Design a Multi-Agent System Architecture

Scenario

You're the lead AI architect at a growing e-commerce company. The executive team wants to implement an intelligent customer service system that can handle complex inquiries requiring multiple types of expertise—product recommendations, order tracking, technical support, and billing issues. Currently, customers get frustrated because they're transferred between different departments, losing context each time.

Your Challenge

Choose and justify your architectural pattern (centralized, decentralized, or hierarchical)

Your Task

Create a comprehensive multi-agent system design that demonstrates your understanding of architectural patterns, agent specialization, and communication protocols.

Steps

1. Define Your System Architecture
   - Choose and justify your architectural pattern (centralized, decentralized, or hierarchical)
   - Explain how this choice aligns with the customer service requirements

2. Design Agent Roles and Specializations

- Define 4-5 specialized agents with clear responsibilities
- Explain how each agent's expertise contributes to the overall customer experience
- Identify potential overlaps and how you'll handle them

3. Map Communication Flows

- Design the communication protocols between agents
- Specify what information gets shared and in what format
- Address how agents will maintain customer context throughout interactions

4. Address Memory and Context Management

- Describe how agents will share and access customer interaction history
- Explain how partial solutions from one agent inform others
- Design safeguards to prevent information loss during handoffs

Submission Output

Submit a system design document (visual diagram + written explanation) that includes

- Architectural pattern choice with justification
- Agent role definitions with specialization rationale
- Communication flow diagram showing agent interactions
- Memory/context management strategy
- Brief explanation of how this design improves customer experience

Conclusion

After submitting your architecture design, you'll enter a guided conversation where you'll evaluate your design choices, identify potential failure modes, and refine your approach based on real-world implementation considerations.