Faysal Aziz

AI-Powered Development

Building Intelligent Microservices for Real-World Systems - 3 days

FA-0089 · 3 days

Agentic & Generative AI · Software Development

Introduction

Software development is undergoing a structural shift. AI is no longer a peripheral tool; it is becoming a core layer in system design, shaping how applications reason, automate, and scale. Developers are now expected to move beyond simply consuming APIs and instead architect systems where AI participates as a decision-making component.

This course is designed to meet that expectation. It bridges foundational AI understanding with practical system integration, focusing on microservices, real-time pipelines, and agentic automation. Participants will work across Angular, ASP.NET, Python (FastAPI), Kafka, and production-based environments, learning how to embed AI into production-grade systems.

The instructor brings over 30 years of industry experience and emphasizes real-world, industry-demanded practices rather than academic abstractions.

Learning outcomes

By the end of this course, participants will be able to:

- Understand the full AI landscape, including machine learning, LLMs, and agentic systems
- Identify opportunities to enhance developer productivity using AI tools and workflows
- Design and implement AI-enabled microservices using Python, Java, and C#
- Integrate AI endpoints into enterprise systems for data analysis and decision-making
- Build and orchestrate agentic automation systems capable of processing work orders and documents
- Deploy AI solutions both via cloud APIs and on-premise AI engines
- Use Kafka for event-driven AI workflows and scalable data pipelines

- Architect secure, maintainable, and scalable AI-powered systems in Linux environments

---

Prerequisites

- Intermediate programming experience in at least one of: Python, Java, or C#
- Basic understanding of web development concepts (REST APIs, HTTP, JSON)
- Familiarity with microservices architecture
- Exposure to Angular or similar frontend frameworks
- Basic knowledge of Linux command line and environment setup
- Understanding of databases and asynchronous processing is beneficial

---

Training outline

1. Foundations of Artificial Intelligence

1. Evolution of AI systems from rule-based to generative models

   Key domains within AI

   1. Machine Learning and Deep Learning
   2. Natural Language Processing
   3. Computer Vision
   4. Generative AI and Large Language Models
3. Understanding model behavior and limitations

   AI system components

   1. Data pipelines
   2. Model training vs inference
   3. API-driven AI consumption

   Overview of modern AI ecosystems

   1. Cloud AI providers
   2. Open-source and on-premise AI engines

2. AI for Developer Productivity

AI-assisted coding workflows

1. Code generation and completion
2. Debugging and refactoring assistance

Designing AI-enhanced development pipelines

1. CI/CD augmentation with AI
2. Automated documentation generation

Prompt engineering for developers

1. Structuring prompts for code and system design
2. Controlling output consistency and reliability

Integrating AI into daily developer tools

1. IDE integrations
2. CLI-based AI workflows

5. Evaluating productivity gains and risks

3. Microservices Architecture for AI Systems

1. Principles of microservices in AI-driven systems
2. Service decomposition strategies
3. API-first design for AI services
4. Stateless vs stateful AI services
5. Containerization and deployment considerations on Linux

Inter-service communication patterns

1. REST
2. Event-driven architecture

4. Building AI Microservices with Python and FastAPI

1. Designing RESTful AI endpoints
2. Structuring FastAPI applications for scalability
3. Handling inference requests and responses
4. Integrating pre-trained models and external AI APIs
5. Managing concurrency and performance
6. Logging, monitoring, and observability

5. AI Integration with ASP.NET and Java Services

1. Creating AI-enabled endpoints in ASP.NET
2. Consuming AI APIs in C# applications
3. Java-based microservices for AI orchestration
4. Cross-language service communication

5. Security considerations in AI API consumption
6. Authentication and authorization for AI services

6. Frontend Integration with JS Framework

1. Designing UI for AI-powered applications
2. Calling backend AI services from JS based systems
3. Handling asynchronous AI responses
4. Visualizing AI-generated insights
5. Managing user interaction with AI systems

7. Data Analysis Using AI Endpoints

1. Structuring data for AI processing
2. Sending structured and unstructured data to AI services
3. Interpreting AI outputs for analytics
4. Building pipelines for continuous data analysis
5. Handling large datasets and streaming data
6. Integrating AI insights into business workflows

8. Event-Driven AI Systems with Kafka

1. Introduction to Kafka in AI architectures
2. Designing event streams for AI processing
3. Producer and consumer patterns
4. Real-time AI inference pipelines
5. Scaling AI workloads using Kafka
6. Fault tolerance and message reliability

9. Agentic Automation and Intelligent Workflows

Understanding agentic AI systems

1. Autonomous decision-making
2. Task decomposition and execution

2. Designing agents for enterprise workflows

Processing work orders and documents

1. Text extraction and parsing

    2. Context-aware decision making

   Multi-step AI workflows

    1. Chaining AI services
    2. Feedback loops and state management
5. Human-in-the-loop systems
6. Monitoring and controlling agent behavior

10. On-Premise AI Deployment

1. Benefits and challenges of on-prem AI
2. Setting up local AI inference engines
3. Hardware and infrastructure considerations
4. Model hosting and optimization
5. Data privacy and compliance
6. Hybrid architectures combining cloud and on-prem AI

11. System Design and Capstone Architecture

1. Designing a full AI-powered microservices system
2. Combining Angular, ASP.NET, FastAPI, Kafka
3. End-to-end data flow design
4. Scalability and performance considerations
5. Security and governance in AI systems
6. Deployment strategies on Linux environments
7. Observability, logging, and maintenance

This course is structured to ensure developers not only understand AI conceptually but can also build, integrate, and deploy AI-powered systems that reflect real-world enterprise demands. The emphasis throughout is on practical architecture, production readiness, and leveraging AI as a core engineering capability rather than an experimental add-on.

Disclaimer

This course outline is provided strictly as a general guideline for the anticipated scope, sequence, and coverage of the training program. While every reasonable effort will be made to deliver the course substantially in accordance with this outline, the trainer reserves the sole and absolute discretion to amend, modify, reorder, expand, reduce, substitute, or otherwise vary any topic, module, exercise, tool, or area of coverage as may be required by participant needs, time constraints, technical considerations, class

progress, operational circumstances, or any other relevant training-related factors. Such changes may be made at any time, with or without prior notice, where deemed necessary or appropriate by the trainer in order to maintain the quality, relevance, and effectiveness of the course delivery.

© 2026 Faysal Aziz · Kuala Lumpur