



ISO 9001:2015

# TRƯỜNG ĐẠI HỌC TRÀ VINH TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ

BÁO CÁO ĐỒ ÁN TỐT NGHIỆP

## NGHIÊN CỨU RAG VÀ XÂY DỰNG CHATBOT HỖ TRỢ MÔN CƠ SỞ DỮ LIỆU

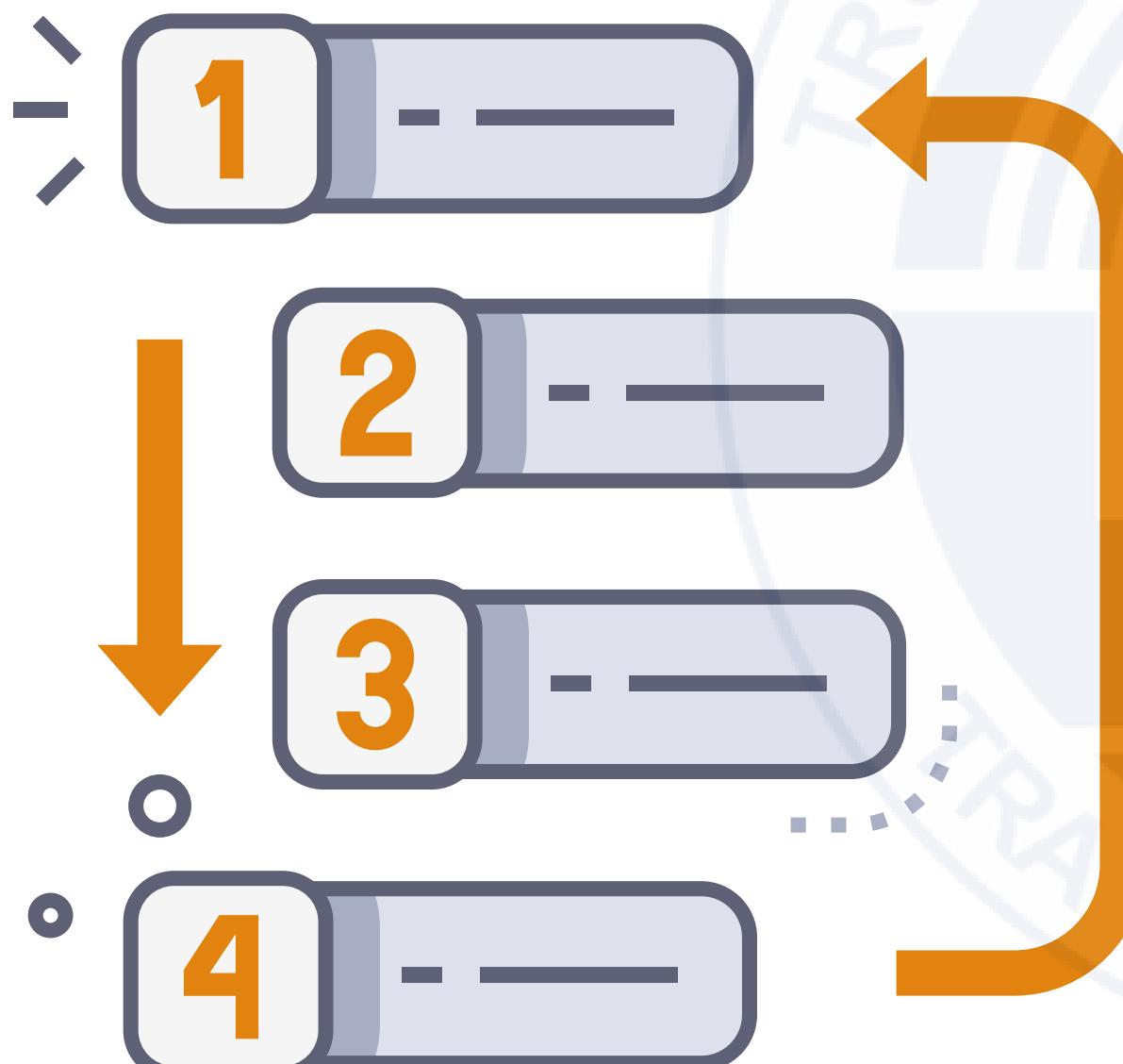
Giảng viên hướng dẫn: TS. NGUYỄN BẢO ÂN

Sinh viên thực hiện: NGUYỄN ĐẠI HOÀNG PHÚC

Vĩnh Long, tháng 7 năm 2025

# NỘI DUNG BÁO CÁO

GỒM 6 PHẦN



- I TỔNG QUAN
- II CƠ SỞ LÝ THUYẾT
- III PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG
- IV TRIỂN KHAI HỆ THỐNG
- V ĐÁNH GIÁ VÀ KẾT QUẢ
- VI KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

# I Tổng quan

## 1.1 Đặt vấn đề:

### •1.1.1 Thách thức với môn Cơ sở dữ liệu (CSDL)

- Khối lượng kiến thức lớn, nhiều khái niệm trừu tượng (chuẩn hóa, giao dịch...).
- Yêu cầu kỹ năng thực hành cao (SQL).
- Sinh viên gặp khó khăn trong việc tự học, tìm kiếm thông tin nhanh và chính xác.

### •1.1.2 Nhu cầu thực tế:

- Cần một công cụ hỗ trợ học tập 24/7
- Giải đáp thắc mắc tức thì dựa trên nguồn tài liệu **chính thống** của môn học.

### •1.1.3 Giải pháp tiềm năng:

- Chatbot ứng dụng trí tuệ nhân tạo, đặc biệt là kỹ thuật Retrieval-Augmented Generation (RAG).



# I Tổng quan

## 1.2 Mục tiêu đề tài:

**Mục tiêu chính:** Nghiên cứu và xây dựng thành công một hệ thống chatbot ứng dụng kĩ thuật RAG để hỗ trợ sinh viên học tập môn CSDL. Cụ thể như sau:

- 1.Nghiên cứu sâu về nguyên lý và kĩ thuật RAG.
- 2.Xây dựng kho tri thức từ tài liệu môn cơ sở dữ liệu.
- 3.Phát triển hệ thống chatbot có khả năng trả lời câu hỏi chính xác, mạch lạc và có trích dẫn nguồn.
- 4.Xây dựng giao diện web thân thiện cho sinh viên và trang quản trị cho admin.
- 5.Đánh giá hiệu quả của hệ thống qua các chỉ số định lượng.



# I Tổng quan

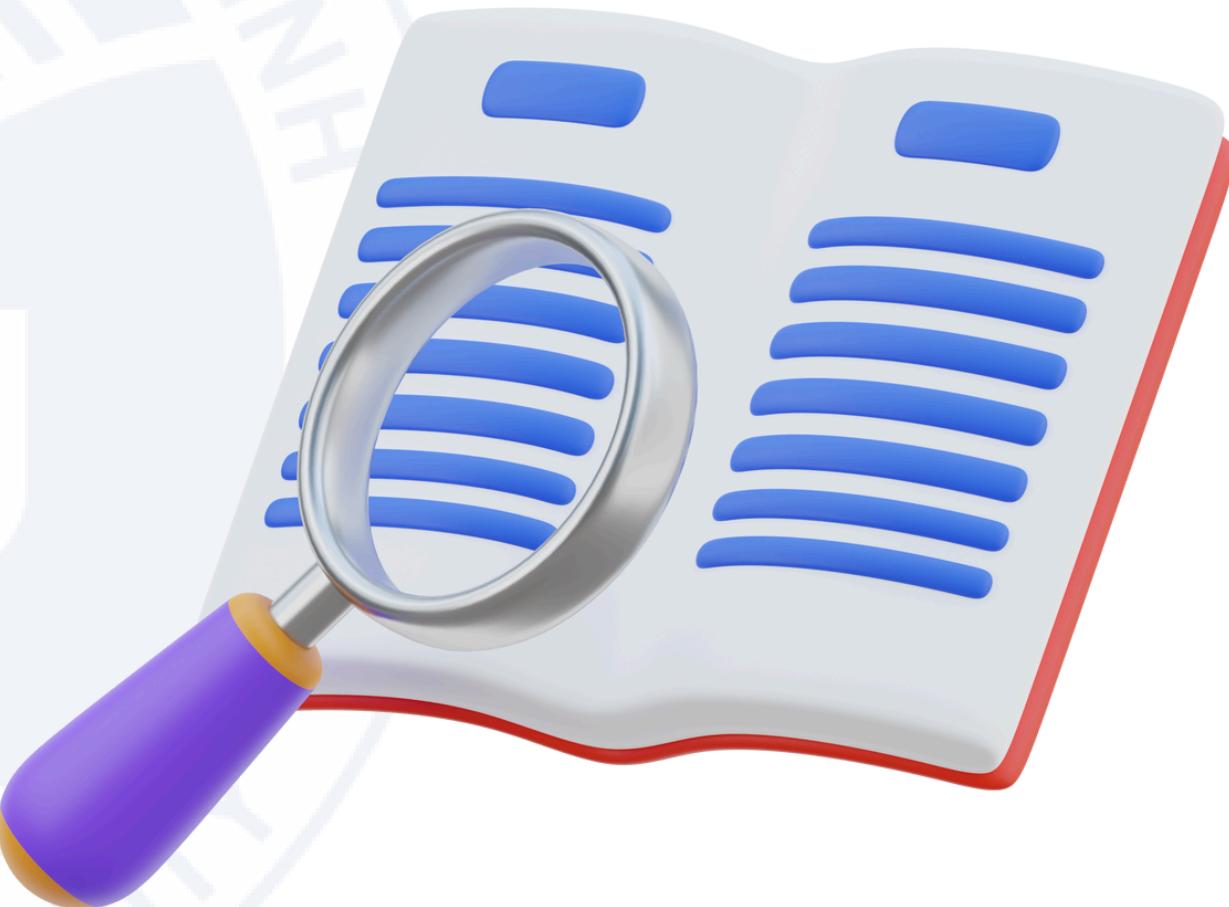
## 1.3 Đối tượng và phạm vi nghiên cứu:

### • 1.3.1 Đối tượng nghiên cứu:

- Các kỹ thuật và công nghệ liên quan đến Retrieval-Augmented Generation (RAG).
- Ứng dụng của RAG trong việc xây dựng hệ thống hỏi đáp thông minh.

### • 1.3.2 Phạm vi nghiên cứu:

- Giới hạn trong nội dung môn Cơ sở dữ liệu đại học cơ bản.
- Trả lời các câu hỏi liên quan đến khái niệm, SQL, thiết kế và truy vấn CSDL,...



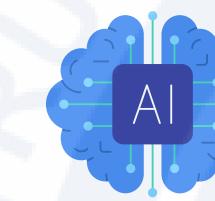
## II Cơ sở lý thuyết

### 2.1 Một số khái niệm liên quan:



#### Chatbot

Chương trình máy tính mô phỏng cuộc trò chuyện của con người qua văn bản hoặc giọng nói.



#### Trí tuệ nhân tạo

Lĩnh vực khoa học máy tính tập trung vào việc tạo ra các hệ thống có khả năng thực hiện các tác vụ đòi hỏi trí thông minh con người.



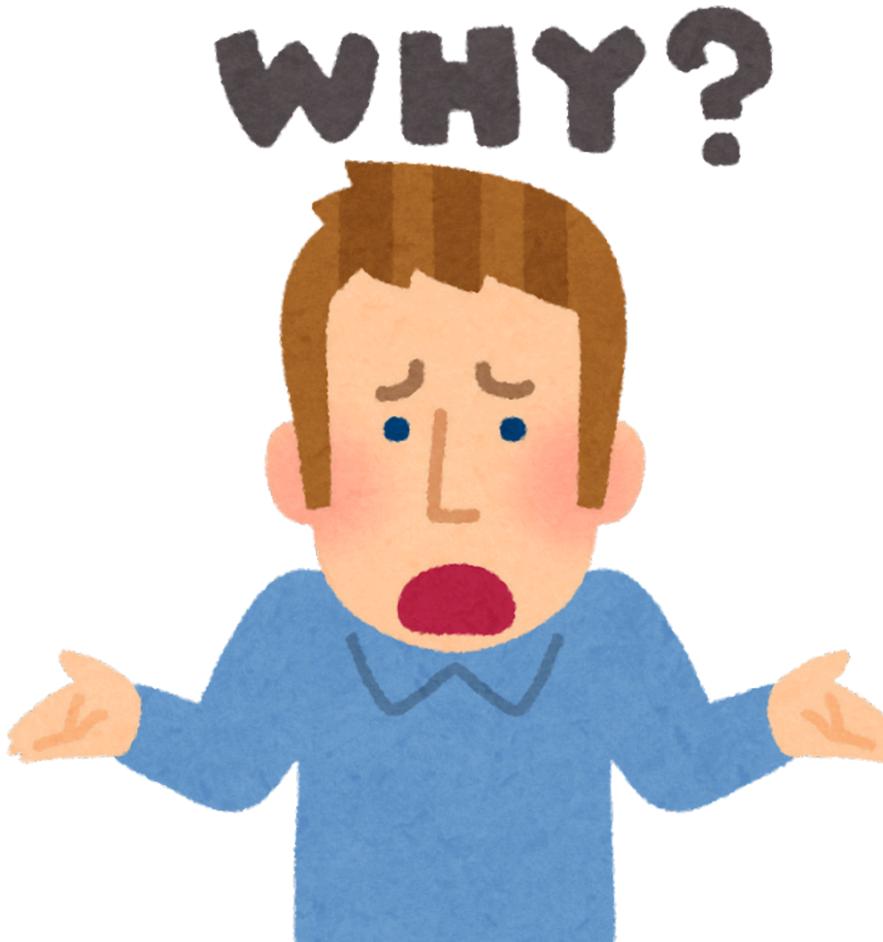
#### Xử lý ngôn ngữ tự nhiên

Một nhánh của trí tuệ nhân tạo cho phép máy tính hiểu, diễn giải và tạo ra ngôn ngữ của con người.

## II Cơ sở lý thuyết

### 2.2 RAG là gì?

- RAG là một kỹ thuật kết hợp giữa "*Truy xuất thông tin*" (Retrieval) và "*Sinh văn bản*" (Generation).
- Giúp các *mô hình ngôn ngữ lớn* (LLM) trả lời câu hỏi dựa trên *tập dữ liệu cụ thể*, thay vì chỉ dựa vào dữ liệu mà chúng được huấn luyện.



### 2.3 Tại sao cần RAG?

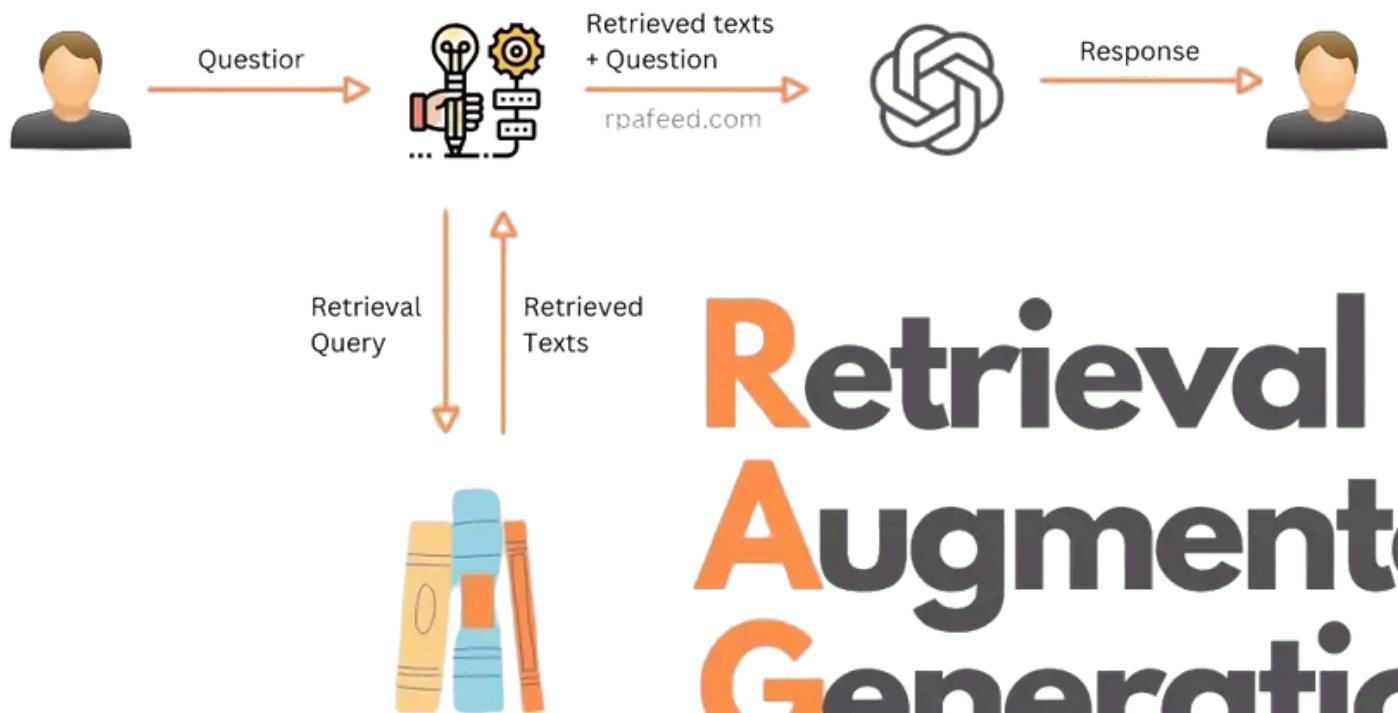
- Khắc phục *ảo giác* (hallucination) của LLM.
- Dễ dàng cập nhật kiến thức: LLM có kiến thức bị giới hạn bởi *thời điểm huấn luyện*; RAG có thể giúp cập nhật thông tin mới mà không cần huấn luyện lại mô hình.
- Tăng tính tin cậy và minh bạch.



## II Cơ sở lý thuyết

### 2.4 Các bước trong quy trình RAG

- 1. Query (Câu hỏi):** Người dùng *gửi câu hỏi*.
- 2. Retrieve (Truy xuất):** Hệ thống *truy xuất* trong kho tri thức Vector (Vector DB) các *đoạn tài liệu* (chunks) có *ngữ nghĩa tương đồng* nhất với câu hỏi.
- 3. Augment (Tăng cường):** Các *chunks* liên quan được *ghép với câu hỏi ban đầu* để tạo thành một *prompt* mở rộng.
- 4. Generate (Sinh câu trả lời):** LLM *nhận prompt* và *sinh ra câu trả lời* dựa trên *ngữ cảnh* đã được bổ sung.



**Retrieval  
Augmented  
Generation.**

## II Cơ sở lý thuyết

### 2.5 Các kĩ thuật và công nghệ

#### 2.5.1 Xử lý ngôn ngữ và tiền xử lý văn bản

##### - Làm sạch & chuẩn hóa

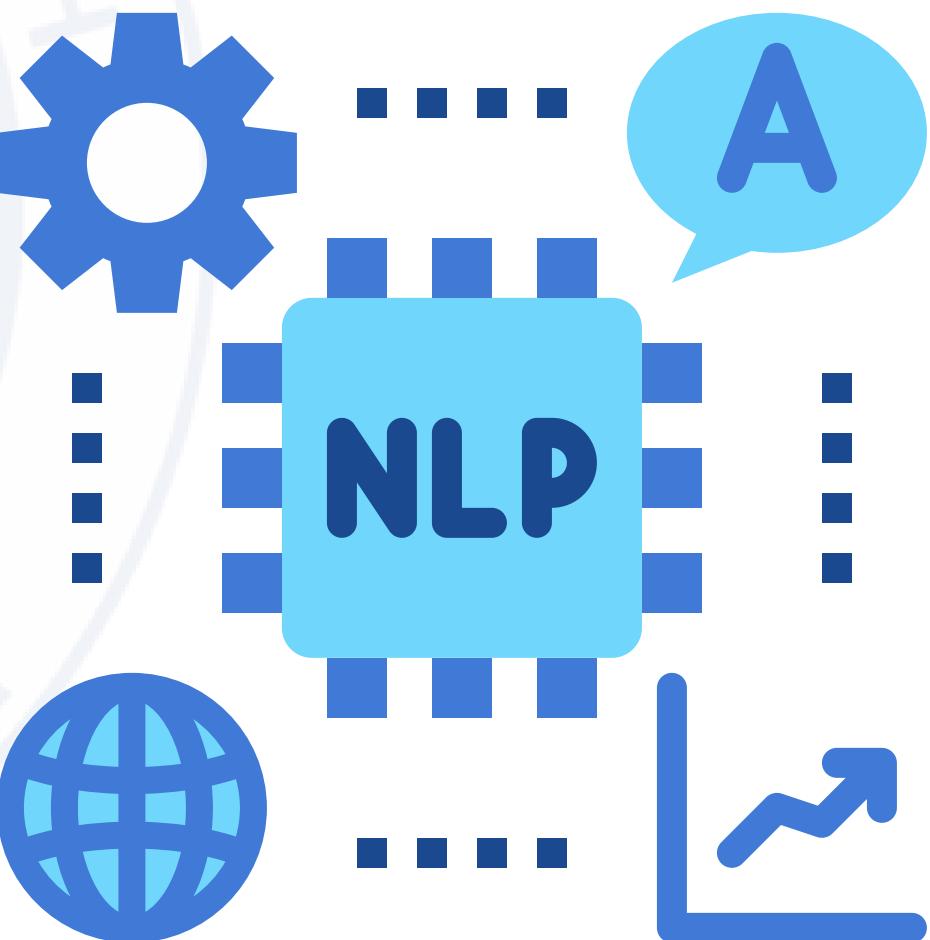
- Xóa các kí phan có HTML, ký tự đặc biệt, khoảng trắng thừa
- Chuẩn hóa Unicode, chuyển chữ thường, thống nhất thuât ngữ

##### - Phân đoạn (chunking)

- Chia văn bản lớn thành đoạn nhỏ.
- Dựa vào cấu trúc logic (tiêu đề, đoạn văn) hoặc chia theo độ dài cố định

##### - Chuẩn bị embedding & indexing

- Mã hóa mỗi đoạn (chunk) thành các vector.
- Lưu vào Vector DB để phục vụ retrieval



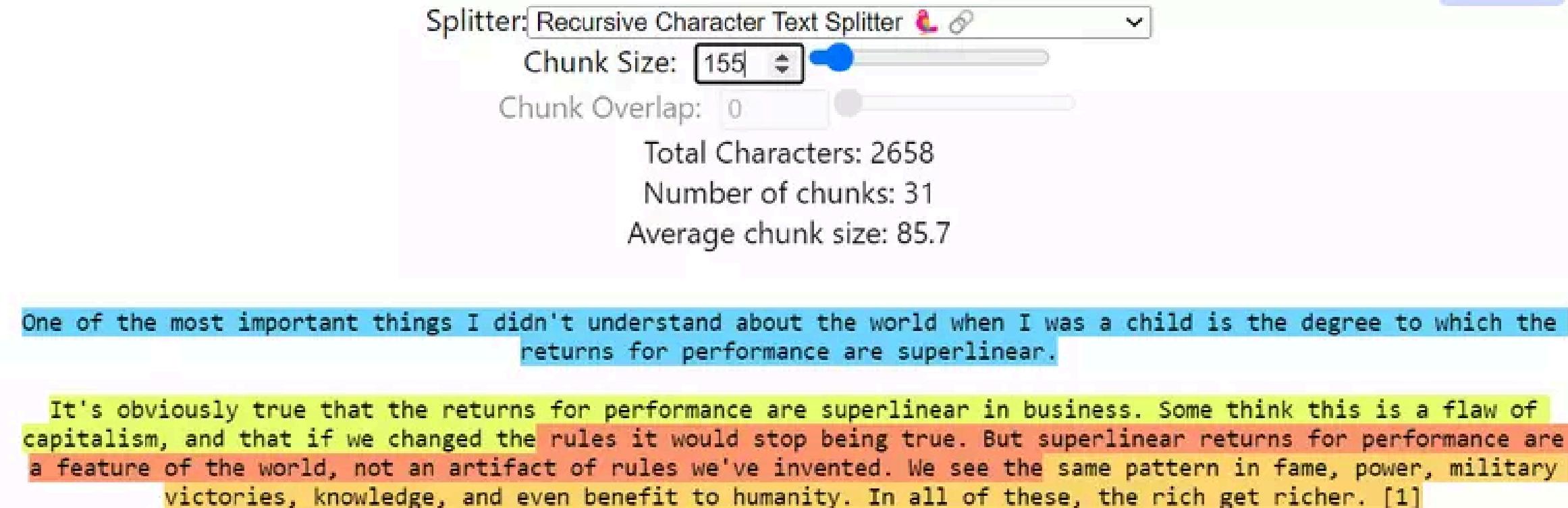
## II Cơ sở lý thuyết

### 2.5 Các kĩ thuật và công nghệ

#### 2.5.1 Xử lý ngôn ngữ và tiền xử lý

Về phương pháp chunking: Đề tài sử dụng kỹ thuật **Recursive Chunking**

Phương pháp này sẽ dựa vào những separators được thiết lập sẵn như: ["\n\n", "\n", " ", ""]. Và từ đó chúng sẽ cắt theo thứ tự ưu tiên dựa trên separators sao cho chunk được lấy ra vừa là **dài nhất có thể vừa giữ được tính toàn vẹn của nội dung.**



## II Cơ sở lý thuyết

### 2.5 Các kĩ thuật và công nghệ

#### 2.5.2 Kỹ thuật nhúng văn bản (Embedding)

- **Mục tiêu:** Biến đoạn văn thành vector số để máy tính có thể so sánh ngữ nghĩa.
- **Embedding** là kỹ thuật ánh xạ văn bản vào không gian nhiều chiều sao cho các đoạn giống nghĩa sẽ có vector gần nhau.
- **Quy trình thực hiện:**
  - Đưa đoạn văn (sau chunking) vào mô hình embedding để tạo ra các vector embedding.
  - Lưu vào vector database (Vector Store) để phục vụ tìm kiếm.



## II Cơ sở lý thuyết

### 2.5 Các kĩ thuật và công nghệ

#### 2.5.3 Cơ sở dữ liệu vector và Truy xuất ngữ nghĩa

##### - Cơ sở dữ liệu vector:

- Lưu trữ và đánh chỉ mục hàng loạt vector embedding hiệu quả.
- Hỗ trợ truy vấn nhanh trên không gian nhiều chiều.
- Ví dụ: FAISS, Pinecone, [Qdrant](#),...

##### - Lý do chọn Qdrant:

- Hỗ trợ metadata & filtering: đính kèm thông tin (tên tài liệu, trang...) để lọc khi truy vấn.
- Dễ triển khai (Docker) và tích hợp API đa ngôn ngữ.
- Khả năng mở rộng khi dữ liệu tăng.



## II Cơ sở lý thuyết

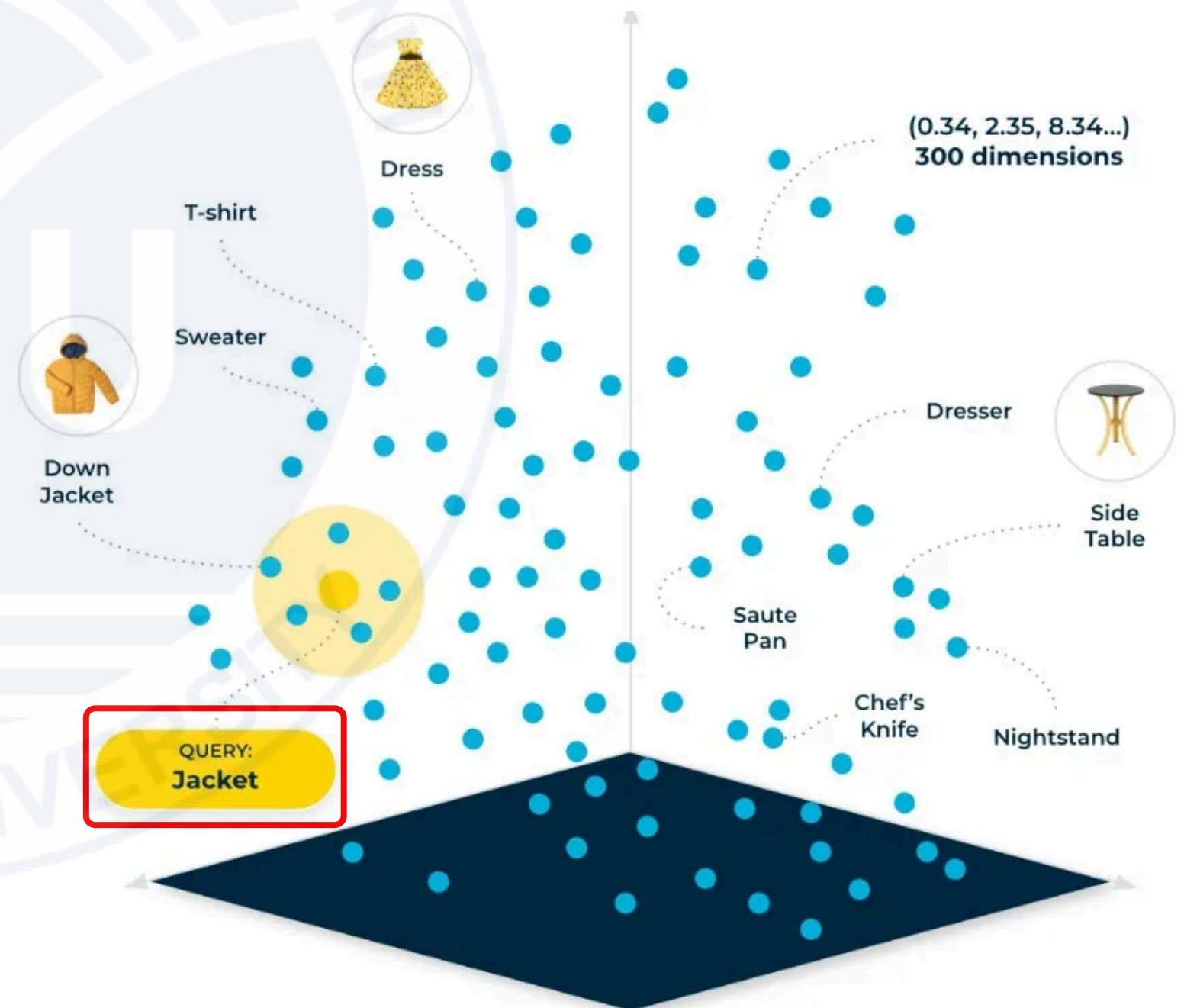
### 2.5 Các kĩ thuật và công nghệ

#### 2.5.3 Cơ sở dữ liệu vector và Truy xuất ngữ nghĩa

**Truy xuất ngữ nghĩa:**

- **Mục tiêu:** Tìm các đoạn văn liên quan nhất đến câu hỏi dựa trên ý nghĩa.
- **Nguyên lý Semantic Search:**

- Biểu diễn câu hỏi và tài liệu dưới dạng vector embedding.
- Tính độ gần nhau trong không gian ngữ nghĩa (dùng cosine similarity).
- Kết quả trả về là những đoạn có nội dung tương tự, dù từ ngữ có thể khác nhau.



## II Cơ sở lý thuyết

### 2.5 Các kĩ thuật và công nghệ

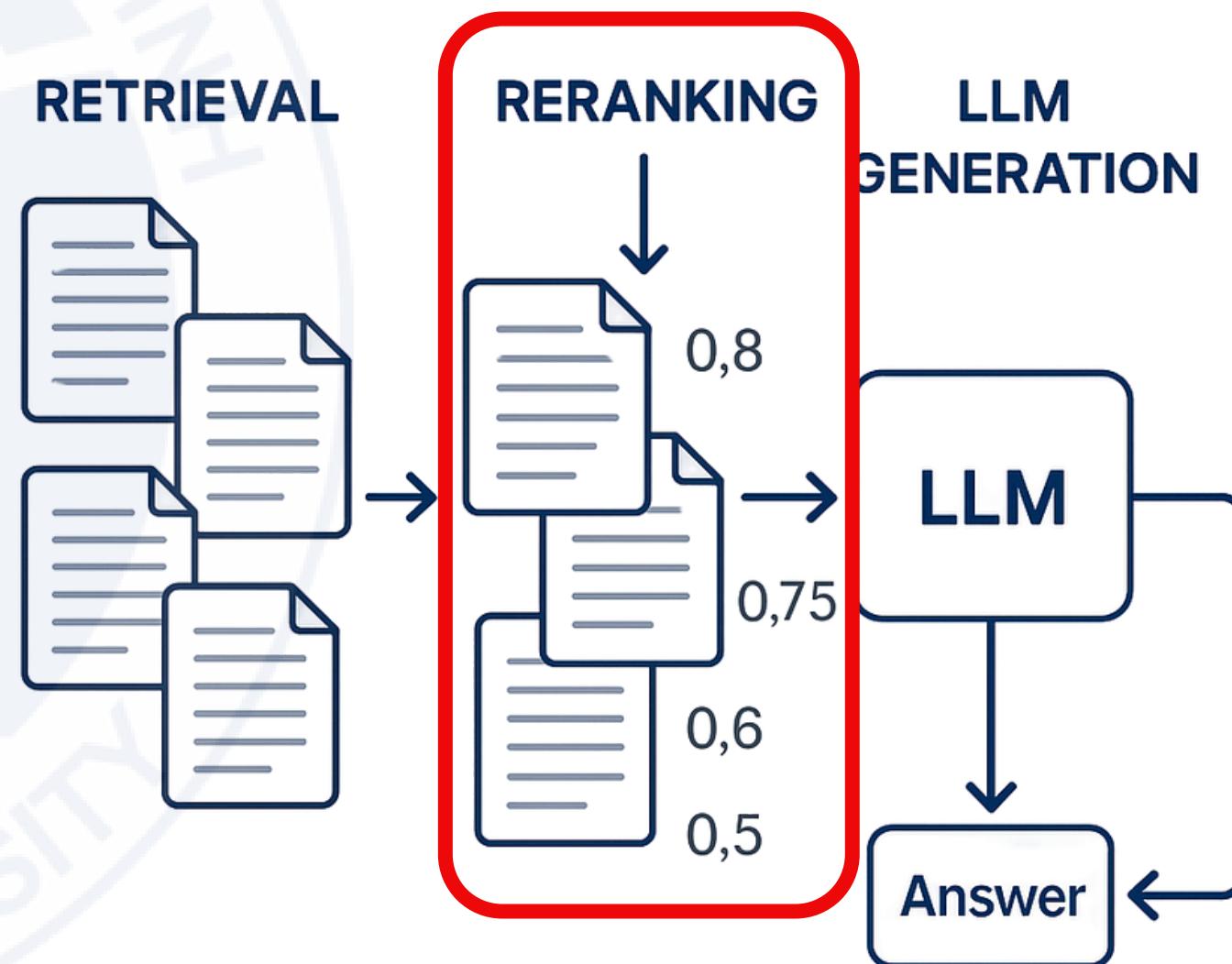
#### 2.5.4 Reranking và mô hình ngôn ngữ lớn

##### Định nghĩa:

- Reranking là bước lọc và sắp xếp lại các đoạn văn, đưa đoạn phù hợp nhất lên đầu, loại bỏ đoạn ít liên quan hoặc trùng lặp.
- Giúp ngữ cảnh đưa vào LLM tinh gọn, sát chủ đề.

##### Quy trình

- Lặp qua từng chunk:
  - Ghép câu hỏi + chunk thành một input duy nhất.
  - Model reranking sẽ tính score relevance (điểm liên quan) mới.
- Sắp xếp các chunk theo score này, chọn ra top-k (ví dụ k=10).



### 2.5 Các kĩ thuật và công nghệ

#### 2.5.5 Reranking và mô hình ngôn ngữ lớn

**Định nghĩa:** LLM là mô hình học sâu với **hàng tỷ tham số**, được **huấn luyện** trên lượng **dữ liệu khổng lồ**, có khả năng sinh văn bản mạch lạc.

**Hạn chế:** Kiến thức bị **đóng băng** ở thời điểm huấn luyện, có thể tạo thông tin sai (**hallucination**) nếu gặp câu hỏi ngoài kiến thức.

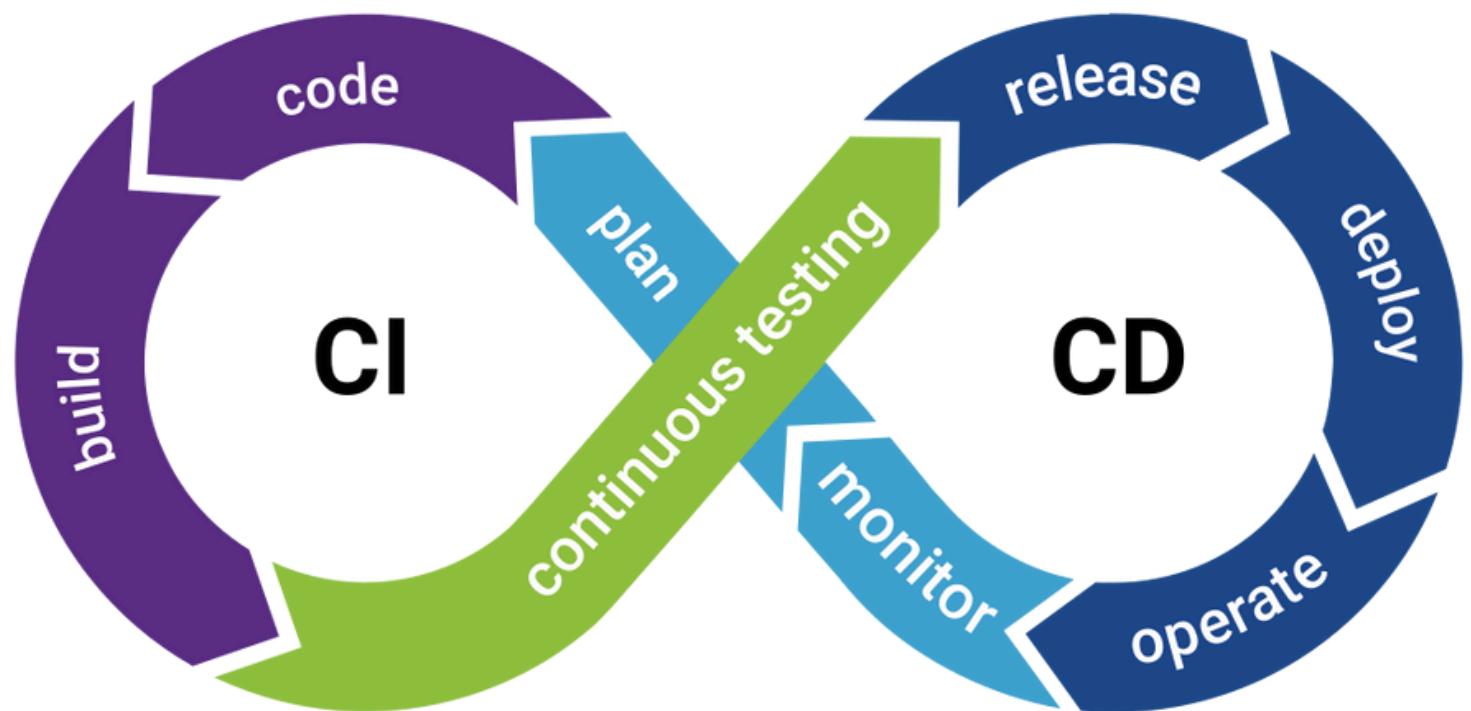
**Ứng dụng trong đề tài:** Sử dụng API Google Gemini – mô hình mạnh, đa phương thức, hỗ trợ phân tích và suy luận tốt.



## II Cơ sở lý thuyết

### 2.5 Các kĩ thuật và công nghệ

#### 2.5.6 Quy trình CI/CD:



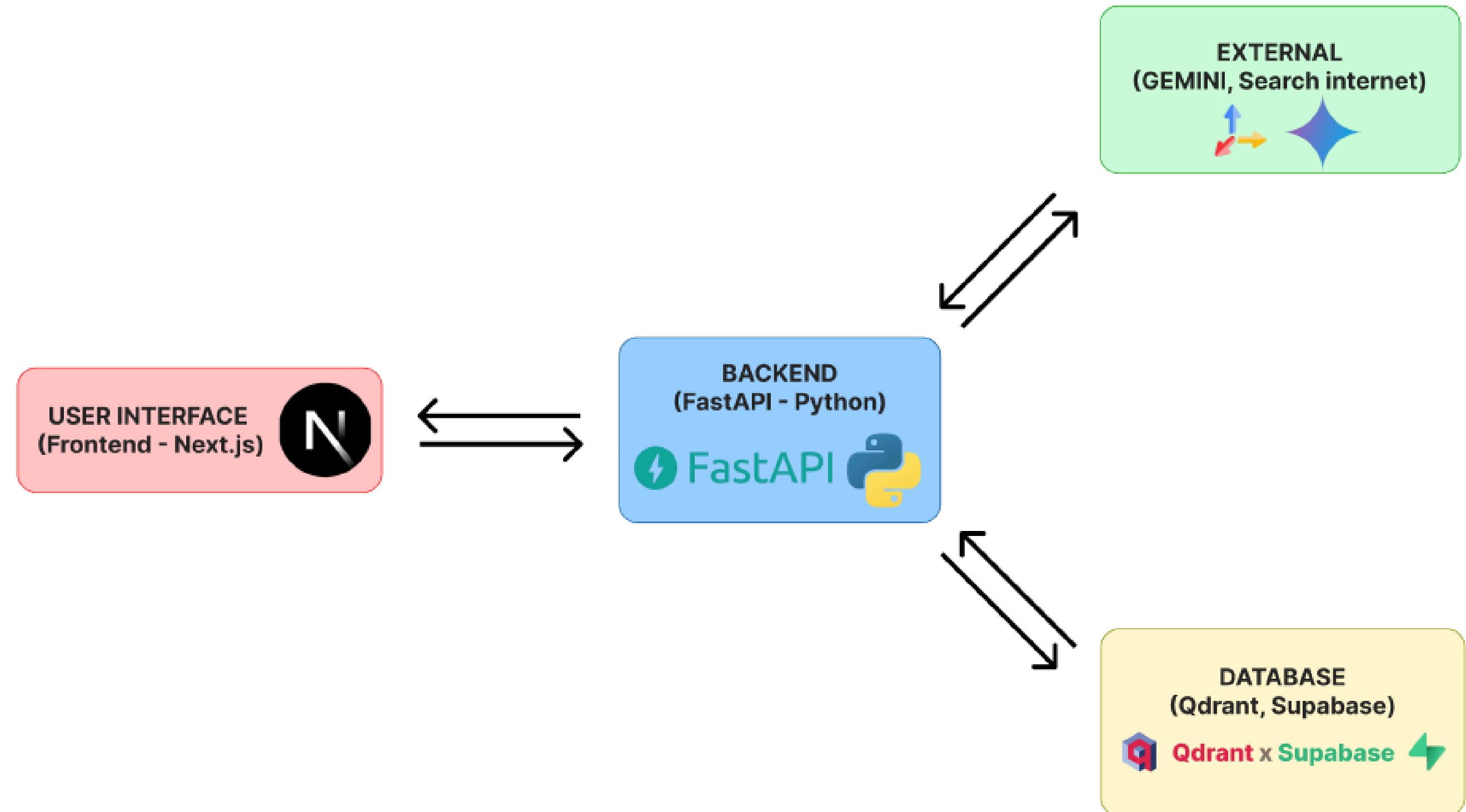
**CI – Continuous Integration:** Là quá trình tích hợp mã **thường xuyên** vào nhánh chung. Mỗi lần tích hợp sẽ được **build & test tự động** để phát hiện lỗi sớm, giúp mã luôn ổn định.

**CD – Continuous Delivery / Deployment**

- **Continuous Delivery:** Sau CI, mã được **tự động đóng gói** và đưa vào môi trường thử nghiệm/staging, sẵn sàng để triển khai đến production.
- **Continuous Deployment:** Nếu tất cả bước kiểm tra đều qua, phiên bản mới được **tự động đưa vào production** mà không cần thao tác thủ công .

# III Phân tích và thiết kế hệ thống

## 3.1 Kiến trúc tổng thể của hệ thống:



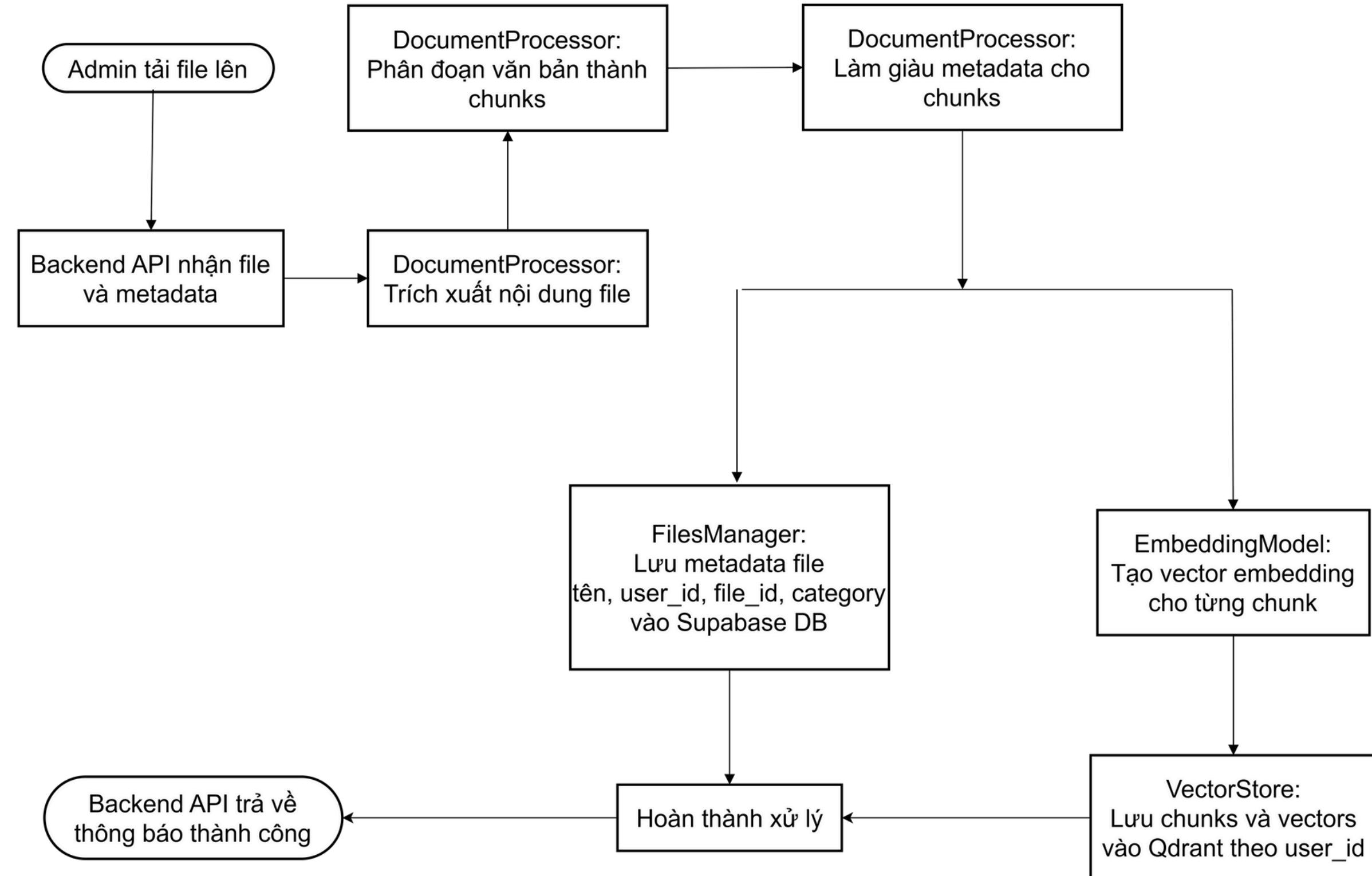
# III Phân tích và thiết kế hệ thống

## 3.2 Công nghệ sử dụng

Hạng mục	Công nghệ	Lý do lựa chọn
Backend	Python, FastAPI	Hệ sinh thái AI/ML mạnh mẽ, hiệu năng cao, phát triển nhanh.
Frontend	Next.js, React, TailwindCSS	Framework hiện đại, tối ưu hiệu suất, an toàn và linh hoạt.
Vector DB	Qdrant	Hiệu năng cao, hỗ trợ metadata filtering, dễ triển khai.
Relational DB	Supabase (PostgreSQL)	Cung cấp BaaS, tích hợp xác thực, lưu trữ quan hệ tiện lợi.
LLM	Google Gemini API	Mô hình ngôn ngữ mạnh mẽ, có khả năng xử lý đa dạng tác vụ.
Triển khai	Docker, Nginx	Đóng gói và triển khai nhất quán, dễ dàng quản lý.

# III Phân tích và thiết kế hệ thống

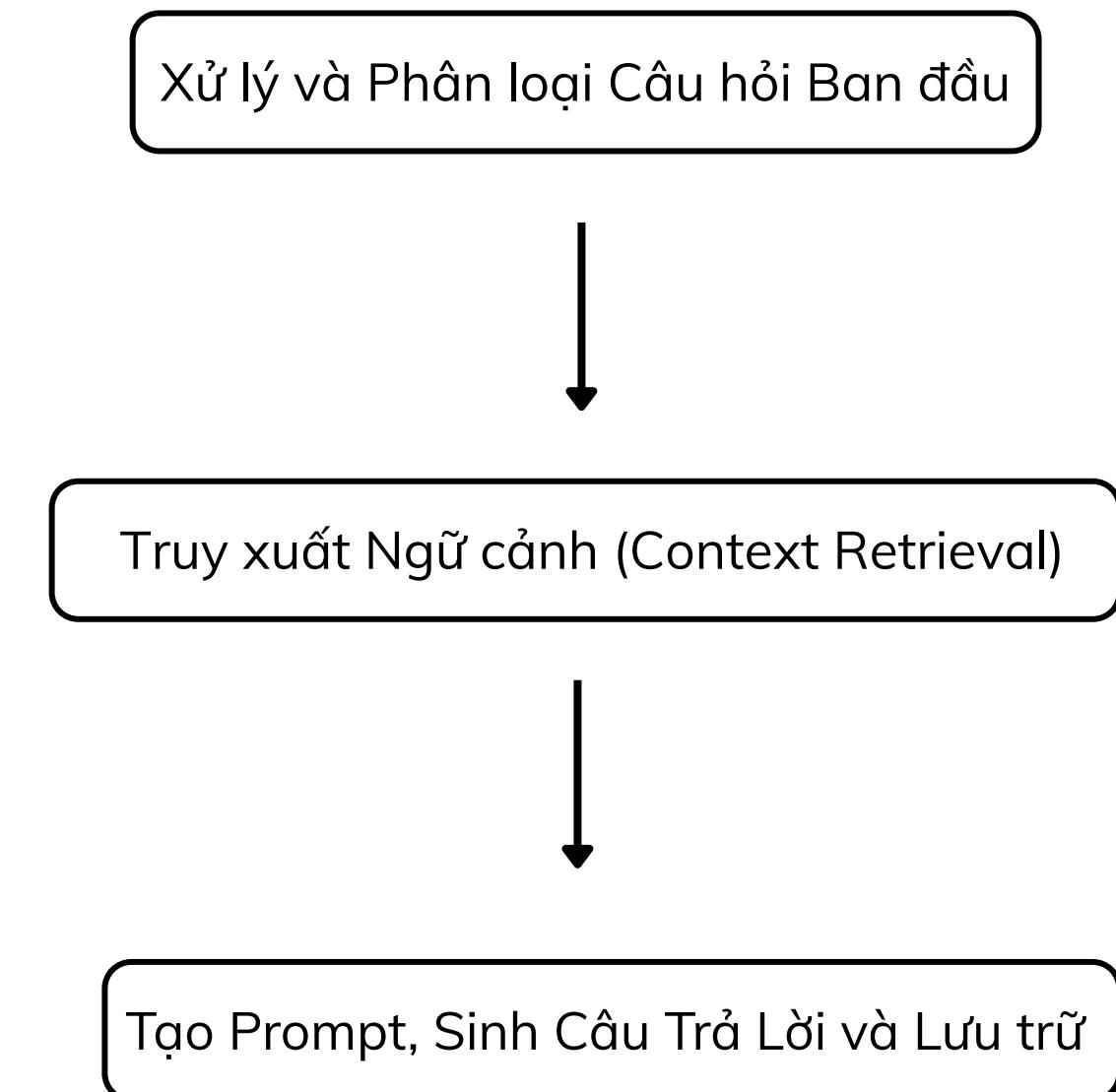
## 3.3 Luồng Nạp Liệu



### III Phân tích và thiết kế hệ thống

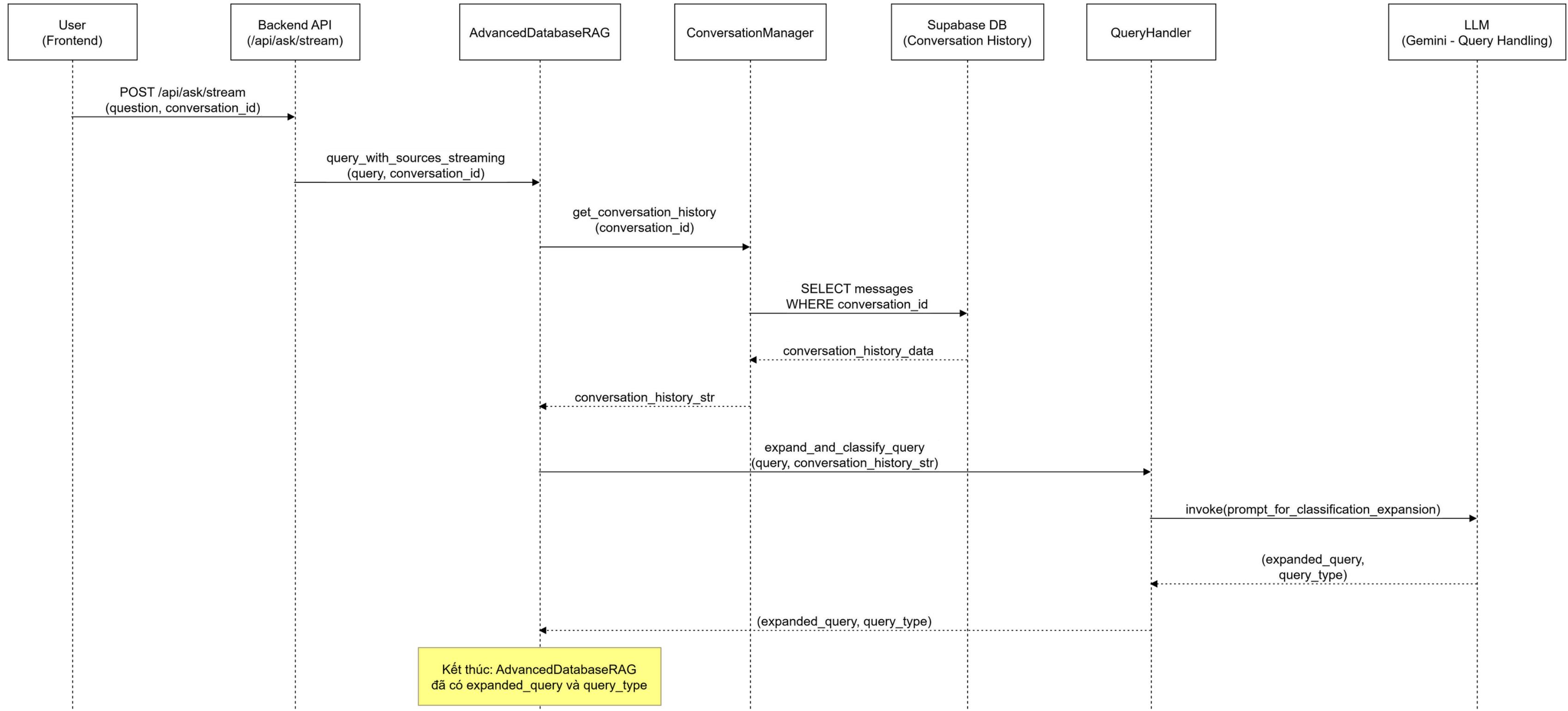
#### 3.4 Luồng hỏi - đáp

Là module xử lý trung tâm của hệ thống, được phân chia thành 3 giai đoạn:



# III Phân tích và thiết kế hệ thống

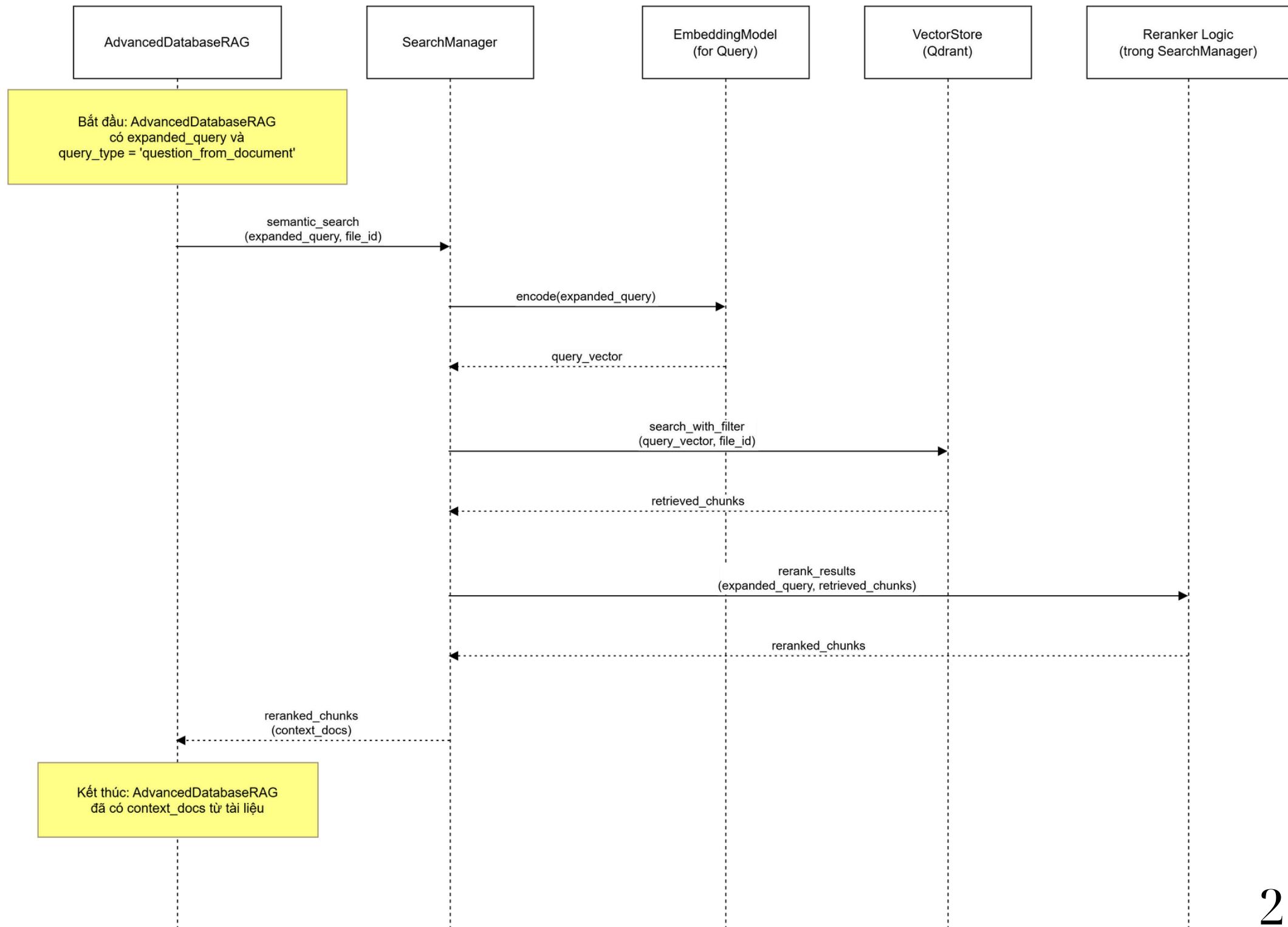
## 3.4.1 Giai đoạn 1: Xử lý và Phân loại Câu hỏi Ban đầu



# III Phân tích và thiết kế hệ thống

## 3.4.2 Giai đoạn 2: Truy xuất Ngữ cảnh (Context Retrieval)

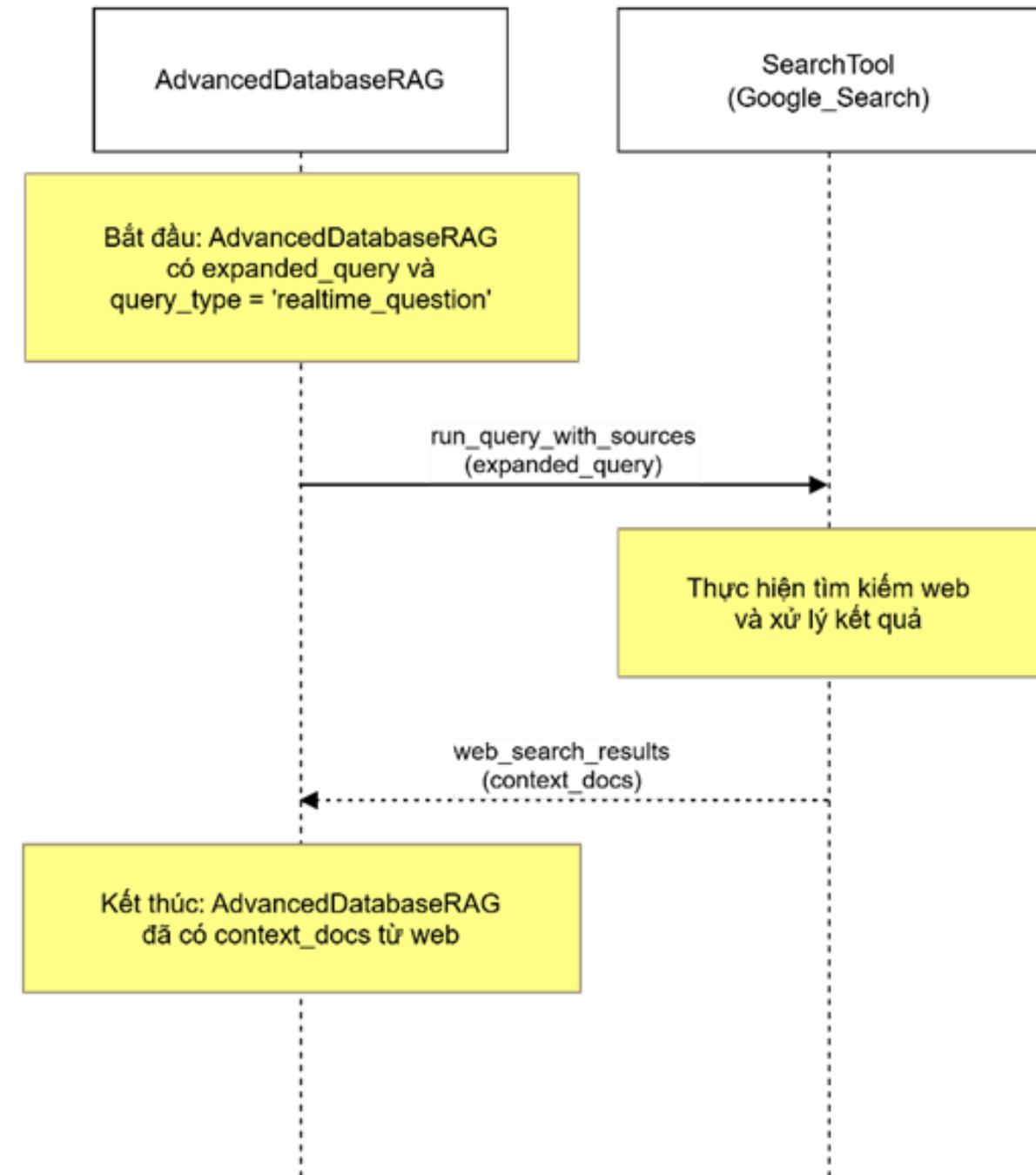
Sau khi đã có được câu hỏi đã được mở rộng và loại của câu hỏi ở bước trước thì đến bước. Nếu câu hỏi được phân thành loại ‘question from document’ - tức là câu hỏi này được hệ thống nhận dạng là thuộc về nội dung bên trong ‘Kho dữ liệu’ thì thực hiện luồng xử lý sau:



### III Phân tích và thiết kế hệ thống

#### 3.4.2 Giai đoạn 2: Truy xuất Ngữ cảnh (Context Retrieval)

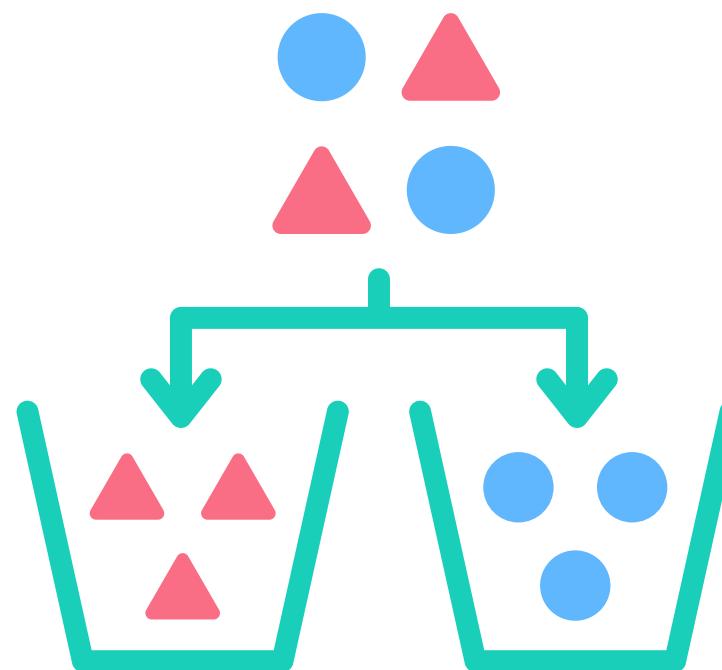
Nếu câu hỏi được hệ thống phân loại là realtime\_question thì hệ thống sẽ sử dụng công cụ tìm kiếm web tích hợp để thu thập thông tin từ internet. (*Dịch vụ search Tavily (Công cụ tìm kiếm internet)*)



### III Phân tích và thiết kế hệ thống

#### 3.4.2 Giai đoạn 2: Truy xuất Ngữ cảnh (Context Retrieval)

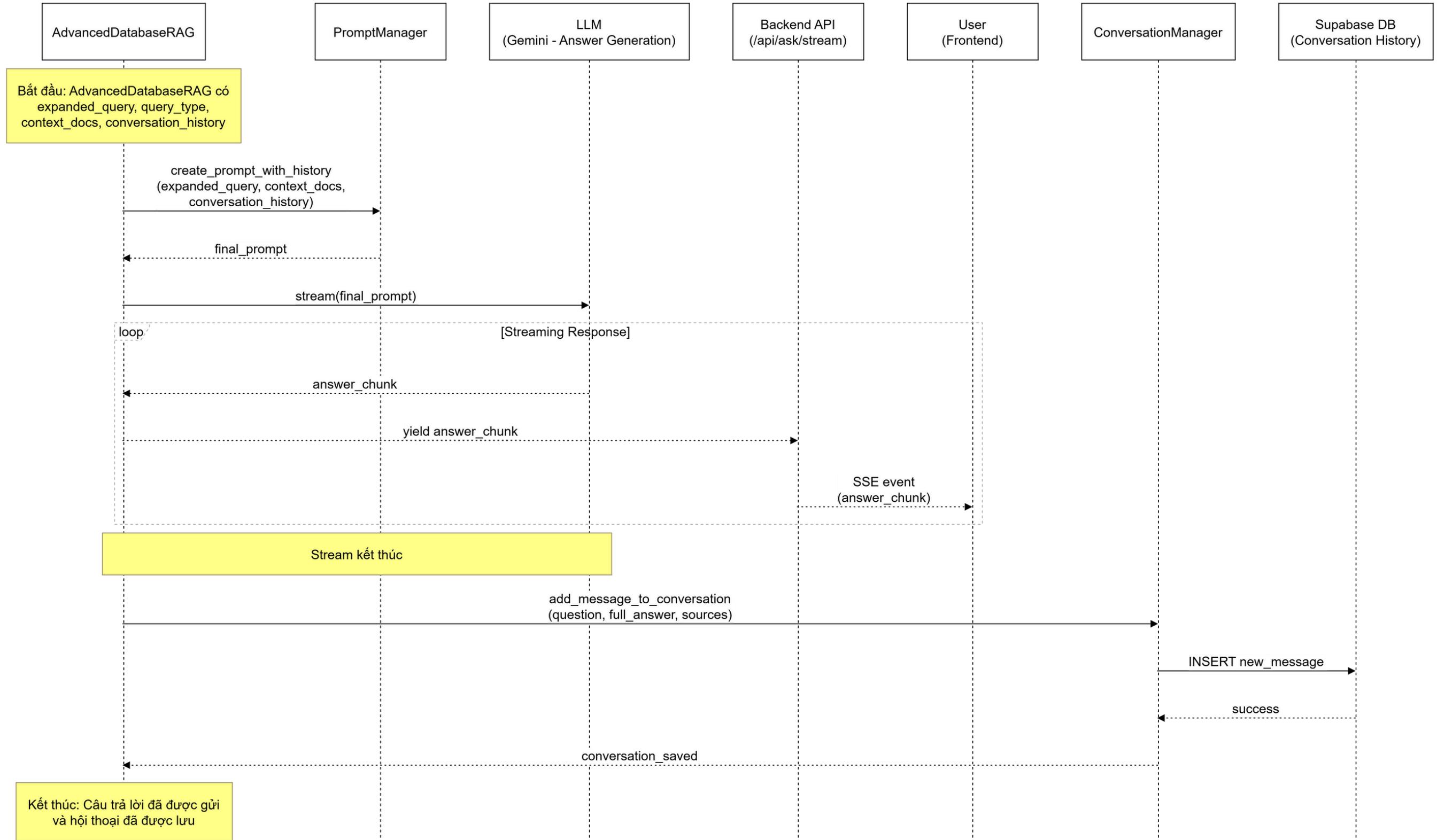
Đối với sql\_code\_task: Loại câu hỏi này thường không yêu cầu truy xuất ngữ cảnh từ tài liệu hoặc web theo cách tương tự, mà sẽ được xử lý trực tiếp bằng LLM kết hợp với PromptManager chuyên biệt cho code task.



Đối với câu hỏi được phân loại thành là other\_question thì sẽ trả về một phản hồi mặc định.

# III Phân tích và thiết kế hệ thống

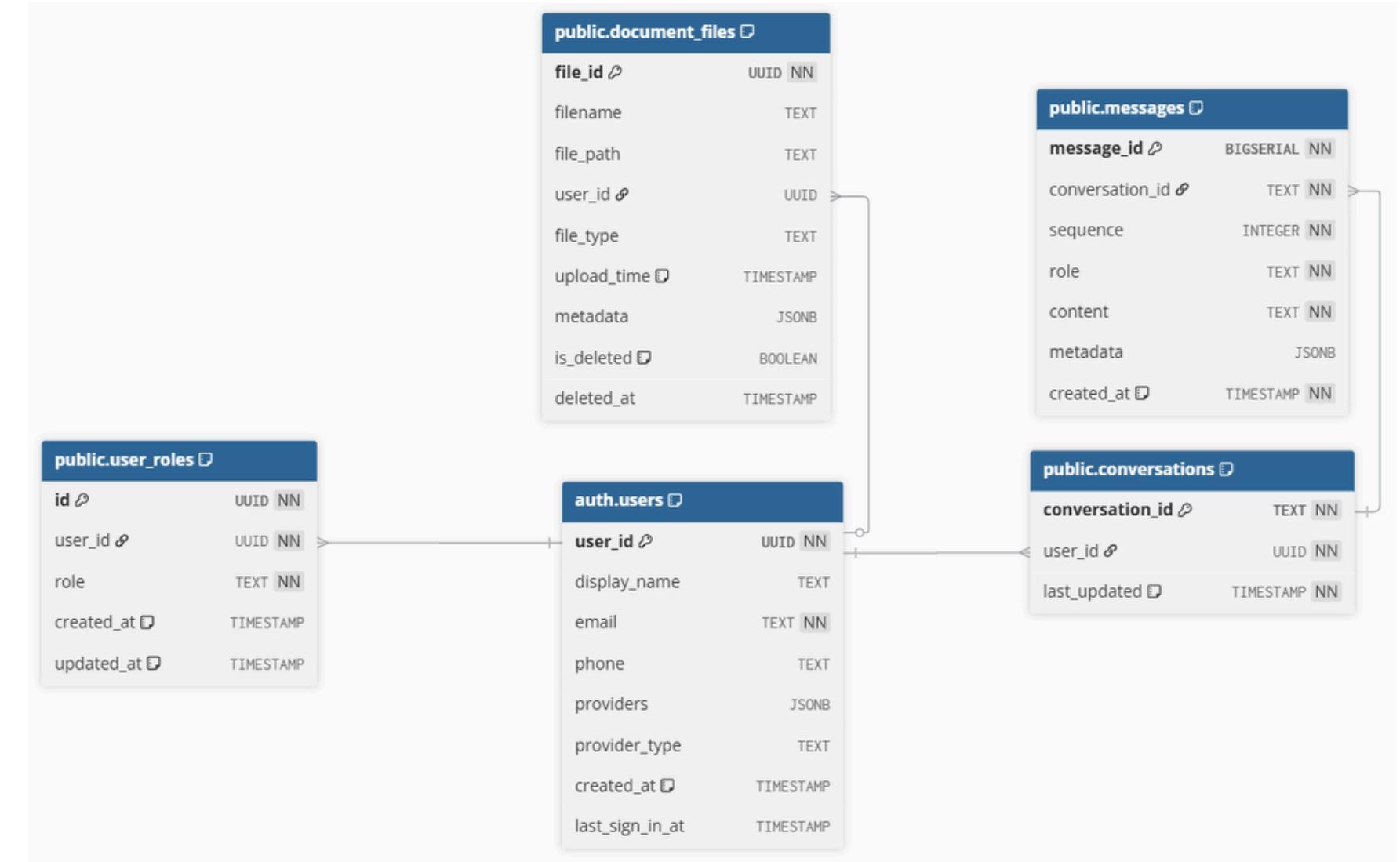
## 3.4.3 Giai đoạn 3: Tạo Prompt, Sinh Câu Trả Lời và Lưu trữ



# III Phân tích và thiết kế hệ thống

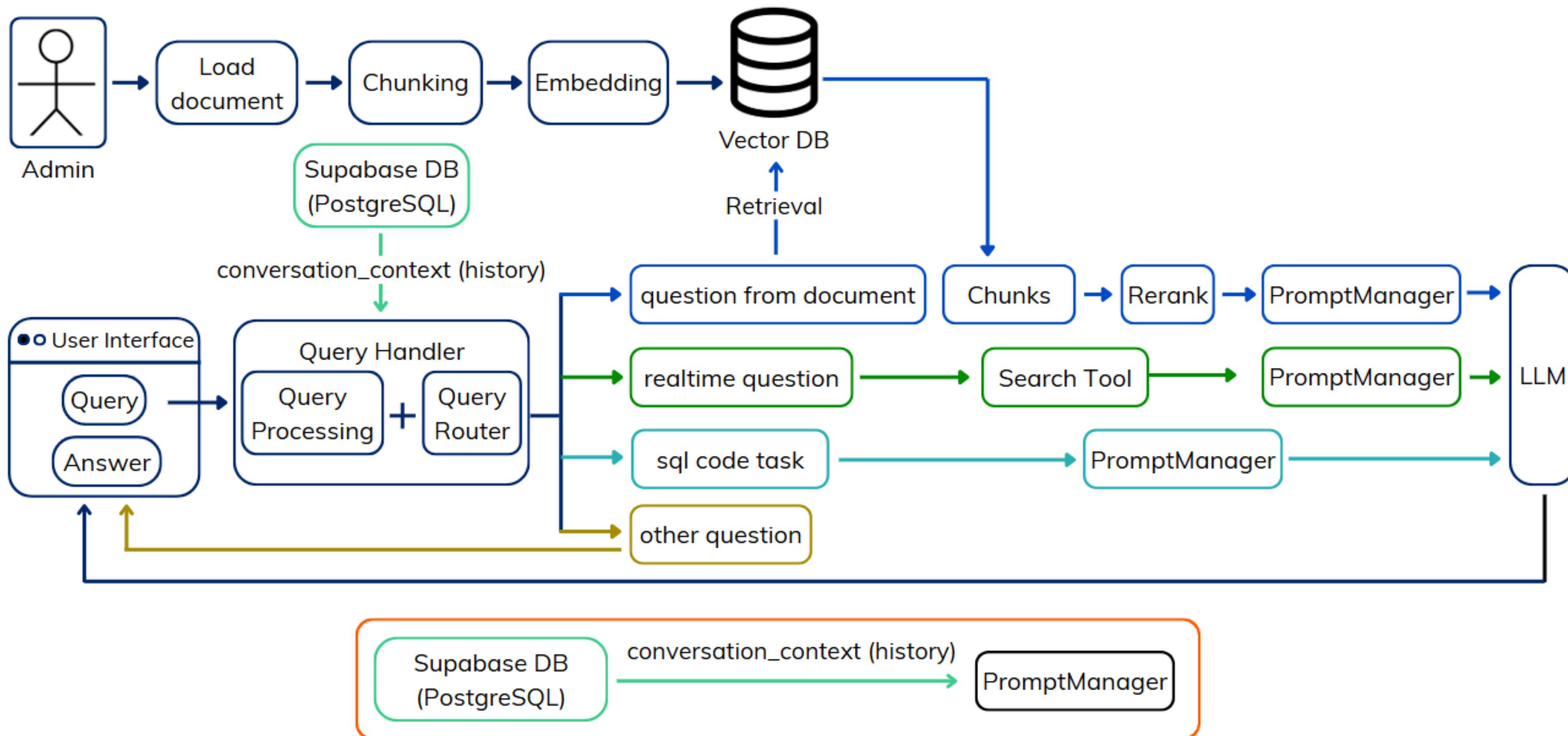
## 3.5 Thiết kế cơ sở dữ liệu

- **users:** Lưu thông tin tài khoản người dùng (được quản lý bởi Supabase Auth).
- **conversations:** Lưu thông tin về các phiên hội thoại.
- **messages:** Lưu trữ chi tiết từng tin nhắn trong mỗi hội thoại.
- **document\_files:** Lưu thông tin metadata về các tài liệu đã được tải lên và xử lý.
- **user\_roles:** Dùng để quản lý quyền của người dùng.



# IV Triển khai hệ thống

## 4.1 Sơ đồ kiến trúc hệ thống đã triển khai:



# IV Triển khai hệ thống

## 4.2 Một số giao diện của người dùng (Sinh viên)

The image displays four screenshots of user interfaces:

- Đăng nhập (Login):** A form for entering login credentials. It includes fields for Email (nguyendaihoangphuc24@gmail.com) and Password, a "Forgot password?" link, a "Login" button, and a "Or" link for Google sign-in.
- Đăng ký (Sign-up):** A form for creating a new account. It includes fields for Email (name@example.com), Password, Confirmation, a "Sign-up" button, and a "Or" link for Google sign-in.
- Hệ thống RAG - Cơ sở dữ liệu (RAG System - Database):** A conversational interface. It shows a list of recent conversations and a message from the system: "Xin chào! Tôi là trợ lý RAG chuyên về cơ sở dữ liệu. Bạn có thể hỏi tôi bất kỳ câu hỏi nào về SQL, thiết kế cơ sở dữ liệu, hoặc các khái niệm liên quan." Below this is a sidebar with links like "Có thể bạn cũng quan tâm:" and "Làm thế nào để tối ưu hóa hiệu suất của một CSDL đã xây dựng?". At the bottom is a text input field for asking questions and a SQL playground section.
- SQL Playground:** A section where users can run SQL queries. It shows a query: "SELECT \* FROM users LIMIT 10;" and its results, which are five rows of user data:

ID	Name	Email	Age
1	Nguyễn Văn A	nguyenvana@example.com	30
2	Trần Thị B	tranthib@example.com	24
3	Lê Văn C	levanc@example.com	35
4	Phạm Thị D	phamthid@example.com	28
5	Hoàng Văn E	hoangvane@example.com	42

# IV Triển khai hệ thống

## 4.2 Một số giao diện của quản trị viên (Admin)

**Dashboard**

Thống kê tổng quan hệ thống

Chủ Nhật, 22 tháng 6, 2025

**Thống kê tổng quan hệ thống**  
Dữ liệu thống kê trong 7 ngày qua

Tổng người dùng	Tổng tài liệu	Tổng hội thoại	Tổng tin nhắn
4	3	40	60
2 người dùng hoạt động	Dung lượng: 476.16 KB	Trung bình 2 tin nhắn/hội thoại	Người dùng: 30   AI: 30

**Thống kê tài liệu**  
Phân loại tài liệu theo định dạng

**Thống kê hội thoại**  
Số lượng hội thoại theo ngày

**Quản lý hội thoại**

Xem và quản lý tất cả hội thoại trong hệ thống

Chủ Nhật, 22 tháng 6, 2025

**Quản lý hội thoại**

Tổng hội thoại: 40, Tổng tin nhắn: 60, Người dùng: 2, User: 30, Tỷ lệ tin nhắn: Bot: 30

**Danh sách hội thoại** | Tim kiếm tin nhắn | Thống kê chi tiết

ID	Email	Tin nhắn đầu	Số tin nhắn	Cập nhật	Hành động
conv_9a5...	nguyendaihoangphuc24@gmail.com	Không có tin nhắn	0	22/06/2025 18:32	<span>Chi tiết</span> <span>Xoá</span>
conv_f9a...	nguyendaihoangphuc24@gmail.com	Khái niệm csdl là gì?	10	21/06/2025 17:09	<span>Chi tiết</span> <span>Xoá</span>
conv_087...	nguyendaihoangphuc24@gmail.com	Không có tin nhắn	1	20/06/2025 20:46	<span>Chi tiết</span> <span>Xoá</span>
conv_4e2...	nguyendaihoangphuc24@gmail.com	Khi nào nên sử dụng ràng buộc UNIQUE?	1	21/06/2025 03:46	<span>Chi tiết</span> <span>Xoá</span>

**Quản lý người dùng**

Quản lý tài khoản và quyền người dùng

Chủ Nhật, 22 tháng 6, 2025

**Quản lý người dùng**

Tổng người dùng: 4, Admin: 1, Students: 3

**Quản lý tài liệu**

Upload và quản lý tài liệu hệ thống

Chủ Nhật, 22 tháng 6, 2025

Tổng file	PDF	DOCX	Khác
3	1	1	1
Đã upload	Tài liệu PDF	Tài liệu Word	TXT, MD, SQL

**Quản lý người dùng**

Quản lý tài khoản người dùng trong hệ thống

Chủ Nhật, 22 tháng 6, 2025

**Quản lý người dùng**

Tìm kiếm theo email...

Email	Vai trò	Trạng thái	Ngày tạo	Đăng nhập cuối	Hành động
panda@gmail.com	Student	Hoạt động	07:42:26 19/6/2025	Chưa có	<span>Chi tiết</span> <span>Xoá</span>
student1@gmail.com	Student	Hoạt động	18:29:57 18/6/2025	Chưa có	<span>Chi tiết</span> <span>Xoá</span>
nguyendaihoangphuc24@gmail.com	Student	Hoạt động	14:56:21 18/6/2025	11:32:09 22/6/2025	<span>Chi tiết</span> <span>Xoá</span>
phucadmin@gmail.com	Admin	Hoạt động	14:55:23 18/6/2025	11:35:49 22/6/2025	<span>Chi tiết</span> <span>Xoá</span>

**Quản lý tài liệu**

Upload và quản lý tài liệu hệ thống

Chủ Nhật, 22 tháng 6, 2025

**Quản lý tài liệu**

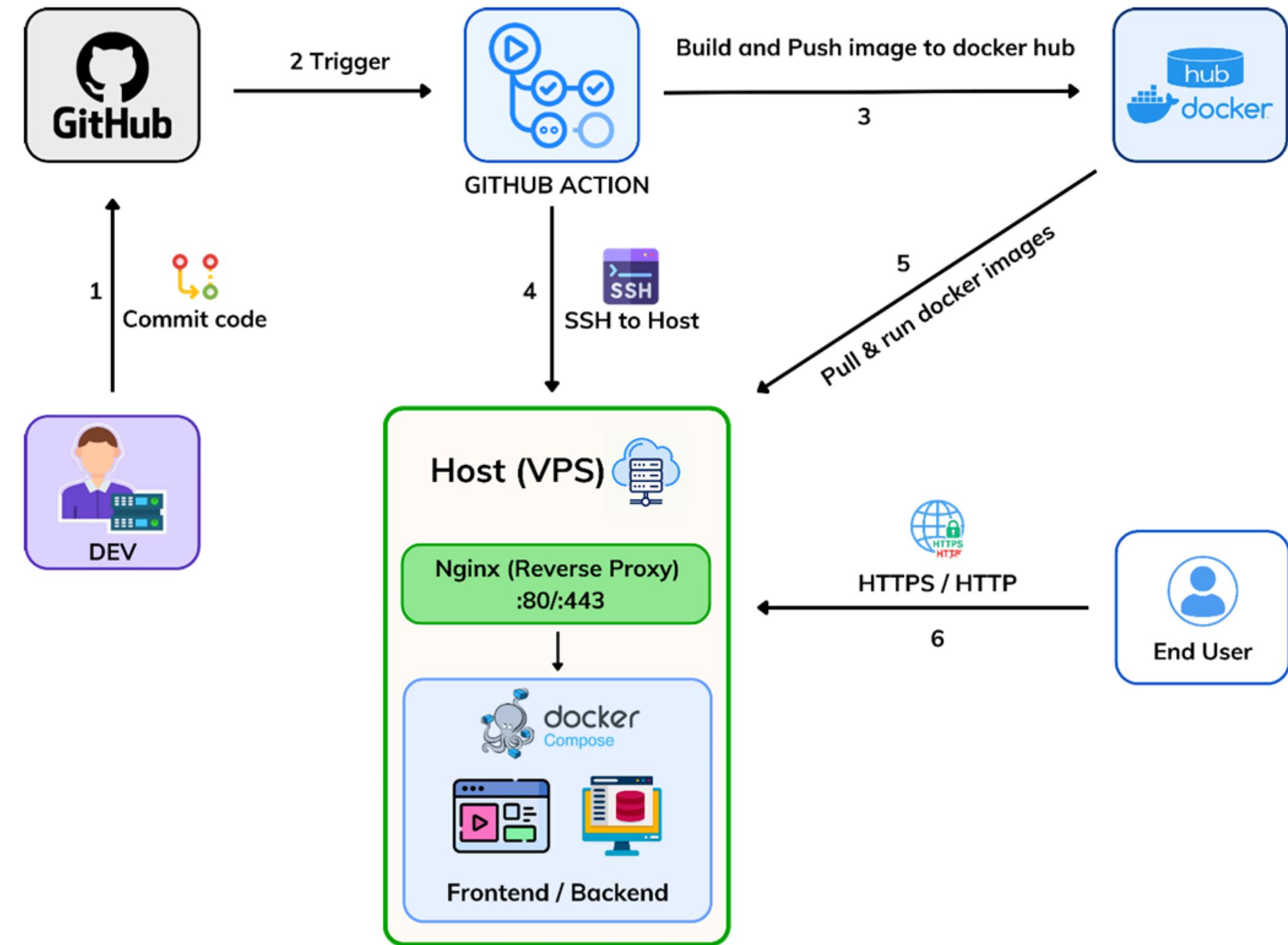
Tìm kiếm file...

File	Kích thước	Danh mục	Ngày upload	Hành động
CSDL_KLTN_GYM.sql	58.84 KB		02:21:09 20/6/2025	<span>Chi tiết</span> <span>Xoá</span>
Xu hướng hiện nay của hội nhập kinh tế quốc tế.docx	32.99 KB		02:20:39 20/6/2025	<span>Chi tiết</span> <span>Xoá</span>
Li_thuyet_Hadoop.pdf	384.33 KB		16:48:04 19/6/2025	<span>Chi tiết</span> <span>Xoá</span>

29

# IV Triển khai hệ thống

## 4.3 Tích hợp hệ thống và Đóng gói/Triển khai



# V Đánh giá và kết quả

## 5.1 Phương pháp đánh giá

Để đánh giá hiệu quả của chatbot, cần xây dựng một bộ câu hỏi kiểm thử, bao quát các chủ đề chính của môn học.

**Các tiêu chí đánh giá:**

**1. Độ liên quan của câu trả lời (Thang điểm 0-2):**

- 0: Không liên quan.
- 1: Liên quan một phần.
- 2: Hoàn toàn liên quan.

**2. Tính đúng đắn dựa trên nguồn (Có/Không):**

Nội dung trả lời có khớp với thông tin trong tài liệu được trích dẫn không?

**3. Tốc độ phản hồi (Response Time):**

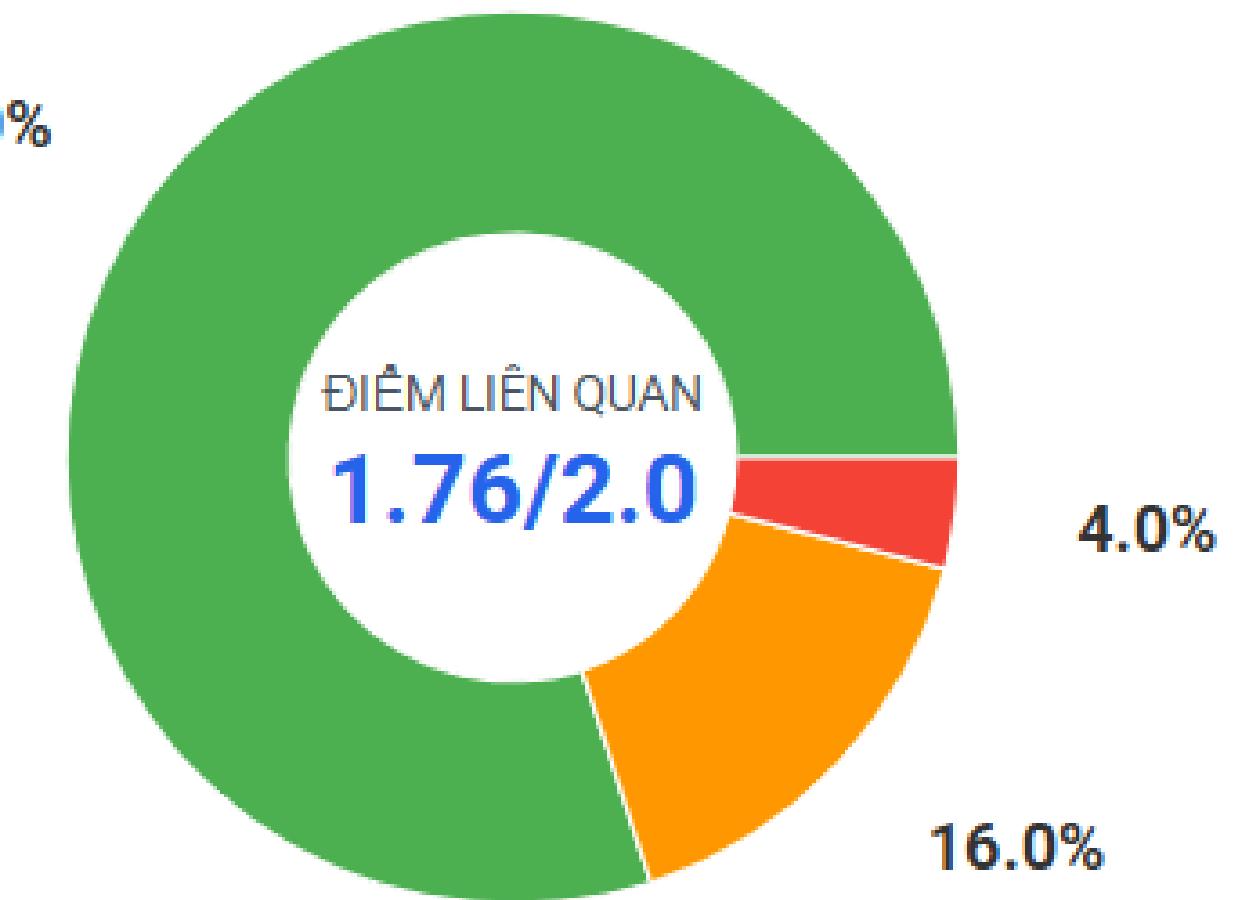
Thời gian (giây) từ lúc gửi câu hỏi đến khi nhận được câu trả lời hoàn chỉnh.



# V Đánh giá và kết quả

## 5.2 Kết quả đánh giá:

### 5.1.1 Kết Quả: Độ Liên Quan

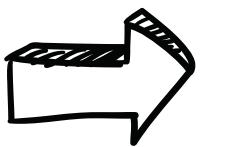
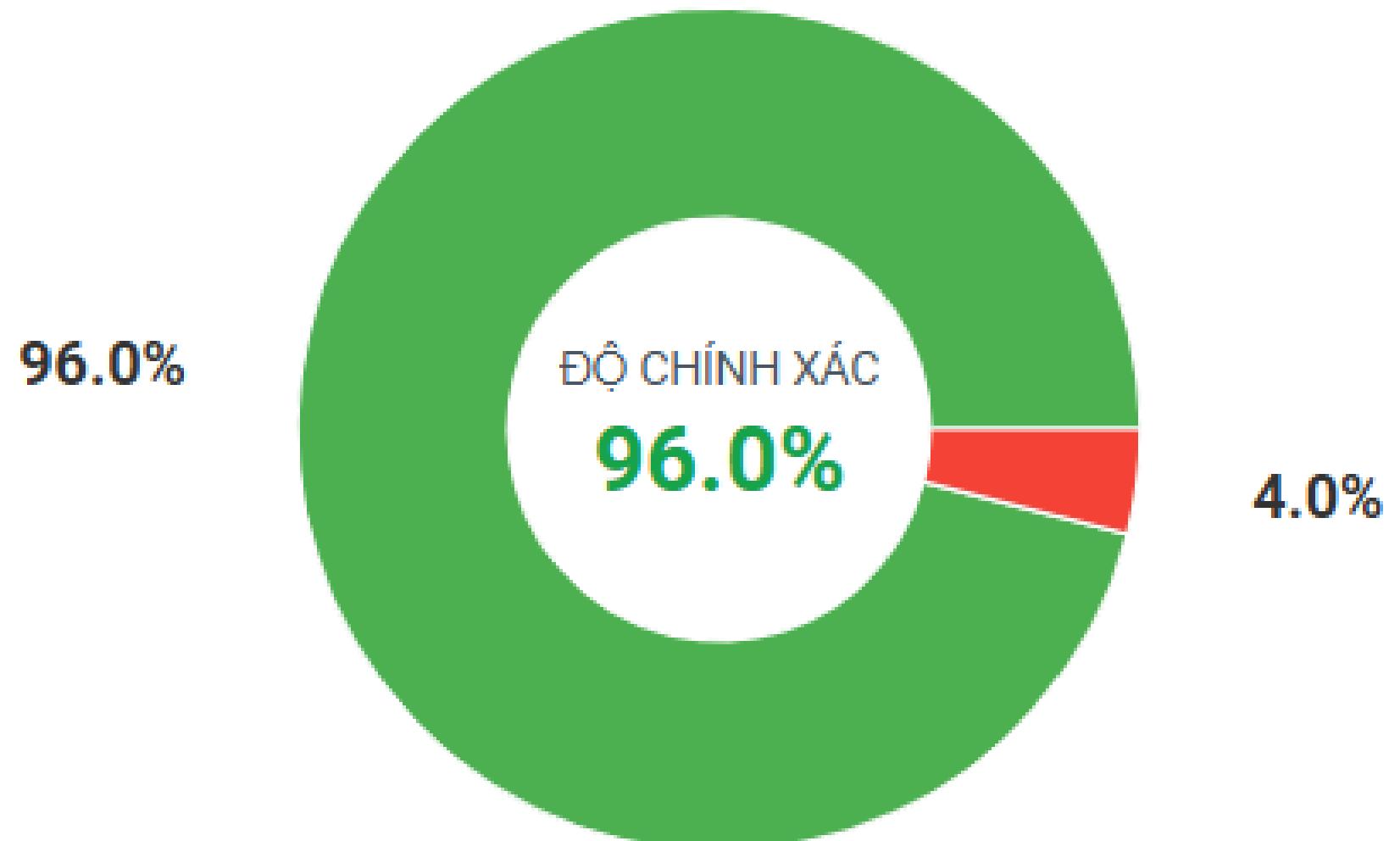


**Nhận xét:** Hệ thống có khả năng hiểu và cung cấp câu trả lời đúng trọng tâm cho phần lớn các câu hỏi.

# V Đánh giá và kết quả

## 5.2 Kết quả đánh giá:

### 5.1.2 Kết Quả: Tính Đúng Đắn (Khớp Nguồn)

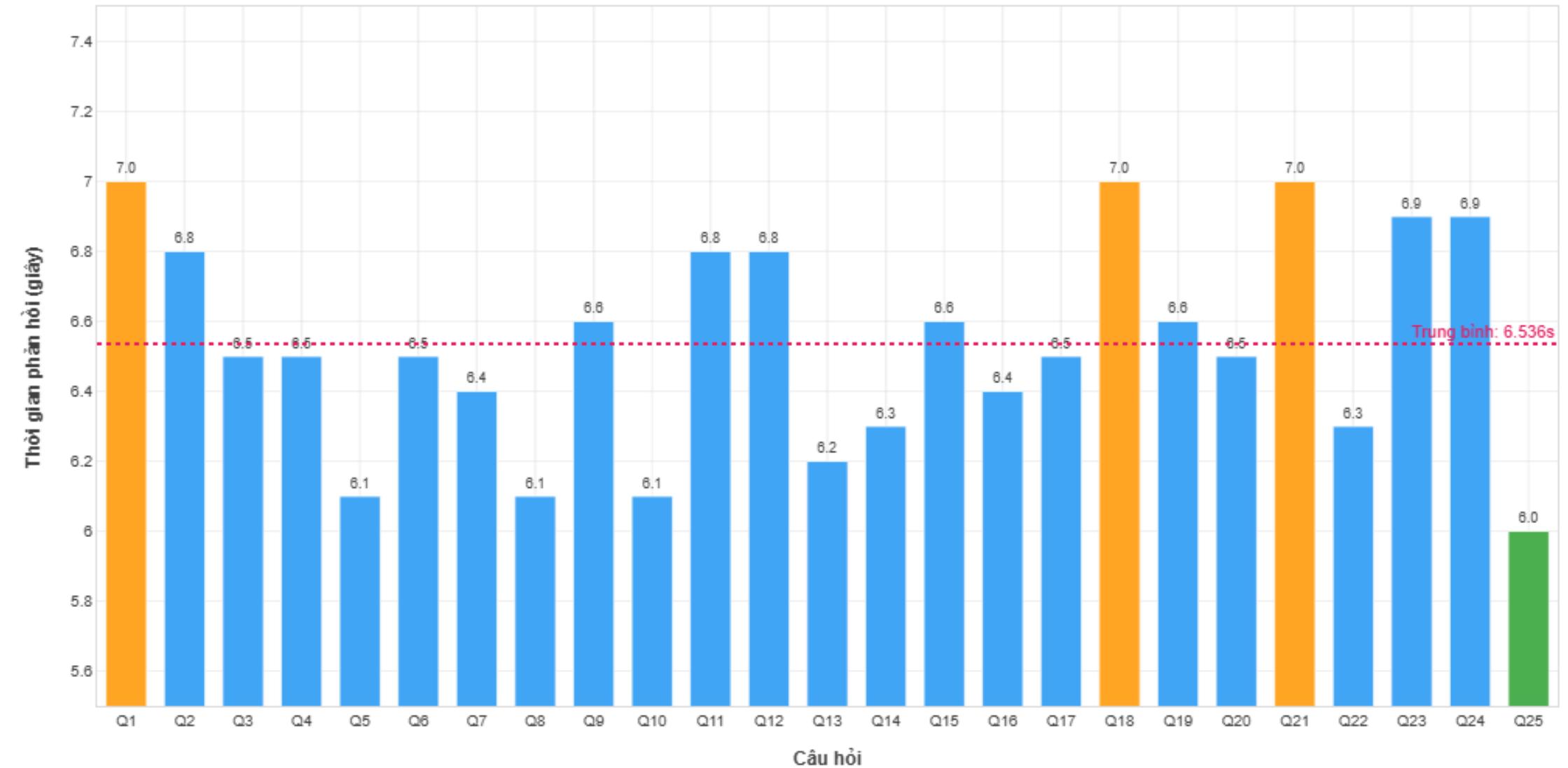


**Nhận xét:** Tỷ lệ khớp nguồn cao cho thấy kiến trúc RAG hoạt động khá hiệu quả trong việc giảm thiểu "ảo giác" và cung cấp thông tin đáng tin cậy từ tài liệu gốc.

# V Đánh giá và kết quả

## 5.2 Kết quả đánh giá:

### 5.1.3 Kết Quả: Tốc Độ Phản Hồi



**Nhận xét:** Thời gian phản hồi khá nhanh và ổn định, nằm trong khoảng chấp nhận được, đảm bảo trải nghiệm người dùng mượt mà, không phải chờ đợi quá lâu.

# V Đánh giá và kết quả

## 5.3 Tổng kết và nhận xét chung:

- **Độ liên quan cao:** **80%** câu trả lời đạt “Hoàn toàn liên quan”, thể hiện khả năng nắm bắt đúng trọng tâm câu hỏi.
- **Tính chính xác nguồn tốt:** **96%** câu trả lời khớp với trích dẫn từ tài liệu tài liệu đầu vào, hạn chế tình trạng “hallucination”.
- **Tốc độ phản hồi ổn định:** Dao động hẹp (**6,0-7,0s**), duy trì trải nghiệm tương tác mượt mà.



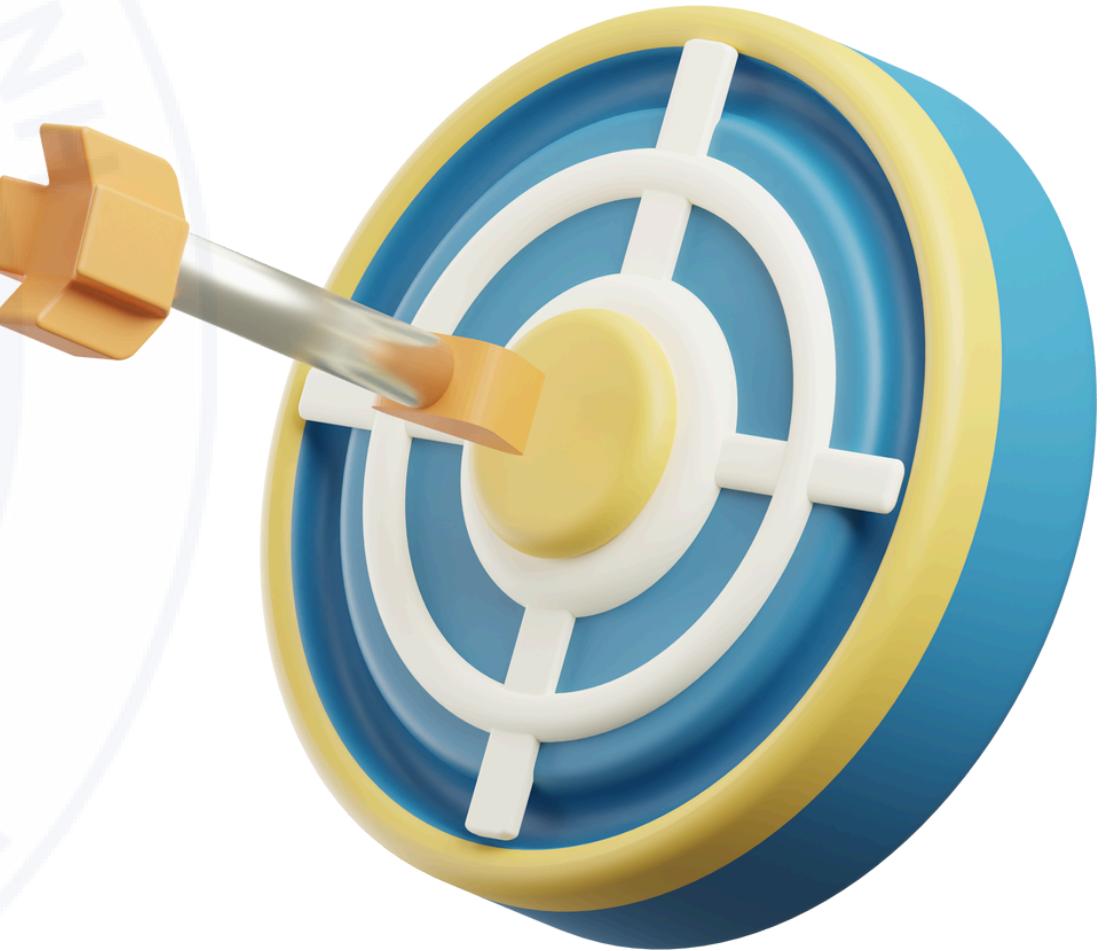
➡ Kiến trúc RAG đã chứng minh được hiệu quả trong bối cảnh cụ thể của đề tài.

# VI Kết luận và hướng phát triển

## 6.1 Kết luận:

### Kết quả đạt được:

- Đề tài đã nghiên cứu và áp dụng thành công kiến trúc RAG để xây dựng một hệ thống chatbot hoàn chỉnh.
- Hệ thống đã đáp ứng được các mục tiêu đề ra: hỗ trợ sinh viên học tập môn CSDL một cách hiệu quả, chính xác và tin cậy.
- Kết quả đánh giá định lượng cho thấy hệ thống hoạt động hiệu quả và ổn định.



# VI Kết luận và hướng phát triển

## 6.1 Kết luận:



### Đóng góp của đề tài:

- Xây dựng một công cụ học tập hữu ích, có tiềm năng ứng dụng thực tế.
- Cung cấp một ví dụ thực tiễn về việc triển khai hệ thống RAG từ đầu đến cuối, từ phân tích, thiết kế, đến triển khai và đánh giá.

## 6.2 Hướng phát triển



- **Nâng cao chất lượng RAG:** Áp dụng các kỹ thuật chunking tiên tiến hơn, thử nghiệm các mô hình embedding/LLM tối ưu cho tiếng Việt.
- **Mở rộng tính năng:** Tích hợp khả năng xử lý đa phương thức (hiểu hình ảnh, sơ đồ), phát triển công cụ gõ lỗi và tối ưu SQL.
- **Đánh giá toàn diện:** Thực hiện khảo sát người dùng để thu thập phản hồi về trải nghiệm và tính hữu ích thực tế.

# XIN CẢM ƠN

QUÝ THẦY, CÔ VÀ CÁC BẠN ĐÃ THEO DÕI  
BÀI THUYẾT TRÌNH VÀ ĐÓNG GÓP Ý KIẾN.

