Build a RAG App - Take-Home Challenge

We're focused on building the next generation of evaluation for AI systems. We love to get into the weeds, and build ourselves - and are looking for people who are just as enthusiastic about this opportunity. This project should give you a chance to learn more about us as well as get your hands dirty

We'd love to see how you approach building a RAG (Retrieval Augmented Generation) application. Don't worry - RAG apps have gotten much easier to build lately, so feel free to use any tools that'll help you get up and running quickly. What we're really interested in is seeing your thought process and where you take things from there!

Time & Expectations

Shoot for 3-4 hours of coding time, plus about an hour for documentation. No need to pull an all-nighter - we'd rather see what you can do within these constraints.

Let's Break This Down

1. Pick Your Format

First things first - choose how you want to build this. Play to your strengths!

- Jupyter Notebook
- Command Line Interface
- Web App

2. Set a Stretch Goal

Here's where you get to be creative! Pick something exciting that'll guide your tool choices. Some ideas:

- Build a slick chat interface with a focus on UX and the end product
- Run everything locally with an open-source LLM
- Deploy it to the cloud so anyone can use it
- Go full data scientist and optimize the responses and focus on performance

3. Find your Data

A RAG application answers questions about a specific set of data. This could be a book, documentation, financial data - anything really. So find a set of data, and then generate some sample questions (another place you could potentially use an LLM).

4. Pick Your Tools

Framework

Choose your framework for spinning up an app quickly:

- LlamaIndex (Docs here)
  - Quick note: Skip LlamaCloud for this one
- LangChain
  - Python fans click here
  - JavaScript folks click here

The LLM that powers the app

Lots of options here:

- The Big Players: OpenAI, Anthropic, or Gemini
  - Most have free tiers - so don't feel the need to spend money
  - Document which model you use and why
- Hugging Face hosted models (Quick signup)
- Run it locally: Mistral, Llama 2, or another open-source model
  - Perfect if you're aiming for that cost-effective angle

The Must-Haves

1. A working RAG app with some interface for Q&A
   - This should take ~75-80% of your time
2. Thorough documentation
   - Clear setup instructions - make it so anyone can follow in your footsteps
   - Tell us why you picked your tools
   - Share what worked, what didn't, and how you dealt with it
   - What would you do next if you had more time?
   - Should be ~20-25% of your time

Tips for Success

- Use those quickstart tools - no need to reinvent the wheel
- Document as you go - future you will thank you

- LLMs are your friend here, don’t be afraid to use them to help, just be sure you take the time to really understand what they tell you.
- Hit a wall? Don't spin your wheels - reach out!
- Keep it focused - better to nail the basics than half-finish three extra features

What Happens Next?

If we love what we see, we'll have you walk us through your creation. We'll geek out about your choices, maybe plug it into Arize (or our OSS tool Phoenix) and see how we can improve things.

Questions?

Stuck? Confused? Just want to sanity check something? Ping Ana or Adam - we're here to help!