Healthcare Knowledge Assistant

Submission Dossier – SLLM-AAI-MD SHAEK AHMED

Comprehensive deliverable for the Acme AI Sr. LLM / Backend Engineer assignment. This dossier consolidates source links, operational notes, testing coverage, and Continuous Integration / Continuous Delivery (CI/CD) disclosures.

Deliverables Overview

Source Code (GitHub Repository)

README (setup workflow + design notes)

Dockerfile

CI Pipeline – ci.yml

GitHub Pages Status Workflow – static.yml

Continuous Delivery Status Page

Docker-Only Setup Instructions

Install Docker Desktop and ensure the daemon is running.

Clone the repository: `git clone https://github.com/shaek666/Healthcare-Knowledge-Assistant.git`.

Set an API key in the terminal (e.g. PowerShell: `$env:HKA_API_KEY = "your-secret-key"`).

Build the container image: `docker build -t healthcare-knowledge-assistant .`.

Run the backend container: `docker run -e HKA_API_KEY=$env:HKA_API_KEY -p 8000:8000 healthcare-knowledge-assistant`.

Optional: mount `-v C:\hka\data:/app/data` (or `${PWD}/data` on POSIX) to persist FAISS indices.

Browse to http://localhost:8000/docs, authorize with the API key, and exercise `/ingest`, `/retrieve`, and `/generate`.

API Catalogue

Endpoint	Method	Description
/ingest	POST (multipart)	Accepts `.txt` documents in English or Japanese. Automatically detects lang
/retrieve	POST (JSON)	Processes a free-form query and returns top matches with cosine similarity s
/generate	POST (JSON)	Produces a mock summary grounded in retrieved passages. Supply `outputl

Automated Testing

Integration coverage resides in `tests/API_test.py`. Run the suite inside Docker for parity with CI using `docker run --rm ghcr.io/shaek666/healthcare-knowledge-assistant:latest python -m pytest`. The harness

stubs heavyweight translators and embedding calls to keep runtime minimal.

CI/CD Pipeline Summary

Continuous Integration (`ci.yml`) runs on every push: dependencies are installed, pytest executes, the Docker image is built, and the artifact is pushed to GitHub Container Registry. Continuous Delivery is demonstrated by the GitHub Pages workflow (`static.yml`), which publishes a refreshed status page after CI succeeds. Deployment of the backend container itself is pending selection of a production host, and this next step is clearly noted on the status page.

Operational Notes

• The GitHub Pages site surfaces the latest CI/CD results and outstanding deployment work.

• Pull the published image locally with `docker pull ghcr.io/shaek666/healthcare-knowledge-assistant:latest`.

• After local changes, rebuild (`docker build ...`) and rerun the container to keep FAISS indices and documents current.

AI Usage Disclosure & Citations

Mock responses in the `/generate` endpoint were drafted with help from OpenAI's ChatGPT and manually reviewed to ensure they remain illustrative and non-clinical.

README sections covering Docker-only setup, design notes, and future improvements were composed with ChatGPT assistance and curated to match the implemented architecture.

GitHub Pages status page (records CI/CD output and pending deployment work)

References & External Links

CI Workflow Runs

Pages Workflow Runs