# Persistent Daemons: Building Continuous AI Agents with Hermes

- 1 Day Course
- Hands-on

## Course Overview

This hands-on workshop dives into building persistent daemon agents using Hermes Agent. Participants learn how to deploy open-source, self-improving background agents designed for continuous execution, cross-session memory accumulation, and real-time messaging integrations. The course focuses on building production-ready long-running processes, using a primary messaging interface (Telegram/Slack) in class alongside boilerplate templates for additional platforms.

## Who Should Attend

- Software Engineers
- Backend Developers
- AI System Engineers

## What You'll Learn

- Understand the architecture of continuous, background persistent AI daemons.
- Implement cross-session memory and automated skill accumulation in Hermes Agent using persistent storage.
- Connect and route agent interactions via webhook and socket interfaces using Python and asyncio.
- Deploy, process-manage, and safeguard continuous self-improving background agents against rate limits and failures.

## Outline

1. Foundations of Persistent Daemons & Hermes Agent

- Introduction to Persistent Daemons: Transitioning from ephemeral request-response pipelines to continuous background execution.
- Hermes Agent Core Architecture: Deconstructing background workers, event triggers, and state loops.
- Setting Up the TUI Interface: Local deployment, terminal controls, and initial daemon configuration.

2. Cross-Session Memory & Skill Accumulation

- Implementing Long-Term State: Configuring persistent vector and key-value memory stores across execution cycles.
- Self-Improving Loops: Mechanizing skill acquisition, feedback processing, and memory indexing.
- Hands-on Lab: Building an autonomous Python agent that retains contextual knowledge across user sessions.

3. Multi-Channel Messaging Integrations

- Connecting External Messaging Platforms: Setting up primary webhook integration (e.g., Telegram or Slack).
- Unified Event Handling: Routing multi-channel inputs and managing concurrent user sessions using boilerplate templates.
- Hands-on Lab: Deploying a Hermes daemon that routes messages and manages real-time interactions.

4. Deployment, Reliability & Autonomous Execution

- Daemonization & Process Management: Running agents reliably as system background services using Systemd and Docker containers.
- Error Recovery & Guardrails: Handling API rate limits, network disruptions, and malformed payloads.
- Capstone Project: Refactoring and daemonizing the multi-channel agent built in Labs 2 and 3 into a self-improving background monitor.

## Prerequisites

- Proficiency in Python (including async/await constructs)
- Basic understanding of REST APIs, Webhooks, and environment configuration
- Local Docker environment and ngrok (or equivalent tunnel) installed
- API keys for primary LLM providers (e.g., OpenAI, Anthropic)

## Next Courses

- ai-agentic-scale-2D: Scaling Daemonized Agent Fleets with Kubernetes
- ai-evals-production-1D: Continuous Evaluation & Guardrails for Autonomous Agents