# The "Agentic" Challenge: Build an AI Product

### Project Overview

Imagine you are the **Lead AI Engineer** tasked with building an **AI Agent** in a domain of your choice. The goal is to move beyond simple "Chat" interfaces and build a robust system that **thinks, acts, and self-corrects.**

---

## 1. Core Requirements (The "Must-Haves")

- **Agentic Logic:** Implement a **Reasoning** Agent (e.g., ReAct, Plan-and-Execute). The agent should not just generate a response. It must articulate its plan and use tools to execute it.
- **Tool Use:** The agent must successfully integrate and call at least **two external tools**
- **Self-Correction:** If a tool returns an error or a potential hallucination is detected, the agent should recognize the issue and try an alternative path.

## 2. Nice-to-Haves

These advanced features are optional but will make your application stand out:

- **Long-Term Memory:** The agent remembers user preferences and context from previous sessions to provide a personalized, continuous experience.
- **Agentic RAG:** Move beyond basic retrieval. The agent should **decide** when it needs to query its knowledge base versus when it can rely on its internal training data.
- **"Vibe-Coded" UI:** We love to see more than just a terminal. A modern, polished UI will give your project a professional edge.

---

## Deliverables

- **GitHub Repo:** Include a clean codebase, a `README.md` explaining your architecture, and any relevant documentation.
- **Demo Video:** A walkthrough of the agent solving a complex, multi-step task.

**Note on Costs:** You are free to use any model of your choice. Many high-performance models offer generous free tiers. **We do not expect you to spend any money to complete this assignment.**