

Modularity

The benefits of breaking system prompts down

- Modules are independently updatable. Fix one without rewriting everything
- Think of it like code architecture. There's a separation of concerns that makes everything cleaner and more maintainable
- Modular architecture gives you:
 - Independent updates - fix one module without touching others
 - Reusability - drop the same module into different prompts
 - Clarity - each module has one job, easy to understand
 - Collaboration - team members can work on different modules simultaneously
 - Debugging - isolate issues to specific modules instead of hunting through a massive block of text

Inputs and Outputs

How these make or break a system prompt

- Input only what you absolutely need to get an expected output. Additional info may deprioritize more important inputs
- Setup a prioritization hierarchy. *i.e. user-input > convo-history > saved-prefs*
- Similarly, output only the necessities as well. The more elements being generated, the higher the chance of confusion or deprioritization
- Take your time cycling through building => reducing => building => reducing. You will likely see that you were overthinking what you need in both parts