Home Programming Assignment

January 2026

PREPARED BY: DR. H.A. QURESHI
hammad.qureshi@pannimanagement.com

Take-Home Assignment: AI Legal Contract Auditor

Develop an AI solution to understand context of different aspects of a contract and label them according to the different classes or categories.

1. Project Overview

The goal is to build a prototype AI system that can ingest complex legal contracts and accurately identify specific clauses (e.g., "Change of Control," "Termination for Convenience," or "Indemnification") while providing a risk assessment for each.

2. The Dataset

You will use a subset of the CUAD (Contract Understanding Atticus Dataset).

https://www.atticusprojectai.org/cuad

Focus: Select relevant agreements from the dataset.

Objective: Extract and analyze the "IP Ownership Assignment", “Price Restrictions”, “Non-compete, Exclusivity, No-solicit of Customers”, “Termination for Convenience” and "Governing Law" clauses.

---

3. Technical Requirements

Task A: Data Pipeline & Extraction

Preprocessing: Implement a robust PDF/text parsing strategy. Legal documents often have complex headers, footers, and numbering that can break standard chunking.

Vectorization: Embed the documents into a vector database (e.g., Pinecone, Weaviate, or ChromaDB).

Task B: RAG Implementation

Build a retrieval pipeline that can answer specific questions about the contract.

Contextual Retrieval: When asked "What are the notification requirements for Price Restrictions?", the system must retrieve the exact relevant paragraph.

Long-Context Handling: Legal contracts are long. Describe or implement a strategy (like Parent Document Retrieval or Reranking) to ensure the AI doesn't lose context.

Task C: Analysis & Risk Scoring

For each identified clause, the AI should:

Summarize the clause in plain English.

Assign a Risk Rating (Low, Medium, High) based on how "vendor-friendly" or "customer-friendly" the language is.

Provide a Citation: The exact page number or section header where the info was found.

---

4. Deliverables

1. Codebase: A GitHub repository containing your processing scripts and the RAG logic (Python/LangChain or LlamaIndex preferred).

2. A README: Documenting:

- How you handled chunking strategy (e.g., fixed-size vs. semantic chunking).

- Which LLM you chose (e.g., GPT-4o, Claude 3.5 Sonnet, or an open-source model like Llama 3) and why.
- How you would handle Hallucination Monitoring in a real-world legal environment.

3. Short Demo: A sample output showing the system reviewing one contract it has never seen before.

---

5. Evaluation Criteria

Criteria	What we are looking for
Precision	Does the system find the exact clause, or does it return generic text?
Data Strategy	How did you handle the messy structure of legal PDFs?
AI Safety	Does the system admit when it cannot find a clause (the "I don't know" response)?
Scalability	Would this architecture work if we uploaded 10,000 contracts?

---

6. Bonus (Optional)

Implement a "Redline Suggestion" feature: If a clause is marked as "High Risk," have the AI suggest a more balanced alternative version of that clause.

---

7. Rules

This is a part of the interview process for the Fullstack AI Engineer position at AP Automated Ltd. advertised in January 2026. You are allowed to look at different tools and algorithms available online if they fit the following requirements:

- Should be freely available with permission for commercial use

- You must clearly state your contribution to the algorithm in improving the results
- Must be in Python
- Code must not be copied and any copied code should be marked clearly with remarks within the code
- Do not share this assignment with anyone without prior approval