

Building Agentic AI Systems: LangGraph and Model Context Protocol

Duration: 4 Days (32 Hours)

Course Overview

This comprehensive 32-hour course equips learners with the skills to design, build, and deploy intelligent agentic AI systems using two of the most powerful frameworks in the modern AI landscape — LangGraph and Model Context Protocol (MCP). Participants will begin by mastering LangGraph's core components, building AI agents from scratch, integrating agentic search tools, enabling persistence and streaming, and incorporating human-in-the-loop workflows. The course then transitions to MCP, covering its architecture, primitives, protocol layer, transport mechanisms, server and client development, and cloud deployment with security best practices. Together, these two modules provide a complete end-to-end foundation for building production-grade, context-aware, and interoperable AI agent applications.

Pre-requisites

- Basic understanding of AI and Machine Learning
- Familiarity with Python and the LangChain framework
- Working knowledge of Large Language Models (LLMs)
- Python 3.13 or above installed on your system

Tools & Platforms

- Python 3.13 or above
- Visual Studio Code (VS Code)
- Claude Desktop App

Course Agenda

PART 1: Create AI Agents Using LangGraph

Module 01: Introduction to LangGraph

- Overview of LangGraph and its significance in AI
- How LangGraph enables structured AI workflows
- When to use LangGraph for AI agent development

Module 02: Build an AI Agent from Scratch

- Core concepts of AI agent development
- Step-by-step process of creating an AI agent
- Hands-on: Implementing a basic AI agent in LangGraph

Module 03: LangGraph Components

- Understanding LangGraph nodes and edges
- Designing modular AI agents with reusable components
- Hands-on: Building and testing different LangGraph components

Module 04: Agentic Search Tools

- What are agentic search tools?
- Implementing intelligent search capabilities in AI agents
- Hands-on: Creating an AI-powered search assistant

Module 05: Persistence and Streaming

- Enabling memory and long-term knowledge retention in AI agents
- Implementing streaming-based interactions for real-time data processing
- Hands-on: Building an AI agent with persistence and streaming

Module 06: Human in the Loop (HITL) AI Agents

- Why human oversight is essential in AI workflows
- Methods for integrating human feedback into AI agents
- Hands-on: Implementing human-in-the-loop decision-making

Module 07: AI-Powered Essay Writer

- Designing AI agents for content generation
- Optimizing LLMs for structured writing tasks
- Hands-on: Building an AI essay writer using LangGraph

Module 08: LangChain Resources & Best Practices

- Exploring LangChain tools and libraries for AI agents
- Optimizing AI workflows with pre-built LangChain utilities
- Hands-on: Using LangChain with LangGraph for enhanced AI applications

PART 2: Mastering Model Context Protocol for Agentic AI

Module 09: MCP Foundations and Ecosystem

- Context management in modern LLM systems
- Fragmentation across AI tools and platforms
- Protocol-based interoperability for AI systems
- MCP positioning within the AI ecosystem
- Open standard philosophy and ecosystem growth

Module 10: MCP Architecture and Roles

- End-to-end MCP interaction model
- Host, Client, and Server responsibilities
- Client–Server decoupling model

Module 11: MCP Primitives and Capability Model

- Tools as executable operations
- Resources as structured data interfaces
- Prompts as reusable instruction artifacts
- Capability exposure and discovery
- Capability negotiation and constraints

Module 12: MCP Protocol and Data Layer (JSON-RPC 2.0)

- RPC-based communication paradigm
- JSON-RPC request and response structure
- Notifications and batching
- Error objects and standardized error codes

Module 13: MCP Transport and Lifecycle

- Transport abstraction in MCP
- Local communication using STDIO
- Remote communication using HTTP
- Server-Sent Events (SSE) for streaming responses
- Initialization handshake
- Version and capability negotiation

Module 14: MCP Server Development (Standard & FAST MCP)

- MCP server responsibilities
- Standard MCP server implementation
- FAST MCP architecture and philosophy
- Performance-oriented server design
- Tool, resource, and prompt registration

Module 15: MCP Client Integration and Execution Flow

- MCP client responsibilities
- Request orchestration and tool execution
- Integration with existing AI clients
- Connector-based integrations

Module 16: Cloud Deployment, Security, and Production Patterns

- Cloud-hosted MCP server architecture
- MCP deployment on Azure
- Authentication and authorization strategies
- QnA and Quiz