The Scenario:

Imagine you're building an automated system for a **Director of Engineering** who needs to prepare for their Quarterly Business Review (QBR). This Director oversees multiple projects and requires a "Portfolio Health Report" to quickly pinpoint **risks, inconsistencies, and unresolved issues**.

Your system's ultimate goal is to analyze a collection of project communications and generate a concise, **high-signal report** that tells the Director exactly where to focus their limited attention.

Your Mission & Deliverables:

Your primary task is to create a **Blueprint** for this system, along with a focused **proof-of-concept implementation** of its core analytical engine.

- **What to Explain:** Your architectural choices for data ingestion, security, cost management, and monitoring should be clearly detailed in your Blueprint document.
- **What to Code:** You'll implement the core analytical logic for the concept, it must be a working code, but it can be a lightweight one.
- **AI Usage:** You are highly encouraged to leverage AI models. Please document which models you used and justify your choices in a `README.md` file.

You will submit a link to your code (GitHub repo or Colab notebook) and your detailed `Blueprint.md`.

Provided Materials:

You will receive a `AI_Developer.zip` file. Inside, you'll find folders for several projects, with each project containing:

- `project_emails in txt`: A file with raw, multi-threaded email conversations.

This is your main deliverable, combining your strategic thinking with practical coding. The key metrics we are looking for are your mindset, the approach, and its structure.

1. Data Ingestion & Initial Processing

Task: Briefly describe how your system would ingest and initially process the provided email text files. Consider how you'd prepare this data for analysis. It is enough to explain, what approach would you chose to make it work on a large scale data, making it scalable.

Deliverable: An explanatory section in your `Blueprint.md`. A simple diagram is encouraged.

2. The Analytical Engine (Multi-Step AI Logic)

Task: The heart of this system is its ability to find and prioritize what truly matters.

- **Define 1-2 critical "Attention Flags"** your system will identify. These are specific categories of issues that demand the Director's immediate attention. Think about what a Director absolutely needs to know. Examples:

    - **Unresolved High-Priority Action Items:** Questions or tasks that have gone unanswered or unaddressed for a significant period.
    - **Emerging Risks/Blockers:** Potential problems or obstacles identified in communications that lack a clear resolution path.
- Design the **multi-step AI process** to detect it. This includes your strategy to **minimize misinformation/hallucination** and incorporate **security considerations** within the analysis process and a final report summarizing it.

Deliverable:

- An explanatory section in your Blueprint.md defining and justifying your solution.
- **Runnable Python code** that implements the detection logic.
- The final, engineered prompt(s) used in your code, presented and explained within your Blueprint.md.

3. Cost & Robustness Considerations

Task: How would you briefly address the following:

- **Robustness:** How would you design the system to be resilient against misleading or ambiguous information?
- **Cost Management:** Briefly explain your strategy for managing the operational costs, especially concerning AI model usage.

Deliverable: A concise explanatory section in your Blueprint.md.

4. Monitoring & Trust

Task: How would you ensure this system remains trustworthy and accurate in a production environment? What key metrics would you track?

Deliverable: An explanatory section in your Blueprint.md.

5. Architectural Risk & Mitigation

Task: Identify the single biggest architectural risk in the system you've designed (even in this simplified version) and propose a brief mitigation strategy.

Deliverable: A concluding section in your Blueprint.md.