4. Brief Reflection

What were the key limitations you faced in this challenge?
The biggest constraint was model reliability under tight linguistic rules. In `agents.ts`, the Doctor and Counselor must follow very specific phrases and JSON-only outputs, which makes small deviations (or a single invalid JSON token) cascade into retries or fallbacks. Latency was another limitation: each user turn can trigger multiple model calls (Doctor → Supervisor, and Counselor in Phase 2), and the supervisor retry loop in the API can add noticeable delays. Finally, instruction-following can be brittle—small prompt wording changes sometimes caused disproportionate behavioral shifts, even when the logic stayed the same.

How would you enhance the system to address those limitations?
I would keep the two-phase design but add more robust guardrails around output formatting and latency. Concretely: introduce a lightweight JSON repair step or schema-constrained output verification to reduce retries; Use a `gpt-5.2` model for its speed and better coherence than smaller models.

What surprised you about building this system?
How sensitive behavior was to seemingly minor prompt edits. Tiny semantic shifts—like tightening a constraint or reordering a rule—could flip the Doctor from "ask one question" to "over-probe," or change the Counselor's formatting. That fragility reinforced why the Supervisor gate and the two-phase split are important: smaller, well-defined tasks are easier for the model to follow consistently than one large, ambiguous instruction set.