Assessment Brief - anonymized for privacy.

You are tasked to build a Code Review Assistant that performs two passes over developer code:

Pass 1 — Code Review: Detect and list issues (bug risk, security, performance, readability, maintainability), with file/line, severity, and a concise suggestion.

Pass 2 — TODO Plan: Re-analyze the issues (and code) to propose a prioritized TODO plan with actionable tasks, sample patch hints, and recommended tests.

Constraints:

- Use OpenAI as LLM with the provided key.
- Build the flow using Langchain Expression Language.
- Implementation in Google Colab.
- Produce structured JSON via Pydantic/Output Parsers.
- Add a simple frontend to paste/upload code, run analysis, and download outputs.
- Do not use quick python libraries like gradio or streamlit --- if possible
- Demonstrate prompt engineering and parameter tuning capabilities

2) Deliverables

- Colab Notebook that installs and imports dependencies securely; defines tools, prompts, chains, and schemas; runs the two-pass pipeline; hosts a UI.
- Outputs (per analysis run): issues list and prioritized tasks.
- Short Notes covering prompting strategy, parameter findings, limitations & next steps.

3) System Architecture (Two-Pass)

[User code/file difference] → [Validation (regex/AST)] → PASS 1: LLM Review → PASS 2: LLM TODO Planner → [preview + JSON download]

4) Input

1. Code file before changes and code file after changes
2. Brief description of what changes have been done
   or use this pull request as reference:
   Implement complete JWT authentication system with Flask by chispasfortuitas · Pull