

Updated on 09/04/2026

Sign Up

GitHub Agentic AI Developer Training (GH-600)

3 days (21 hours)

Overview

The GH-600 Certification training course prepares you for the official GitHub GH-600 certification and enables you to master the design, integration, and governance of AI agents in modern development workflows.

This intermediate-level course helps you prepare for the GitHub GH-600 exam, designed for professionals who deploy, monitor, and operate AI agents within a GitHub-driven production SDLC.

You'll learn how to design an agent-based architecture, distinguish between planning, reasoning, and execution, and then configure the tools, permissions, MCP servers, and execution environments appropriate for each use case.

The training also covers memory and state management, result evaluation, error analysis, instruction tuning, and the orchestration of multiple agents. You will apply these concepts to scenarios that reflect the skills assessed in the exam.

By the end of the training, you will be able to effectively prepare for the GitHub GH-600 certification, ensure agent autonomy with safeguards, maintain traceability of actions, and implement human validations proportionate to the risks.

Like all our training courses, this one will introduce you to **the latest stable version** of the technology and its new features.

Objectives

- Understand the skill areas assessed by the GitHub GH-600 certification.

- Design AI agent architectures integrated into the SDLC.
- Configure the tools, permissions, and MCP servers used by agents.
- Evaluate, diagnose, and adjust an agent's behavior based on actionable artifacts.
- Implement governance, security, and human oversight mechanisms.

Target Audience

- AI Engineer
- Software Developer
- DevOps Engineer
- Platform Engineer
- Solutions Architect

Requirements

- Hands-on experience with the software development lifecycle and code reviews.
- Proficiency with Git and GitHub.
- Basic knowledge of GitHub Actions workflows.
- Basic familiarity with GitHub Copilot or AI-powered development assistance tools.

Technical Prerequisites

- A computer running Windows 11, a recent version of macOS, or a recent version of Linux, with 16 GB of RAM recommended.
- A recent version of Git, GitHub CLI, and Visual Studio Code must be installed.
- A GitHub account with access to GitHub Copilot to complete the workshops.
- A stable internet connection without any filtering that blocks GitHub services, Visual Studio Code extensions, or MCP services.

Curriculum for our GH-600 – GitHub Certified Agentic AI Developer training course

[Day 1 – Morning]

Agentic Fundamentals and Exam Overview

- Overview of the GitHub GH-600 certification, its positioning, and the skill areas assessed.
- Role of AI agents in a production SDLC, responsibilities, and interactions with
- Distinguish between an assistant, an agent, an autonomous workflow, and deterministic automation.
- Define the inputs, outputs, success criteria, and operational limits of an agent-based task.
- Identify anti-patterns, risks of misuse, and situations requiring supervision.

[Day 1 - Afternoon]

Agent Architecture and Integration into the SDLC

- Structure an agent-based architecture around the planning, reasoning, and execution phases.
- Configure explicit instructions, structured plans, and validation criteria prior to action.
- Position agents within branch flows, issues, pull requests, and continuous integration pipelines.
- Determine the appropriate level of autonomy based on impact, reversibility, and business risk.
- Hands-on workshop: Design a pull request review agent with an action plan, success criteria, and human validation.

[Day 2 - Morning]

Tools, MCPs, and Runtime Environments

- Select the tools required for an agent and apply the principle of least privilege.
- Understand the role of MCP servers in accessing external resources, services, and data.
- Configure permissions, authorization lists, and scopes of use for agent tools.
- Choose an execution context, a deployment scope, and an appropriate branching strategy.
- Set up secure execution paths with error handling, recovery, rollback, and escalation.
- Hands-on workshop: Configure an agent equipped with an MCP server, limited permissions, and a controlled deployment scope.

[Day 2 - Afternoon]

Memory, State, and Execution Continuity

- Distinguish between short-term memory, long-term memory, and external memory based on the nature of the task.
- Define rules for memory scope, expiration, cleanup, and reset.
- Preserve progress, decisions, and results as persistent artifacts.
- Prevent obsolete contexts, conflicting information, and drift during long-running executions.
- Resume agent-based work across tools and environments without duplication or divergence.
- Hands-on workshop: Implement an execution resumption scenario with persistent state and context drift control.

[Day 3 - Morning]

Agent Evaluation, Diagnosis, and Optimization

- Define qualitative and quantitative evaluation metrics aligned with the development intent.

- Use logs, traces, plans, outputs, and workflow artifacts to analyze the results.
- Identify root causes related to reasoning, tools, context, or the environment.
- Adjust instructions, constraints, tools, permissions, and memory strategies.
- Use code analysis and automated scans as indicators of quality and security.
- Hands-on workshop: Diagnose an agent failure based on traces and propose a measurable tuning plan.

[Day 3 - Afternoon]

Orchestration, Safeguards, and Final Preparation

- Apply orchestration models to coordinate multiple agents sequentially or in parallel.
- Isolate agents, organize data transfers, and resolve conflicts related to changes or decisions.
- Ensure the observability of multi-agent workflows with auditable logs, artifacts, decisions, and results.
- Implement guardrails, explicit approvals, and human-in-the-loop mechanisms.
- Review the six exam domains, analyze certification scenarios, and optimize your response strategy.
- Hands-on workshop: conduct a certification case study on a secure multi-agent workflow, then justify the architectural and governance choices.

Target Audience

This training is intended for both individuals and companies—large or small—seeking to train their teams in new, advanced IT technologies or to acquire specific domain knowledge or modern methodologies.

Entry-Level Assessment

The pre-training assessment complies with Qualiopi quality standards. Upon final registration, the learner receives a self-assessment questionnaire that allows us to evaluate their estimated proficiency in various types of technologies, as well as their expectations and personal goals for the upcoming training, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could pose challenges for monitoring and ensuring the smooth running of the training session.

Teaching Methods

Hands-On Training: 60% Practical, 40% Theoretical. Training materials distributed in digital format to all participants.

Organization

The course alternates between theoretical input from the instructor—supported by examples and reflection sessions—and group work.

Assessment

At the end of the session, a multiple-choice quiz is administered to verify that participants have successfully acquired the skills.

Certification

A certificate will be issued to each trainee who has completed the entire training program.