Technical Task: Junior Agentic AI Engineer - Cape.io

Thank you for your interest in the Junior Agentic AI Engineer role at Cape.io. As a next step, we would like to see how you approach coding with AI.

The Challenge

We are giving you a choice between two tasks. Please choose **ONE** of the options below. Both tasks are designed to take roughly **3-4 hours**. Pick the one that you feel best demonstrates your strengths.

Option A: The Visual Intelligence Extractor (Multimodal)

Focus: Computer Vision, Unstructured Data, Structured Output.
Goal: Create a script that "watches" a short video file and extracts structured creative data.

1. **Input:** A local MP4 video file (you can use any stock video of a city or nature).
2. **Process:** Use Python to extract 3-5 keyframes.
3. **AI Task:** Send these frames to a Multimodal LLM (e.g., GPT-4o, Gemini, or Claude 3.5 Sonnet).
4. **Output:** Return a strict JSON object containing:
    - mood: (String) The emotional vibe.
    - tags: (List) Objects detected.
    - colors: (List) Dominant hex codes.
    - ad_copy: (String) A potential Instagram caption based on the visual.

Option B: The "Critic & Creator" Loop (Agentic Workflow)

Focus: Agent Orchestration, State Management, Structured Data.

Goal: Build a multi-agent workflow that creates and refines ad copy, eventually formatting it for the **Ad Context Protocol**.

1. **Input:** A Product Name (e.g., "Neon Energy Drink") and Target Audience (e.g., "Gen-Z Gamers").
2. **Process:** Create two "Agents" (functions, classes, or LangGraph nodes):
    - **The Creator:** Generates draft ad copy.
    - **The Critic:** Reviews the copy against strict rules (e.g., "Must be under 15 words," "Must contain an emoji").
3. **The Loop:**
    - The *Creator* proposes a caption.
    - The *Critic* evaluates it.
    - **If** rejected, the *Critic* sends specific feedback, and the *Creator* tries again.
    - **If** approved, the loop ends.

4. **Output:** The script must output the final approved result.

🌟 **Nice to Have (Bonus): Ad Context Protocol (AdCP) Alignment**
Cape.io is adopting the Ad Context Protocol (AdCP) to standardize how our agents communicate.

- **The Challenge:** Instead of the agents simply passing text strings back and forth, structure the **Final Output** (and optionally the intermediate messages) as a valid JSON object inspired by AdCP schemas.
- **Example Structure:** Your final output should look something like this, rather than just plain text:

None

```
{
  "adcp_version": "1.0",
  "task": "creative_generation",
  "payload": {
    "target_audience": "Gen-Z Gamers",
    "creative_assets": [
      {
        "type": "text_ad",
        "content": "Level up your game with Neon Energy! ⚡ #Gaming",
        "metadata": {
          "length": 45,
          "sentiment": "energetic"
        }
      }
    ],
    "brand_safety_check": "passed"
  }
}
```

- *Why this matters:* We want to see if you can make agents produce **structured, machine-readable data** that could theoretically be sent to a DSP or another AdCP-compliant agent.

Tech Stack Requirements

- **Language:** Python 3.
- **Frameworks:** You may use standard libraries, LangChain, LangGraph, or AutoGen.
- **API:** You may use OpenAI, Anthropic, or Gemini APIs. *If you do not have access to an API key, please let us know and we will provide a temporary one.*

🤖 **Policy on AI Assistants**

At Cape.io, we believe in using the best tools for the job. You are **allowed and encouraged** to use AI coding assistants (ChatGPT, Copilot, etc.) for this task. However:

1. **Transparency:** If you use an LLM to generate significant logic, please include a comment or screenshot of your prompt.
2. **Ownership:** You are responsible for the code. "The AI wrote it" is not an excuse for bugs.
3. **Understanding:** Be prepared to explain every line of code during the review interview.

Deliverables

Please provide a link to a GitHub repository (or a Google Colab notebook) containing:

1. Your Python code.
2. A README.md explaining:
    - Which option you chose and why.
    - How to run the code.
    - Any trade-offs you made or challenges you faced.

Good luck! We look forward to seeing what you build.