**SHL Assessment Recommendation System — Approach Summary**

**Objective:**

To develop a recommendation system that returns the most relevant SHL assessments based on user input. The system is accessible via a web interface and a RESTful API that accepts a query string and returns a ranked list of assessments in JSON format.

**Tools & Libraries Used:**

* **Web Scraping**: requests, BeautifulSoup
* **Data Processing**: pandas, re
* **Text Embedding**: sentence-transformers (MiniLM-L6-v2)
* **Similarity Search**: FAISS, NumPy
* **API Deployment**: FastAPI
* **Web Interface :** Streamlit

**Step-by-Step Implementation:**

**1. Data Scraping**

* Scraped assessment metadata and details from paginated SHL product catalog using requests + BeautifulSoup.
* For each assessment, also visited individual detail pages to extract information like:
  + Duration, Description, Job Levels, Remote Testing Support, and Test Type.

**2. Data Cleaning**

* Missing values in critical fields like Job Levels and Languages were imputed with default values.
* Rows missing duration were dropped.
* Final dataset saved as shl\_prepackaged\_solutions\_detailed.csv.

**3. Text Embedding + Indexing**

* Concatenated assessment fields into text format.
* Embedded the data using MiniLM-L6-v2 model via sentence-transformers.
* Embeddings normalized and indexed using cosine similarity via NumPy (or optionally FAISS for large-scale use).

**4. Recommendation Logic**

* A user query is embedded and compared to dataset embeddings using cosine similarity.
* Top k results are returned based on score.
* Output includes: Assessment Name, URL, Duration, Remote Support, IRT Support, and Similarity Score.