Curso ATDEV240 - Programação Agêntica com GitHub Copilot

Duração: 21h

Introdução

Esta formação capacita os programadores para tirarem partido do GitHub Copilot e das técnicas de programação agêntica, entregando software mais depressa e com mais eficácia. Os participantes aprendem a colaborar com agentes de IA ao longo de todo o ciclo de vida do software, transformando requisitos de alto nível em soluções a funcionar, sem perder o controlo sobre qualidade, segurança e arquitetura. Ao dominar a programação agêntica, o programador trabalha de forma mais produtiva e passa mais tempo em análise, desenho e decisão, em vez de tarefas repetitivas de escrita de código.

Público-alvo

Este curso destina-se a programadores que queiram aumentar a sua produtividade com programação agêntica através do GitHub Copilot.

No final do curso

Após a conclusão deste curso, os formandos deverão ser capazes de:
• Utilizar o GitHub Copilot ao longo de todo o ciclo de vida do software.
• Integrar o GitHub Copilot com MCP e com ferramentas de linha de comandos.
• Configurar skills, agentes e hooks.
• Delegar tarefas no GitHub Copilot Cloud Agent.
• Rever e verificar código gerado por IA.
• Construir aplicações à medida com o GitHub Copilot SDK.

Pré-requisitos

Experiência de programação. Todos os laboratórios e demonstrações assentam em código C#, mas os princípios abordados aplicam-se a qualquer linguagem de programação.

Exames de certificação

Contribui para a preparação do exame GH-300: GitHub Copilot, da certificação GitHub Copilot, da GitHub (exame não incluído neste curso).

Conteúdo programático em detalhe

O papel em mudança do programador
• De quem escreve código a analista, arquiteto de agentes e revisor
• O que é a programação agêntica?
• O ciclo agêntico: raciocinar, usar ferramentas, observar, iterar
• O panorama das ferramentas de programação agêntica: GitHub Copilot, Claude Code, OpenAI Codex e Cursor
Introdução ao GitHub Copilot e aos LLM
• Compreender os LLM
• Conhecimento estático vs. conhecimento dinâmico de um LLM

- Licenças do GitHub Copilot
- GitHub Copilot CLI vs. Visual Studio (Code)
- Escolha do modelo: qual o melhor para cada trabalho
- Compreender as questões de privacidade e segurança
- LAB: ativar a licença gratuita do GitHub Copilot

Fundamentos do GitHub Copilot

- Conclusões de código e sugestões da edição seguinte
- Copilot Chat: em linha vs. em painel
- Modos do agente: Ask, Edit, Agent e Plan
- Tipos de agente do Copilot
- Geração de mensagens de commit e de descrições de pull request
- Compreender as ferramentas do GitHub Copilot
- LAB: refatorar e documentar código legado com as funcionalidades normais do chat

IA ao longo do ciclo de vida do software: análise e desenho

- Extrair e clarificar requisitos com IA
- Especificações de prompt engineering
- Desenho de APIs e de arquitetura com o Copilot
- Architecture Decision Records (ADR) como contexto orientador
- Gerir issues e itens de trabalho
- LAB: desenhar a plataforma de comércio eletrónico — especificações, API e arquitetura

Integrações com MCP e ferramentas de linha de comandos

- Integrações universais de IA com o Model Context Protocol (MCP)
- Servidores MCP úteis
- Considerações de segurança dos servidores MCP
- Eficiência de contexto: MCP vs. ferramentas de linha de comandos
- Dar ao Copilot ferramentas de linha de comandos
- LAB: criar uma interface moderna com o Copilot e a CLI do Playwright

Skills, agentes e hooks

- Custom Instructions
- Definir agentes especializados
- Definir capacidades e scripts reutilizáveis com skills
- Automatizar a limpeza de código gerado por IA com hooks
- Quando usar instructions, skills ou agentes?
- LAB: construir instructions, skills e agentes próprios

Alargar a equipa de desenvolvimento com o GitHub Copilot Cloud Agent

- Introdução ao Cloud Agent
- Definir requisitos de funcionalidades com issues do GitHub
- Atribuir ao Copilot a revisão de um pull request
- Orientar o Copilot
- Rever alterações com o GitHub Codespaces
- GitHub Agentic Workflows: definir fluxos autónomos de IA
- Desencadear agentic workflows a partir de issues, comentários e agendamentos
- LAB: juntar o GitHub Copilot à equipa de desenvolvimento do seu projeto

IA ao longo do ciclo de vida do software: implementação e testes

- Usar as suites de testes como guardrails do agente
- O ciclo de test-driven development com o Copilot
- Análise estática, linters e analisadores como sinais de feedback
- Hooks para orientação determinística do agente
- Verificação ponta a ponta da interface com a CLI do Playwright
- LAB: construir e testar a plataforma de comércio eletrónico com o Copilot

IA ao longo do ciclo de vida do software: rever e verificar código gerado por IA

- Rever com eficiência grandes diffs produzidos por IA
- Análise estática e verificação de segurança como sinais objetivos de revisão
- Verificar o comportamento com testes, Playwright e ferramentas de benchmarking
- Orientar o agente para produzir alterações incrementais e revisíveis
- LAB: rever, verificar e reforçar a plataforma de comércio eletrónico

Construir aplicações à medida com o GitHub Copilot SDK

- Introdução ao GitHub Copilot SDK
- O núcleo agêntico e o ciclo de execução do Copilot
- Montar o SDK e autenticar
- Integrar ferramentas próprias e servidores MCP por código
- Construir interfaces próprias e agentes para tarefas específicas
- LAB: construir uma ferramenta de automatização própria com o Copilot SDK