readytech

AI Engineer Technical Screening

Emerging Technology Team

We appreciate your interest in joining our Emerging Technology team. The objective of this challenge is to assess your ability to build a robust, generative AI–powered application while showcasing good coding practices, initiative, and familiarity with appropriate testing paradigms for AI features. Please read the requirements carefully.

Steps and Requirements

Step 1: Build an 'LLM Survey Bot' Web Application

1. **Question Management and Survey Creation**
   - As an admin, you can create or manage a set of questions.
   - For each question, you can **optionally** provide a text description of what a "quality answer" might look like.
     - It is your responsibility to figure out how (or if) these guidelines are integrated into your AI-based scoring or validation.
     - Some questions may have no guidelines; you should consider how to handle these cases as well.
   - After finalising your questions, you can generate public shareable links to distribute the survey. Each link should be uniquely tied to an ID for tracking.
2. **Participant Experience**
   - When a participant visits a share link, they should be greeted by a user-friendly chatbot that attempts to guide them through each question **in sequence**.
   - **However**, participants might:
     - Refer to a previous or future question when responding.
     - Decide to skip certain questions altogether (or come back to them later).
     - Provide incomplete or low-quality answers.
   - The chatbot should remain as "on-rails" as possible—encouraging the user to proceed one question at a time—but also be able to handle users who want to revisit or revise earlier answers, or move out of order.
3. **Answer Storage and Scoring**

- All responses should be stored in a reliable manner (e.g. database, file storage, or any solution you choose).
- Each answer must have an associated quality score from 1–5. The method of generating these scores is up to you, but consider whether/how the user-provided guidelines might inform that process.

Step 2: Adaptability to Budget-Constrained Models
Your product manager has informed you that you may need to switch to a lower-cost LLM in the future. Without knowing the exact model that will be selected:

1. **Demonstrate or Describe an Evaluation Approach**
   - Show how you would determine any performance impacts when changing underlying models (e.g. from a high-end model to a more budget-friendly one).
   - Consider how you might compare outputs, measure answer quality, and handle trade-offs in capability vs. cost.
2. **Consider Testing and Quality Assurance**
   - Propose how you would ensure ongoing reliability and correctness across multiple models. (Hint: Think about automated comparisons, LLM-based judge methods, or other creative testing paradigms.)
   - You do not need a full implementation, but we should see evidence of your testing strategy and reasoning.

Additional Notes and Requirements

- **Flexibility of Tools**
  - We do not have a preference for specific AI/ML frameworks or hosting services or models. Use what you are most comfortable with or what best demonstrates your strengths.
  - Non-AI components (databases, frameworks, libraries) are also up to you. Anything from plain HTML and a text file to more advanced setups is acceptable.
  - We prefer use of models from commercial providers like OpenAI, Anthropic or Gemini. There is no requirement to use local models.
- **Deliverable and Submission**
  - Please provide a link to a public GitHub repository containing your code.
  - Include a clear README with instructions on how to install, configure, and run your web application, including which model providers you use.

- Please provide a short video (can be just a screen recording) with a demo and voiceover of you walking us through your application and explaining its features. The video is meant to help us better understand your application and your thinking. Focus on the content not the video production. The video should be 2-5 minutes in length.
- We do not expect you to fully complete every single feature—partial functionality is acceptable. However, please ensure you demonstrate some coverage (at least in part) of **both** Step 1 and Step 2.
- To confirm submission please send an email back to this chain confirming task completion and providing the requested links.

- **Overall Expectations**
  - We are looking for interesting solutions that signal innovation and reliability. This application is not expected to be production grade.
  - Beyond the basics, we want you to:
    - Experiment and surprise us with novel features and approaches
    - Demonstrate/describe evaluation, observability and structured outputs for generative AI
  - Above all, we are looking for an AI engineer who enjoys building things and can learn and deliver value quickly in a very fast paced and changing field.
