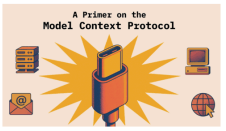
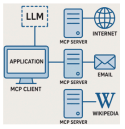
Model Context Protocol

Duration: 2 days Price: € 1699

Open Schedule





Model Context Protocol

Audience Course Model Context Protocol

This course is intended for AI developers, and software architects who want to build AI applications that connect to external data sources and tools using the Model Context Protocol.

Prerequisites Course Model Context Protocol

Participants should have solid programming experience in Python or TypeScript and understanding of API development.

Realization Training Model Context Protocol

The course combines theoretical sessions with hands-on labs guided by a trainer. Real-world case studies are central to the training experience.

Model Context Protocol Certificate

After completion, participants receive a certificate of participation in Model Context Protocol.

Content Model Context Protocol

The course Model Context Protocol from SpiralTrain teaches you how to build AI applications that seamlessly connect to external data sources and tools using the standardized Model Context Protocol. You will learn how to implement MCP servers and clients, integrate contextual information into AI workflows, and create extensible applications that leverage real-world data effectively.

MCP Fundamentals

The course Model Context Protocol starts with an overview of the protocol including the AI context problem, JSON-RPC foundation, transport mechanisms, resource concepts, tool definitions, and real-world use case examples.

Protocol Architecture

Next the architecture patterns are explored, covering communication flow, message types, capability negotiation, session management, error handling, connection lifecycle, protocol extensions, and versioning strategies for robust implementations.

Server Implementation

This module covers building MCP servers including SDK installation, implementing resource providers, developing and registering tools, context management, data source integration, authentication methods, configuration options, and comprehensive testing approaches.

Client Development

Here participants learn to build MCP clients including SDK integration, connection management, server discovery, resource access patterns, tool invocation, prompt usage, response handling, state management, and error recovery mechanisms.

Integration Patterns

This part focuses on integrating MCP with LLMs such as Claude and OpenAI. Topics include context injection, multi-server setups, fallback strategies, caching mechanisms, rate limiting, and monitoring solutions.

Production Deployment

The course concludes with production deployment strategies including containerization, scaling solutions, load balancing, security hardening, API gateway setup, version management, documentation standards, testing strategies, monitoring tools, and hands-on production projects.

Modules Model Context Protocol

Module 1: MCP Fundamentals	Module 2: Protocol Architecture	Module 3: Server Implementation
MCP Protocol Overview	Architecture Patterns	Server Setup
AI Context Problem	Communication Flow	SDK Installation
Protocol Specifications	Message Types	Resource Providers
JSON-RPC Foundation	Request-Response Cycle	Implementing Resources
Transport Mechanisms	Capability Negotiation	Tool Development
Client-Server Model	Session Management	Tool Registration
Resource Concepts	Error Handling	Context Management
Tool Definitions	Connection Lifecycle	Data Source Integration
Prompt Templates	Protocol Extensions	Authentication Methods
Use Case Examples	Versioning Strategy	Configuration Options
Security Basics	Best Practices	Testing Servers
Module 4: Client Development	**Module 5: Integration Patterns**	**Module 6: Production Deployment**
Client Setup	LLM Integration	Deployment Strategies
SDK Integration	Claude Integration	Containerization
Connection Management	OpenAI Integration	Scaling Solutions
Server Discovery	Context Injection	Load Balancing
Resource Access	Multi-Server Setup	Security Hardening
Tool Invocation	Fallback Strategies	API Gateway Setup
Prompt Usage	Caching Mechanisms	Version Management
Response Handling	Rate Limiting	Documentation Standards
State Management	Monitoring Solutions	Testing Strategies
Error Recovery	Logging Practices	Monitoring Tools
Performance Optimization	Real-World Examples	Production Projects