🕶️

Vibe Mapping Agent [SHARED]

This task tests your ability to **translate fuzzy human intent** into structured, logic-driven systems.

🎯 Goal

Build a conversational agent that takes in a **vibe-based shopper query** (e.g., *"something cute for brunch"*), then

1) asks **targeted 1-2 follow-ups (not more)**,

2) **maps to "inferred" attributes**,

and combines both 1 and 2 to **recommend products** from a structured catalog — along with a **justification** response to the shopper.

💡 Example Flow (for inspiration)

Shopper: *"Something casual for a summer brunch"*

Agent: "Lovely! Do you have a preference between dresses, tops & skirts, or something more casual like jeans?"

Shopper: "*Probably dresses or tops and skirts"*

Agent: "Any must-haves like sleeveless, budget range or size to keep in mind?"

Shopper: *"Want sleeveless, keep under $100, both S and M work"*

Example mapping *(you can use some other approach)*

```
json
CopyEdit
{
  "category": ["dress", "top"],              // ← from shopper
  "sleeve_length": "sleeveless",              // ← from shopper
  "price_max": 100,                          // ← from shopper
```

```
  "size": ["S", "M"],                    // ← from shopper
  "fabric": ["linen", "cotton"],         // ← inferred from “summer”
  "fit": "relaxed"                       // ← inferred from “casual”
}
```

Agent Justification: “Based on your casual summer brunch vibe, I selected **breathable fabrics** like linen and cotton and **relaxed fit** in **sleeveless** dresses and tops **under $100** that match your size.”

🛠️ What You’ll Build

Component	Description	Notes
Chat-style UI or CLI	Simple interface where we input a query and see: – Follow-up questions – SKU recommendations – Brief justification	We provide `apparels_shared.csv`
Follow-up Engine	Ask **1-2** contextual follow up questions (**not more** — don’t make it a quiz :)	- See “Follow up Axes” below as starting point - Agent should only ask questions about aspects NOT provided by user
Mapping Layer	Translates vibe terms like “relaxed” or “elevated” or “cute” into structured attributes **using LLM prompts, keyword dictionaries, or embeddings**	We provide an example list in `vibe_to_attribute.text` — feel free to extend
Recommendation	Uses shopper conversation + inferred attributes to filter and recommend relevant SKUs	

📎 Follow-Up Axes

Example areas that the agent can clarify [To playing the level field in case you aren’t into fashion]

1. **Size**
2. **Budget**
3. **Category** (tops, dresses, jeans)
4. **Fit Preference** (relaxed / tailored / bodycon)
5. **Occasion / Season**
6. **Sleeve length / Knee length / Coverage Preference**

FYI: Gap gives a real world example here.

Deliverables

1. **Fully deployed Demo link (simple chatbot - no fancy UI)**
2. GitHub Repo and README

Evaluation

1. **Conversational quality (40%)**
   -
   Relevant follow ups
2. **Mapping of Vibes into inferred attributes (30%)**

   - Techniques used (LLMs, embeddings, hybrid, rule based)
3. **Mapping accuracy (15%)**
4. **Code hygiene** (15%)