

AI for Software Developers - 40 hours

Course description

The future of software engineering is here — and it's powered by Artificial Intelligence. This course is designed for developers who want to understand, adapt, and thrive in the new era where human creativity and machine intelligence work hand in hand.

Participants explore how AI is reshaping every aspect of development — from coding and debugging to design, testing, and deployment. They gain hands-on experience with today's most powerful tools, including GitHub Copilot, Gemini, Cursor, Claude, and Continue.dev, and learn how to integrate them seamlessly into their daily workflow.

Through practical projects, students master AI-assisted programming, writing effective prompts, generating and reviewing code, and building real applications together with AI — all while maintaining control, context, and quality. They also learn about “vibe coding”, an emerging paradigm where developers communicate with AI through natural language to build complete software systems faster and smarter.

The program goes far beyond simple code generation: participants discover how AI transforms the entire software development lifecycle (SDLC) — from writing user stories and designing system architectures, to automating testing, DevOps, and maintenance. They also acquire the critical skills to secure, audit, and govern AI-generated code, ensuring reliability and compliance.

In the final stages, learners experiment with AI agents and workflow automation, using frameworks like LangChain, LlamaIndex, CrewAI, and Agent Builder to create intelligent systems that plan, refactor, test, and deploy code autonomously.

By the end of the course, graduates emerge as AI-empowered developers — professionals who know how to build faster, think smarter, and lead innovation in a world where coding is no longer just human.

Course Contents:

- **Module 1: The AI-Augmented Developer - Mindset and Ecosystem**
 - How AI is reshaping software development: From IDE helpers to autonomous code agents.
 - Overview of the AI dev landscape (Copilot, Gemini, Cursor, Claude, Continue.dev).
 - AI coding assistants vs. AI pair programmers vs. AI code agents
 - Realistic expectations - what AI can and cannot do (yet).
 - Ethics, privacy, and IP considerations in AI-assisted coding.
- **Module 2: AI-Assisted Development in Practice**
 - Integrating AI tools into your daily workflow (IDE plugins, terminals, documentation, tests).
 - Prompting for developers — writing effective technical prompts.
 - Pair-programming with AI: debugging, refactoring, generating unit tests, code comments, documentation, and API clients.
 - Best practices: review discipline, context awareness, data leakage prevention.
 - Demo: Building a small REST API using AI assistance only.
 - Hands-On:
 1. Build and test a CRUD app.
 2. Use AI to generate and fix unit tests.
 3. Prompt AI to explain legacy code and generate documentation.
- **Module 3: “Vibe Coding” — Natural-Language Driven Development**
 - What is vibe coding and why it’s emerging.
 - From Copilot to Devin-style workflows — conversational coding agents.
 - Setting up a local vibe coding environment (e.g., Continue.dev, OpenHands).
 - When to use vibe coding vs. when to stick to traditional dev workflows.
 - Managing the “AI handoff problem”: maintaining generated code, version control, and re-generation strategies.
 - Pitfalls: hallucinated logic, security flaws, incomplete edge-case handling.
 - Hands-On: Use a vibe coding tool to build a simple end-to-end project (e.g., a task tracker). Then, manually review and extend the generated code — showing the “beyond the AI” principle.
- **Module 4: The AI-Enhanced Product Lifecycle**
 - How AI transforms each stage of the SDLC:
 - Requirements: AI for user story drafting, acceptance criteria generation, backlog refinement.

- Design: AI-generated architecture diagrams, sequence diagrams, and UX mockups.
- Development: collaborative coding with AI.
- Testing: AI-assisted test plan creation, test data generation, QA automation.
- DevOps: AI in CI/CD pipelines, monitoring, and deployment scripting.
- Maintenance: AI for bug triage, ticket summarization, changelog generation.
- Module 5: Security & Quality Code in AI-Generated Code
 - Common vulnerabilities in AI-generated code (injection, missing validation, hardcoded secrets).
 - How to perform security audits and static analysis.
 - Tools
 - Demo (optional):
 - Use AI to generate a login API, then analyze vulnerabilities manually and with a scanner.
 - Use AI again to fix security issues — discuss what it missed.
- Module 6: Advanced Workflows & Automation with AI Agents
 - Beyond code generation: agents that can plan, test, refactor, and deploy.
 - Using AI to orchestrate workflows (LangChain, LlamaIndex, CrewAI, Agent Builder).
 - Building internal AI developer tools and assistants.
 - Connecting AI tools to issue trackers (Jira, GitHub), docs, databases, etc.
 - Integrating AI with CI/CD pipelines.
 - Demo: building an AI Agent workflow
- Module 7: Emerging Trends & The Road Ahead
 - From copilots to autonomous dev agents — how fast is it moving?
 - Evolution of local LLMs, open-source models, and on-device dev tools.
 - Future of IDEs — integrated multimodal assistants (voice, sketch-to-code, screen understanding).
 - AI and software architecture evolution.
 - Impact on developer roles, hiring, and education.
- Module 8: Final Project

Participants work in small teams to:

 - Pick a small real-world software feature or product concept.
 - Use AI tools across the lifecycle: requirements → design → implementation → test → doc → deployment (optional).
 - Present outcomes, what worked, what didn't, and lessons learned (optional).