1. Problem Understanding

The task was to build an intelligent system that, given a job description or natural language query, recommends up to 10 relevant SHL assessments from their product catalog. The goal was to replace keyword-based filtering with a more semantic, AI-driven recommendation process.

---

2. My Approach

I followed a Retrieval-Augmented Generation (RAG-like) approach using embeddings:

- First, I collected and cleaned SHL's product catalog, combining each product's name and description into a searchable format.
- I used the sentence-transformers library (all-MiniLM-L6-v2) to generate vector embeddings for both the catalog entries and user queries.
- I indexed these vectors using FAISS for efficient similarity search.

When a user enters a job title, job level, and job description — or uploads a resume — I generate a prompt using that data, encode it as a query vector, and search for the top 5–10 closest assessments using cosine similarity.

---

3. Enhancements & Bonus Features

- **Prompt Tuning**: Structured prompts improve semantic matching and guide the model to focus on role-relevant traits.
- **Resume Upload**: Users can upload a PDF resume, from which I extract and auto-fill the job description using PyMuPDF.
- **Relevance Scoring**: I rank results based on normalized similarity scores to show how confident the model is in each match.
- **UI/UX**: I used Streamlit to build a clean, responsive interface that can run fully in the browser.

---

4. Tech Stack

- Streamlit for UI
- sentence-transformers for embeddings
- FAISS for indexing and nearest-neighbor search
- PyMuPDF for PDF text extraction
- pandas for catalog handling

---

5. Final Submission

- **Web App**: (live link)
- **GitHub Repo**: (GitHub link)