# Full Stack Developer Evaluation Project

## 📌 Problem Statement

You are tasked with building a lightweight Q&A system that allows users to ask questions from a small set of predefined legal FAQs and receive relevant, AI-generated responses. The system should demonstrate basic retrieval-augmented generation (RAG) principles, combining data storage, retrieval, and natural language generation in a simple user-facing interface.

## 🎯 Objective

Design and implement a minimum viable product (MVP) that:

1. 1. Stores a small dataset of sample FAQs.
2. 2. Retrieves the most relevant FAQs when a user asks a question.
3. 3. Uses AI to generate a contextual answer.
4. 4. Displays the answer to the user in a simple interface.
5. 5. Logs user interactions for later review.

## ✅ Acceptance Criteria

- At least 5–10 FAQs are available in the system, and data is queryable for retrieval.
- User can enter a question through a simple interface.
- The system retrieves top 1–2 relevant FAQs.
- Retrieved FAQs are used to generate a contextual AI answer.
- AI responses appear clearly in the interface.
- User queries and AI responses are stored for later review.
- The system is simple enough to run end-to-end in a demo environment within ~2 hours.

## 📊 Evaluation Parameters

### 1. Functional Evaluation

- Correctness: Does the system retrieve relevant FAQs and provide meaningful AI answers?
- Completeness: Are all acceptance criteria implemented (data, retrieval, answer generation, logging)?
- User Experience: Is the interface clear, usable, and intuitive?
- Reliability: Does the system handle multiple queries without breaking?

### 2. Technical Evaluation

- Architecture & Design: Is the system modular, with clear separation between data, retrieval, AI, and interface layers? Is the solution extensible beyond the MVP?
- Code Quality: Readability (clear, clean, well-commented), maintainability (organized structure, minimal hardcoding), error handling (graceful handling of bad inputs or empty results).
- Efficiency: Are retrieval and AI calls efficient for the dataset size? No unnecessary complexity introduced.
- Technical Choices: Justification of chosen libraries/frameworks. Simplicity vs. over-engineering balance.