

AWS Kiro (Agentic IDE)

Duration- 2-Days

Course Objective-

This course helps participants learn how to use AWS Kiro to build applications using **spec-driven development, AI agents, automation hooks, and MCP integrations**. By the end, learners will design, build, and automate a real project using Kiro workflows.

Kiro is an **agentic AI IDE** that enables developers to plan using structured specs, automate workflows with hooks, guide AI with steering rules, and integrate external tools through MCP servers.

Course Outcomes-

- Understand Kiro ecosystem and workflow.
- Work comfortably in Kiro environment.
- Build structured AI-driven development workflow.
- Build features using conversational AI.
- Automate development workflows.
- Control AI behavior effectively.
- Extend Kiro with external integrations.
- Complete real-world AI-assisted development project.

Day 1 — Foundations & Spec-Driven Development

Module 1: Introduction to Kiro & Agentic Development

- What is AWS Kiro & use cases
- Agentic IDE concept
- Spec-driven vs traditional coding
- Kiro architecture & components

Lab

- Install Kiro & explore interface
- Create first project workspace

Module 2: Getting Started with Kiro

- Installation & setup
- Navigating Kiro UI & panels
- Opening projects and managing context

- Agentic chat basics

Lab

- Launch Kiro
- Import sample application
- Use chat to explore codebase

Module 3: Spec-Driven Development Workflow

- Creating specs from prompts
- Requirements → Design → Tasks workflow
- Spec structure (requirements, design, tasks)
- EARS requirement syntax overview

Lab

- Generate project spec
- Review design.md & tasks.md
- Execute spec tasks

Module 4: Agentic Chat & AI Coding

- Conversational coding
- Debugging with AI
- Automating code generation
- Context management

Lab

- Generate new feature using chat
- Refactor code using AI assistance

Day 2 — Automation, Integration & Advanced Features

Module 5: Agent Hooks Automation

Topics

- Event-driven automation concepts
- Hook triggers & actions
- Use cases (testing, documentation, security checks)

Lab

- Create hook for auto test generation
- Configure automation workflow

Module 6: Steering & AI Guidance

- Steering files (.kiro/steering)
- Setting coding standards
- Managing project rules
- Persistent AI context

Lab

- Create steering file
- Define project conventions
- Apply AI guidance

Module 7: MCP Servers & Tool Integration

- Model Context Protocol (MCP)
- Integrating external tools & APIs
- MCP configuration & use cases

Lab

- Enable MCP server
- Integrate external tool

Module 8: End-to-End Project

Topics

- Build full application using Kiro workflow

Capstone Lab

Participants build:

- Requirements specification
- Automated workflows
- Feature implementation
- Integrated AI automation