4) Multi-Agent System

Objective

Build a Multi-Agent System (MAS) where multiple autonomous agents work together to achieve a common or complex goal. Agents should have clear roles, responsibilities, and communication mechanisms. This is an open-ended challenge allowing for creative use cases such as logistics, smart cities, simulations, games, disaster management, robotics, financial systems, etc.

1. Build a Multi-Agent System

- Design a system with at least 3 agents.

- Each agent should have a distinct role and decision-making logic.

- Agents must collaborate, negotiate, or coordinate to accomplish a shared task or solve a problem.

- The system should be autonomous, with agents acting without manual intervention once launched.

2. Agent Design Requirements

- Define the role, inputs, outputs, and logic of each agent.

- Describe how agents communicate (e.g., message-passing, shared data).

- Implement basic rules of behavior (state machines, rule-based, etc.).

- Consider how agents handle conflicts (e.g., resource contention).

3. Web Dashboard / Visualization

- Create a web dashboard showing agent states, tasks, and progress.

- Display communication logs or coordination mechanisms.

- Optionally include a live simulation or visualizations.

Tech stack is open-ended.

Checklist

- Multi-agent system code

- At least 3 agents with distinct logic

- Web dashboard showing progress and state

Ideas

- Smart factory simulation

- Airport traffic and maintenance

- Game AI bots (cooperative agents)

- Ecosystem sim: predator/prey/environment

Expected Output:

- Dashboard Clearly showing the roles of each agent and output from each agent.

- Communication and status of each agent (Working, Pending, successful)

- Final output of the System

Suggestion: Use case is open ended and you are free to choose any use case of your choice. Use your creativity to think of best possible use case and how impacts the people around you.