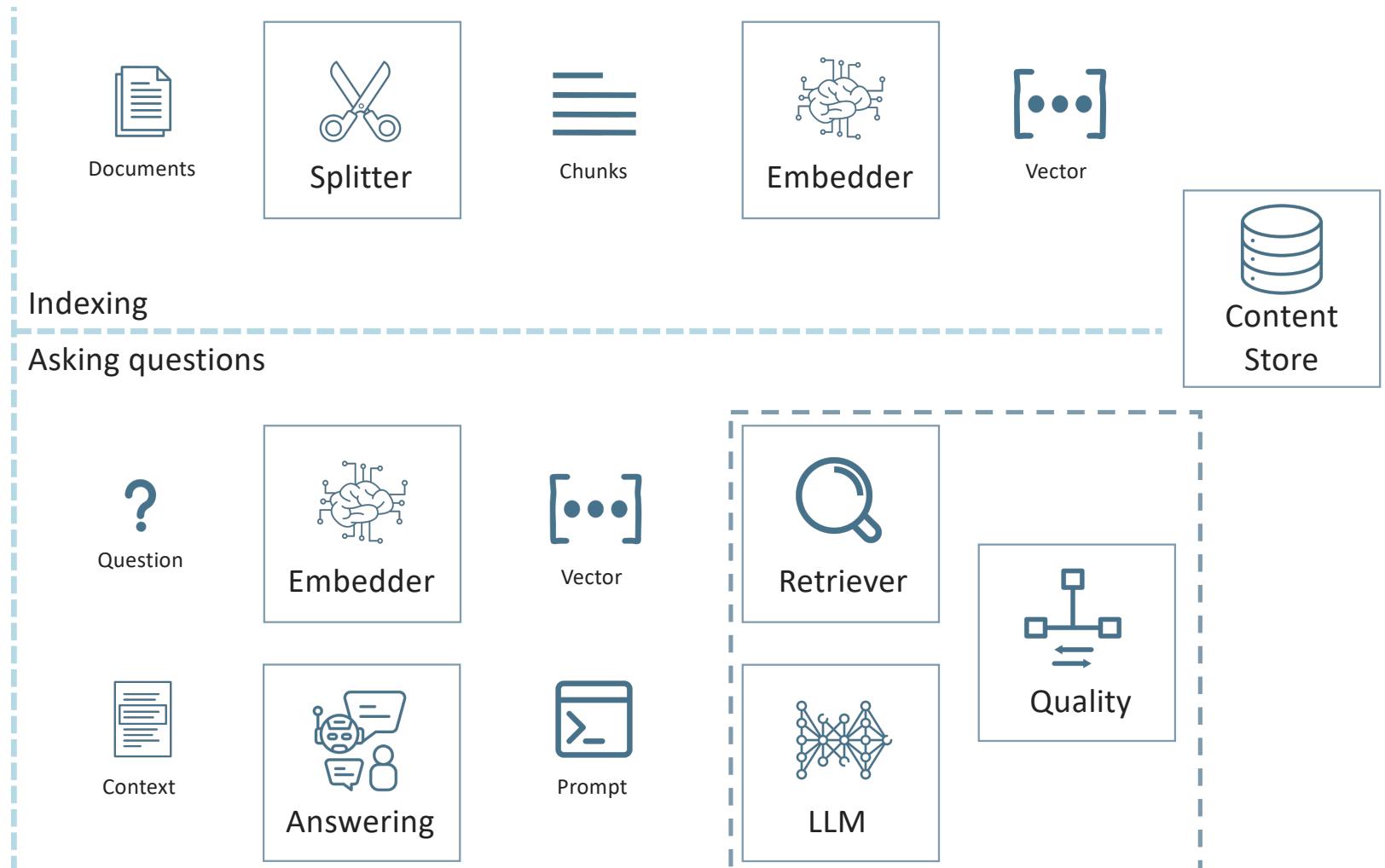
Overall Solution





Java: AppStepN...

Python: app_stepN_...





secret_key=

J: AppStep6AllTogether

P: app_step6_all_together



Thanks!



Daniël Spee



Jettro Coenradie