

AI Engineer - Technical Assessment

1. System Design Task (Core)

Goal: Evaluate how you would build a module that:

- Suggests **job titles** based on a user's free-text description
- Suggests **additional skills** based on already selected skills
- Uses semantic similarity + vector database (preferred, but can be argued that it is not required.)
- Allows you to add your own thought process or improvements

Your Task:

Explain how you would design this system end-to-end.

In your answer include:

- How you would process text inputs
- How you would use embeddings & a vector DB
- How you would combine semantic similarity with popularity or other ranking logic
- What the final output structure would look like
(High-level overview only - no need to go deep into internals.)

2. Practical Task (Light Coding)

Implement a **minimal working demo** (in any language + any libraries) that:

1. Loads an embedding model of your choice
2. Stores a small list of:
 - job titles
(you can create your own items or take an existing taxonomy like EMSI)

- skills
(you can create your own items or take an existing taxonomy like EMSI)
3. Embeds them into a vector store (can be in-memory)
 4. Given the text:
"Senior backend engineer with ML interests"
return the most relevant job titles, or based on job title suggest additional job titles.
 5. Given the selected skills:
["Python", "Vector Databases"]
return additional suggested skills
 6. Include a short explanation of how you approached the ranking.
-

3. Thought Process

In a short paragraph, explain:

- When semantic similarity might fail
- How you would mitigate such cases (fallback logic, filters, metadata, reranking, etc.)

This section is important — we want to understand your reasoning, not only the code.

4. Optional Bonus

For more insight you can try and use GloPros search and you will be suggested skills/job titles, but I can give you a hint that it is not working great

Think of ways how it should work in production, what are some additional things that user would benefit from for this suggestion feature.