



AI Agents Solutions for the Future

Utilizing Tools in LLM

In this module, participants develop essential skills for integrating and executing functions within LLM models. They learn to create and validate tool diagrams, manage errors effectively, and implement fallback solutions. The module explores various tool types, including search, computation, and data retrieval, with a strong emphasis on security considerations. Through practical applications, students design interfaces for function execution, build result validation systems, and develop solutions to enhance tool reliability in LLM models. Example projects include creating a personal assistant, a data analysis agent, and an intelligent information retrieval system.

Deep Research & Browser Utilization

This module focuses on structuring research tasks, retrieving web data, and analyzing diverse document formats such as PDFs. Participants learn techniques for extracting, synthesizing, and verifying structured data from multiple sources. Additionally, they gain experience integrating research workflows with computer systems and applications. By the end of this module, students will be equipped to design research strategies, retrieve and validate data, and integrate AI agents with operating systems. Practical projects include building a research assistant, a document analysis tool, and a market research system.

Single-Agent Solutions (Part 1)

The first part of this module introduces the foundations of AI agent development, including memory management and task execution monitoring. Participants explore self-correcting mechanisms, chain-of-thought reasoning, and ReAct patterns for enhanced agent intelligence. They apply these concepts to design agent architectures, implement memory systems, and develop self-improving AI solutions. Example projects include a personal productivity assistant, a customer service agent, and an AI-powered educational tutor.

Single-Agent Solutions (Part 2)

Building on the previous module, this section delves into advanced reasoning techniques, tool orchestration, and human-AI collaboration. Participants learn to handle ambiguity, refine conversational interactions, and design agents capable of complex analysis. The module emphasizes intuitive user interfaces and decision-support mechanisms. Example projects include a legal assistant, a financial advisor, and a creative writing support agent.

Agent-Based Retrieval-Augmented Generation (RAG)

This module covers the application of Retrieval-Augmented Generation (RAG) in AI agents. Participants explore dynamic retrieval strategies, optimize vector databases, and implement iterative information retrieval methods. By the end of the module, students will be able to design RAG systems tailored for conversational AI and break down complex queries into manageable components. Practical applications include developing knowledge bases, technical support systems, and AI-driven educational assistants.

Multi-Agent Systems (Part 1)

The first part of this module focuses on designing multi-agent architectures, defining specialized agent roles, and establishing communication protocols. Participants learn to coordinate workflows and implement group decision-making processes. The module provides hands-on experience in developing collaborative agents for applications such as business analysis and content management.

Multi-Agent Systems (Part 2)

This advanced module explores complex collaboration patterns, hierarchical agent structures, and conflict resolution strategies. Special attention is given to ethical considerations and security mechanisms in multi-agent environments. Students design and implement systems with structured agent hierarchies and negotiation protocols. Example projects include supply chain management solutions and AI-driven creative teams with specialized roles.

Evaluation of AI Agents

Participants learn techniques for assessing AI agent performance, establishing benchmarking standards, and conducting quality assurance tests. The module introduces both automated evaluation tools and human-driven testing frameworks. Students develop systems for monitoring agent quality, building test infrastructures, and creating benchmarking methodologies. Example projects include customer service agent evaluation frameworks and adversarial testing systems.

AI Agent Monitoring & Optimization

This module focuses on designing monitoring systems for AI agents, tracking performance metrics, and optimizing resource usage. Participants implement intelligent error detection mechanisms, collect user feedback, and refine operational efficiency. Example projects include developing advanced monitoring dashboards and anomaly detection systems for AI agents.

Deployment & Production with MCP Applications

The final module prepares participants to deploy AI agents in real-world production environments, with a special focus on the Model Context Protocol (MCP). Students learn best practices for ensuring scalability, version control, and secure access. By the end of this module, they will be able to design robust production architectures and deploy AI agents with high reliability. Example applications include enterprise-grade agent platforms and multi-agent systems with automated deployment and version management.