AI Agent Development and Usage Training

Eğitim Açıklaması

Agent Design, Frameworks, and No-Code Automation – 2-Day Program

The objective of this training is to enable participants to understand the concept of AI agents at both theoretical and practical levels, and to develop agent-based systems using both code-based approaches and no-code/low-code tools.

By the end of the training, participants will be able to design agent architectures, build automation workflows using tools such as n8n, develop agents with frameworks like LangChain, CrewAI, and Agno, and create enterprise-level agent use cases.

Training Outcomes

- Technical understanding of AI agent concepts
- Ability to develop agents using both no-code and code-based approaches
- Skills to design enterprise-grade agent architectures
- Ability to manage agent security risks
- Capability to strategically identify agent use cases

Eğitim İçeriği Nedir?

Day 1 – AI Agent Fundamentals and No-Code Automations

What is an AI Agent?

- Difference between agents and LLMs
- Reactive vs autonomous agents
- Concept of tool-using agents
- Memory and planning mechanisms
- Agent lifecycle

Agent Architecture

- Input
- Planning
- Tool Selection
- Execution
- Memory Update
- Feedback Loop

No-Code Agent Systems

- n8n fundamentals and workflow structure
- Trigger → Action → Tool chain
- Connecting LLMs via webhooks
- API integrations
- Zapier vs n8n comparison

Hands-on – n8n Agent Workflow

- Analyzing user requests
- Interpreting with LLM
- Making API calls
- Generating reports
- Sending outputs via Slack or email

Day 2 – Agent Frameworks and Code-Based Development

Introduction to Agent Frameworks

- LangChain – agent executor, tool binding, memory, chains
- CrewAI – role-based multi-agent structure, planner/worker model
- Agno – lightweight agent approach
- Upsonic – enterprise agent architecture and lifecycle management

Agent Development Techniques

- Plan-first agent approach
- Multi-agent coordination
- Tool abstraction
- Memory persistence
- RAG integration
- MCP integration

Agent Security

- Prompt injection risks
- Tool misuse and permission scoping
- Output validation
- Token and cost control

Enterprise Agent Architecture

- Central orchestrator
- Domain-based agent structures
- Monitoring and logging
- Audit and governance model

Final Hands-on Project

- Designing a simple agent
- Integrating tools
- Implementing memory
- Building a multi-agent scenario
- Applying security controls