Тестирование приложения с помощью curl

Ваше приложение — FastAPI-сервер на порту 8000. Основной эндпоинт для RAG — POST /query (полный цикл: поиск + генерация ответа). Вот готовый curl-запрос для теста. Запустите его после docker-compose up (сервер должен быть доступен на http://localhost:8000).

1. Тест /health (GET — проверка статуса сервисов)

bashcurl -X GET http://localhost:8000/health

Ожидаемый ответ (JSON): {"status": "degraded", "timestamp": "...", "services": {"api": true, "qdrant": false, "vllm\_gen": false, "vllm\_embed": false}} (degraded, если vLLM/Qdrant недоступны; healthy, если всё OK).

Цель: Быстрая проверка, что API живой.

2. Тест /query (POST — полный RAG-запрос)

bashcurl -X POST http://localhost:8000/query \

-H "Content-Type: application/json" \

-d '{

"query": "Что такое RAG?",

"collection\_name": "docs",

"top\_k": 3,

"score\_threshold": 0.4,

"temperature": 0.1,

"use\_hybrid": true,

"bm25\_weight": 1.2,

"use\_rerank": true

}'

Параметры (из QueryRequest):

query: Вопрос (обязательный).

collection\_name: Имя коллекции в Qdrant (обязательный; замените на вашу, e.g., "test\_coll").

top\_k: Кол-во документов (по умолчанию 5).

score\_threshold: Порог релевантности (0.4 — стандарт).

temperature: Для генерации (0.1 — deterministic).

use\_hybrid: Гибридный поиск (true — рекомендуется).

bm25\_weight: Вес BM25 (1.2 — баланс).

use\_rerank: Rerank (true — улучшает точность).

Ожидаемый ответ (JSON, QueryResponse):

json{

"answer": "RAG — это Retrieval-Augmented Generation...",

"sources": [

{

"file\_name": "example.pdf",

"chunk\_text": "RAG — Retrieval-Augmented Generation...",

"relevance\_score": 0.85,

"chunk\_id": "chunk1",

"search\_type": "hybrid"

}

],

"query": "Что такое RAG?",

"collection\_name": "docs",

"processing\_time": 2.3,

"status": "success",

"search\_type": "hybrid"

}

Цель: Полный тест (векторизация → поиск → генерация). Если Qdrant/vLLM недоступны — ошибка 500 с деталями в логах.

3. Тест /search (POST — только поиск документов)

bashcurl -X POST http://localhost:8000/search \

-H "Content-Type: application/json" \

-d '{

"query": "Что такое RAG?",

"collection\_name": "docs",

"top\_k": 5,

"score\_threshold": 0.4,

"use\_hybrid": true

}'

Ожидаемый ответ: {"query": "...", "documents": [...], "total\_found": 2, "status": "success"} (список чанков без генерации).

Советы по использованию

Форматирование JSON: Если в терминале ошибка — используйте jq: curl ... | jq . (красиво выводит JSON).

Логи: В другом терминале docker-compose logs -f app — увидите шаги (embedding, search, gen).

Ошибки: Если 500 — проверьте env (.env с VLLM\_HOST и т.д.), Qdrant запущен ли.

Swagger UI: Откройте http://localhost:8000/docs — интерактивный тест без curl.