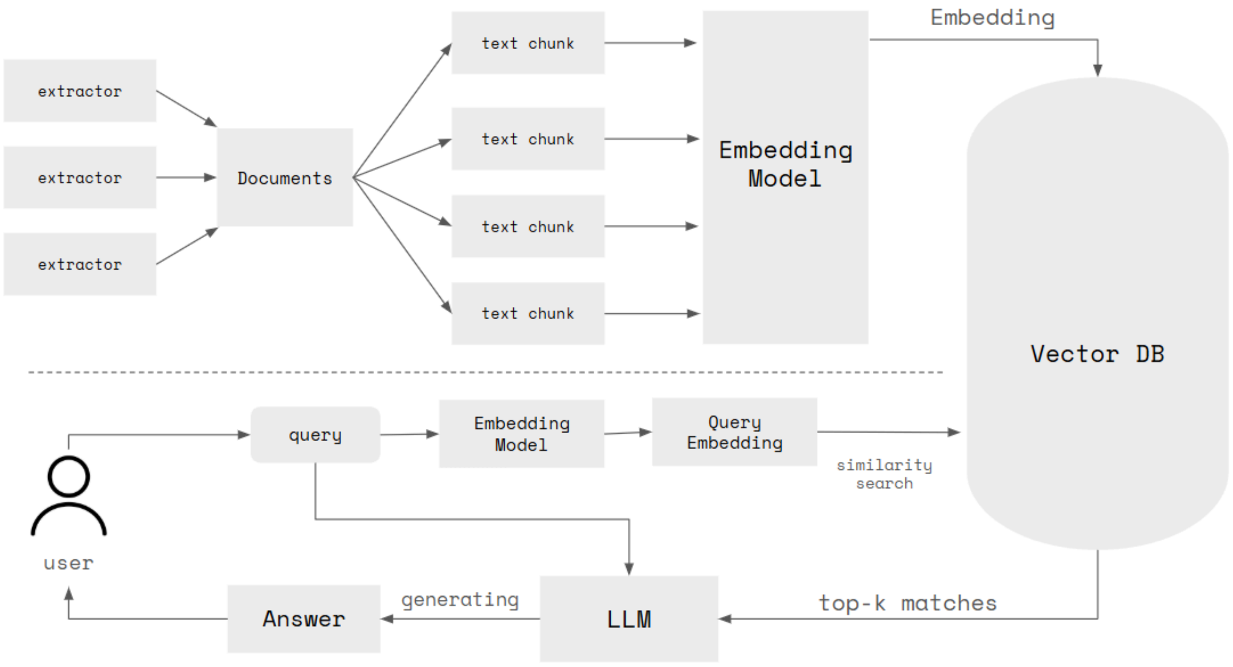
Brief Architecture



1. Since the given dataset is pdf files, and they are unique enough that one content extracting approach would fit to all the pdf files. My solution is to tailor extractor to extract the desired and relevant content for the RAG system.
   1.1 textbook-like pdf which contains a table of content, rich in image, diagram and table and long text content. I use a library called unstructured to extract the content.
   1.2 slide-deck like pdf which contain irrelevant content like emoji, icon, logo. I use a library called unstructured to extract the content.
   1.3 Thai government pdf which often contain broken embedding due to unique Thai characters, e.g. Thai number. I realize unstructured is not capable in this case where it contains broken embedding characters, so I move to high precision OCR to extract Thai texts and use unstructured to extract images, diagrams and tables.
2. The rest of the system is similar to standard RAG. which is chunking the content, indexing via embedding model and storing them in a vector database.
3. The user's query is being embedded via an embedding model and perform similarity search in the vector database to find top-k matches results. Pass it to LLM alongside with the user's query to generate the answer.