



BUILD AI MULTI-AGENT WITH CLAUDE CODE

Course ID : VIBE-CODE-L2



เข้าสู่เว็บไซต์



Duration: 2 Days

(12 Hours) 09:00 AM – 04:00 PM



Price: 21,900 THB

* (excluding VAT 7%)

* Eligible for 200% tax deduction



Training Schedule

www.9experttraining.com

Category: Artificial Intelligence

Build production-ready Multi-Agent Systems using Claude Code × OpenCode for cross-tool orchestration — complete with the Failure Modes you must know how to avoid.

Objectives

1. Work with multiple AI Agents simultaneously as a system — without manually controlling each Agent.
2. Build specialized AI Assistants for your own work on Claude Code and OpenCode.
3. Manage and monitor AI Agent progress through Multica.ai with real-time visibility.

Objectives (Continued)

4. Understand and avoid common Multi-Agent failure modes such as infinite loops and file conflicts.
5. Orchestrate AI Agents across tools and roles with confidence.

Target Audience

1. Developers
2. Tech Leads
3. Business Owners & Founders
4. AI Enthusiasts
5. DevOps / SRE Engineers

Prerequisites

1. Basic Terminal / Command Line skills (cd, ls, mkdir)
2. Strong Prompt writing skills
3. Recommended: completion of VIBE-CODE-L1 or equivalent Claude Code experience

System Requirements

1. Windows 11 / 10
2. Claude Account (Free) — set up before training day
3. Claude for Desktop (required for Claude Code)
4. Google Antigravity or VS Code
5. Claude Subscription (9Expert invites participants during class)
6. OpenAI API (9Expert provides account during class)

TOPICS TRAINING



DAY 1 - Morning Session

9:00 AM – 12:00 PM

1. How AI Agents Work as a Team

- Understanding AI Agents and how they differ from Chatbots
- Team operating patterns for AI Agents
- Real-world use cases in business contexts

2. Technical Environment Setup

- Basic Terminal commands (cd, ls, mkdir)
- VS Code configuration: managing Workspaces
- Understanding JSON document structure
- Configuring Environment Variables

3. Claude Code Specialist Agent

- Introduction to Claude Code and its core capabilities
- Building Specialist Agents for specific task domains
- Workshop: Build your first Agent in Claude Code — a Code Reviewer Agent

4. OpenCode Setup & Multi-model

- Introduction to Claude Code and its core capabilities
- Building Specialist Agents for specific task domains
- Workshop: Build your first Agent in Claude Code — a Code Reviewer Agent



DAY 1 - Afternoon Session

1:00 PM – 4:00 PM

5. Permission Boundary

- Define the scope of Agent permissions
- Security best practices for Agent access control

6. Persistent Memory

- Managing Agent memory across sessions
- Context Management techniques
- RAG vs LLM Wiki — choosing the right memory approach
- Build-in Memory vs API Memory

7. From Solo Builder to Multi-Agent System

Take the Web App or Dashboard built in the Vibe Coding course and delegate it to a 3-Agent team running simultaneously — without detailed manual control:

- Frontend/UI Agent: owns the screen and interface layer exclusively
- Backend/Logic Agent: owns calculations and API integrations
- QA/Reviewer Agent: verifies that code from the first two agents works together correctly



DAY 2 - Morning Session

9:00 AM – 12:00 PM

8. Multi-Agent on Claude Code

- Multi-Agent Architecture concepts
- Inter-Agent communication protocols
- Real-world application examples

9. Multi-Agent on OpenCode

- Multi-Agent patterns on OpenCode
- Workshop: Build a 3-Agent team (Frontend, Backend, QA) on OpenCode using Sequential Workflow

10. The "Gate" Pattern & AI Guardrails (The Governance)

- Gate Logic: set Stop Conditions to halt workflows when criteria are met
- Quality Guardrails: define Code/Content quality standards that must pass before proceeding
- Safety & Cost Guardrails: auto-stop conditions to control Token budget spend
- Human-in-the-Loop: connect Gate logic to an approval system
- Workshop: Build a "Quality Gate" to catch Errors and a "Cost Gate" to prevent Token Overflow



DAY 2 - Afternoon Session

1:00 PM – 4:00 PM

11. The "Synthesize" Pattern (The Integration)

- Aggregation Logic: techniques for combining Output from multiple Agents
- Workshop: Build a "Synthesizer Agent" to merge Frontend and Backend code into a single application that can be run and tested end-to-end

12. Collaboration Layer (Multica.ai vs. Ruflow / AgentOps)

- Use Multica.ai to monitor Agent progress via a Kanban Board
- Introduction to AgentOps for Observability — pinpoint exactly where an Agent gets stuck in a Loop
- Workshop: Set up a Kanban Board and assign tasks to Agents through a human-readable interface

13. Capstone Workshop (The Masterpiece)

- Build a complete Multi-Agent system incorporating both Gate and Synthesize patterns for a real business scenario
- Use a Pre-flight Checklist before handing off to Agents for fully autonomous operation



 [Download PDF](#)



Social Media



Tel 02-219-4304



training@9expert.co.th



9Experttraining.com



@9EXPERT

Published Date : 13 July 2026 | [Page : 4](#)