Faysal Aziz

Modernizing .NET Development with AI

Leverage AI-driven coding, Copilot, and Python–C# integration - 2 days

FA-0325 · 2 days

Agentic & Generative AI · Software Development · DevOps, Cloud & Infrastructure · Automation, RPA & Power Platform

---

Introduction

The .NET world isn't standing still. While enterprise teams push .NET 8 into Linux production environments, smart developers are discovering how AI can transform their entire workflow—from initial design to final deployment.

This course cuts through the noise to show you what actually works. You'll harness GitHub Copilot's full potential, leverage ChatGPT, Gemini, and Claude as development partners, and seamlessly blend Python capabilities into your C# codebase. We'll build intelligent workflows that think and adapt, using proven prompt engineering techniques that deliver results.

No theoretical frameworks or overcomplicated toolchains. Just practical, immediately applicable skills for .NET developers ready to work smarter, not harder.

Learning outcomes

By the end of this intensive 2-day course, participants will be able to:

- Prompt-engineer effectively in C# to maximize AI assistant output
- Use and compare ChatGPT, Gemini, Claude, and Manus for .NET coding
- Seamlessly integrate Python components into .NET 8 apps
- Set up and use AI-assisted tools like GitHub Copilot within .NET environments
- Automate parts of the coding workflow via agentic Python tools with C# orchestration

- Implement backend API integrations and deploy AI-driven .NET services on Linux

---

Prerequisites

- Professional C# experience with .NET Core / .NET 8
- Comfort working on Linux (CLI-based build & deployment)
- Basic Python knowledge (or willingness to dive in quickly)
- Familiarity with REST APIs and Git

Training outline

1. 1. Professional .NET 8 Setup on Linux

- Install .NET 8 SDK/Runtime on Linux
- Configure VS Code or Rider with C# tooling, IntelliCode, GitHub Copilot
- Set up project templates, Docker environment, and CI pipelines

2. 2. Prompt Engineering for C\

- C#-specific prompt patterns: doc comments, method stubs, LINQ operations
- Using AI to scaffold classes, generate unit tests, and fix errors
- Hands-on: craft prompts in VS Code, evaluate ChatGPT vs Copilot vs Tabnine

3. 3. Model Comparison: ChatGPT, Gemini, Claude, Manus

Overview of strengths:

  - ChatGPT: flexible & versatile
  - Gemini: large context handling
  - Claude: high accuracy for enterprise code
  - Manus: agentic execution capabilities
- Live demos: generate a .NET utility, write tests, compare correctness, readability, and integration

4. 4. GitHub Copilot and IDE Integration

- Install & configure Copilot with .NET 8 projects
- Use Copilot for code completion, refactoring, test generation
- Best practices for review and version control with AI-generated code

5. 5. Python in C#: Integration Patterns

- Overview of **IronPython** and .NET interop
- Embedding Python engines into C# apps for scripting and AI tasks
- Use cases: calling LLM-based Python modules from within .NET
- Hands-on lab: create a C# console app that invokes a Python module to handle AI-driven text or data processing

6. 6. Agentic Automation: Task Decomposition in .NET + Python

- Building agentic workflows with Python (e.g., Manus-like scripts)—writing schedulers and task orchestrators
- Expose endpoints in C# to invoke Python agents for background tasks or pre-commit checks
- Hands-on: automate a multi-step development task, like generating code, testing, and packaging via agentic Python routines

7. 7. Backend Integration & Deployment

- Setup REST APIs in ASP.NET Core invoking both C# and Python modules
- Use prompt-engineered snippets for API logic
- Deploy full-stack app on Linux using Docker and .NET 8
- Testing and monitoring strategies for AI-generated code

8. 8. Capstone Demo & Best Practices

- Develop a mini AI-enhanced .NET service: e.g., GitHub webhook that triggers Python agents to run code analysis
- Compare approaches using different LLMs
- Wrap-up discussion on security, code ownership, licensing, and productivity trade-offs

---

Instructor Profile

Your instructor brings **30+ years** of production-grade expertise in .NET development, systems programming, and AI integrations—offering highly relevant, real-world training far beyond academic theory.

© 2026 Faysal Aziz · Kuala Lumpur