Coding Assignment: Multi-Agent Regulatory Assistant

Develop a multi-agent system that answers regulatory questions about medical devices.

Use of AI

We want to see that you understand what you're building. Therefore, we want ***NO*** use of AI that programs the solution for you. Of course, we use AI for programming at Nakamo, but here it's about demonstrating that you have the understanding and know what you're doing.

Task Description

Main Components:

- **Orchestrator Agent:** An LLM coordinates all other agents
- **RAG Agent:** Searches provided PDFs for relevant information
- **Response Agent:** Creates the final answer from the found information

Input/Output:

- Input: Text string with regulatory question
- Output: Text string with structured answer including source citations

Technical Requirements

- Python as programming language
- LLM integration for Orchestrator and Response Agent (free choice of LLM)
- Vector Store for RAG (e.g. ChromaDB)
- Agent communication and workflow must be self-implemented
- Allowed: LLM Function/Tool Calling APIs for individual agents
- Not allowed: Multi-agent frameworks (AutoGen, CrewAI,

LangGraph)
- The orchestrator should decide via LLM which agents to use

Provided Materials

- FDA_Policy_Device_Software_Functions.pdf
- WHO_Medical_Device_Regulations.pdf
- FDA_Design_Control_Guidance.pdf

Example Queries

- Is our AI-powered MRI analysis tool considered a medical device software?
- What are the design control requirements for verification and validation?
- Compare FDA and WHO approaches to risk management for medical devices
- What documentation is needed for a mobile app that monitors heart rate?

Code Quality

- Focus on functional prototype, not production code

Timeline & Submission

- Submission: Code with README and demo video
- Brief paper which covers the following points of the solution:
  - Key considerations for the RAG implementation
  - Prompting structure and thoughts behind it
  - Agent orchestrating with pro and cons

The goal is to demonstrate skills in LLM integration, RAG implementation, and agent orchestration.