



Claude for Wireless Core Operations and 5G Core NOC Automation

- 5 Days Course
- Hands-on and Lecture

Course Overview

This advanced course teaches learners to use Claude as a practical AI assistant for Wireless Core Operations, 5G Core NOC analysis, agentic AI design, project collaboration, and code-assisted simulator development. It is delivered as 10 half-day sessions, each lasting approximately three to four hours.

Learners progressively build a 5G Core NOC simulator that demonstrates real-time fault detection, KPI monitoring, fault isolation, root-cause analysis, remediation recommendations, supervisory decisioning, guardrails, audit logging, rollback, and human approval for high-risk actions.

The course follows a five-layer agentic architecture covering wireless core data sources, connectors and data processing, use-case agents, supervisory orchestration, and guardrailed execution. Hands-on work centers on Claude Chat, Claude Code, custom local MCP servers, and an open-source Open5GS or free5GC environment.

The final capstone demonstrates how agentic AI can detect, triage, isolate, prioritize, and mitigate faults affecting Data, Voice, 911, and Messaging services. Microsoft Copilot is included only as a final strategic discussion; the practical build remains centered on Claude.

Who Should Attend

- Wireless Core Operations teams
- NOC analysts and incident responders
- 5G Core, IMS, and messaging support teams
- Network engineers and automation engineers
- AI enablement teams supporting telecom operations
- Technical trainers and solution architects
- Managers evaluating AI-assisted operations for 24x7 firefighting environments

What You'll Learn

- Use Claude Chat to explain, analyze, summarize, and reason through Wireless Core and 5G Core operational scenarios.
- Apply prompt engineering with risk assessments and guardrail tags to KPI monitoring, fault triage, root-cause analysis, runbook generation, and incident reporting.
- Configure Claude Code and build custom local MCP connectors that safely expose local files, system state, and database entries.
- Explain the five-layer agentic architecture of data sources, connectors and data processing, use-case agents, supervisory decisioning, and guardrailed execution.
- Use Claude Code to inspect, understand, test, and safely modify an open-source 5GS wrapper and NOC simulator.
- Design specialist agents for KPI monitoring, fault detection and isolation, root-cause analysis, remediation, guardrails and risk, and supervisory orchestration.
- Map Wireless Core faults to critical Data, Voice, 911, and Messaging services.

- Implement or extend a NEXUS-style supervisor that routes faults by severity, priority, specialization, and risk.
- Simulate data access failures, call setup failures, call drops, messaging failures, performance degradation, critical faults, and node Out-of-Rotation decisions.
- Implement guardrails, audit logging, human approval checkpoints, rollback workflows, and incident reporting.
- Generate post-mortems, shift handoff reports, and executive incident summaries.
- Discuss how Microsoft Copilot could support enterprise collaboration after the Claude-centered simulator and operational model are established.

Outline

1. Claude Basics and the Wireless Core NOC Mission

- Claude orientation and Claude Chat fundamentals.
- Using Claude to support technical analysis and operations teams.
- Using Claude to explain telecom concepts and summarize operational problems.
- Initial AI safety boundaries and read-only limits in production networks.
- The role of a NOC in a 24x7 firefighting environment.
- Why 5G Core operations are complex.
- Overview of AMF, SMF, UPF, PCF, UDM, AUSF, NRF, gNB, UE, PDU sessions, QoS, and MTTR.
- Introduction to the five-layer agentic AI architecture.
- Introduction to the local Open5GS or free5GC architecture.
- Hands-on exercise: use Claude to create a 5G Core glossary.
- Hands-on exercise: verify the local open-source 5G Core sandbox.
- Hands-on exercise: summarize the Wireless Core firefighting problem and draft the five-layer architecture map.

2. Prompt Engineering and Early Governance for Fault Triage

- Role prompting and context setting.
- Structured JSON and Markdown output.
- Diagnostic prompting and escalation logic.
- Adding LOW_RISK and HIGH_RISK classifications to technical recommendations.
- Alarm interpretation and symptom-versus-root-cause analysis.
- Severity classification and evidence gathering.
- Integrating standard operating procedure guardrails into diagnostic workflows.
- Hands-on exercise: create prompts for KPI review, fault classification, and runbook lookup.
- Hands-on exercise: produce standardized risk level, human approval, and rollback metadata.
- Hands-on exercise: test prompts against high latency, data access failure, and call-drop incidents.

3. Wireless Core Data Sources and Local Telemetry Extraction

- Technical document and operational log analysis.
- Requirements extraction and runbook drafting.
- Creating service-isolation tables.
- Preparing raw files for ingestion by local tools.
- Identifying active AMF, SMF, and UPF logs, subscriber database status, and network traces.
- Identifying critical indicators for Data, Voice, 911, and Messaging services.
- Analyzing delay, low speed, latency, packet loss, and chronic degradation indicators.
- Hands-on exercise: locate and extract diagnostic logs and trace files from the local 5G Core.
- Hands-on exercise: use Claude to draft operational runbooks from local evidence.
- Hands-on exercise: build a machine-readable data-source inventory and service-isolation matrix.

4. Claude Code, Local MCP Connectors, and Shared Workspace

- Claude Code installation and terminal configuration.
- Designing custom local MCP servers with Node.js or Python.
- Exposing local files, log directories, status scripts, and database state through MCP tools.
- Separating read-only tools from write-action scripts.
- Telemetry ingestion and event normalization.
- Protecting high-risk operational scripts with local approval gates.
- Hands-on exercise: install and configure Claude Code in the local sandbox.
- Hands-on exercise: build a local MCP server that retrieves 5GS logs and database state.
- Hands-on exercise: configure MCP safety policies that disable destructive tools.

5. Claude Code Foundations and 5GS Wrapper Walkthrough

- Reading and tracing unfamiliar codebases with Claude Code.
- Using Claude to explain system APIs.
- Safe local code modification and small refactoring.
- Verifying changes with local unit tests.
- Open-source 5GS script-wrapper topology.
- Programmatic network-function states: Active, Offline, and Out of Rotation.
- Telemetry exporters, severity metrics, and event loggers.
- Hands-on exercise: inspect the 5GS control scripts from the command line.
- Hands-on exercise: generate a visual architecture map of the codebase.
- Hands-on exercise: create a script wrapper that queries subscriber session states.

6. KPI Monitoring and Real-Time Fault Detection

- Writing lightweight dashboard scripts or UI components with Claude Code.
- Converting raw telemetry streams into real-time metrics.
- Debugging state transitions and event logs.
- Monitoring active UEs, throughput, latency, packet loss, and MTTR.
- Simulating high latency, low throughput, packet loss, and chronic degradation.
- Designing the KPI Monitoring Agent behavior model.
- Hands-on exercise: build a Python or Node.js script that polls local 5GS metrics.
- Hands-on exercise: feed metrics to Claude Code through an MCP tool to trigger detection alerts.
- Hands-on exercise: simulate a chronic UPF degradation loop and verify automatic detection.

7. Use-Case Agents and Wireless Core Specialist Agents

- Multi-agent role design and responsibility scoping.
- Implementing specialist-agent prompts with controlled tool access.
- Managing safe handoffs between specialist scripts.
- Defining escalation boundaries when a specialized tool fails.
- Designing KPI Monitoring, Fault Detection and Isolation, RCA, Remediation, and Guardrail and Risk agents.
- Designing AMF, SMF, UPF, PCF, AUSF and UDM, IMS and Voice, Messaging, and gNB domain specialists.
- Introducing the NEXUS Supervisor routing model.
- Hands-on exercise: write the underlying prompts for each specialist agent.
- Hands-on exercise: build a file-based message queue for diagnostic handoffs.
- Hands-on exercise: create routines that retrieve telemetry for Data, Voice, 911, and Messaging faults.

8. NEXUS Supervisor, Critical Service Isolation, and RCA Flow

- Writing supervisory orchestration logic in local scripts.
- Creating dynamic routing tables from incoming JSON telemetry.

- Implementing fallback strategies when script tasks fail.
- Generating chronological Markdown audit logs of agent decisions.
- Supervisory routing, severity classification, and priority queues.
- Critical service isolation for Data, Voice, 911, and Messaging.
- Root-cause analysis correlation loops.
- Hands-on exercise: build the NEXUS Supervisor event-processing loop.
- Hands-on exercise: implement immediate priority escalation for 911 and IMS events.
- Hands-on exercise: generate automated RCA reports in a local output folder.

9. Guardrails, Security, and Human-in-the-Loop Node Control

- Reviewing and auditing previously implemented prompt guardrails.
- Implementing interactive CLI approval prompts for high-risk tools.
- Testing edge cases to verify that unauthorized scripts are blocked.
- Designing state-rollback workflows.
- Enforcing Layer 5 guardrails and execution limits.
- Node Out-of-Rotation decision paths.
- Comprehensive audit logging and security boundaries.
- Hands-on exercise: require human confirmation before restarts or Out-of-Rotation actions.
- Hands-on exercise: build an automatic rollback mechanism for failed recovery actions.
- Hands-on exercise: conduct red-team tests using unauthorized destructive-command requests.
- Hands-on exercise: verify that local security profiles block prohibited actions.

10. Capstone Simulator Demo and Copilot Enterprise Discussion

- Orchestrating the complete local 5G Core NOC simulation.
- Validating the Detection → Triage → RCA → Remediation → Guardrail Check → Approval → Action → Verification workflow.
- Simulating real-time critical faults and KPI drops.
- Simulating high latency, packet loss, and random call drops.
- Simulating UE registration and PDU session failures.
- Simulating IMS call setup failures and call drops.
- Simulating mobile-originating and mobile-terminating messaging failures.
- Simulating 911 service-routing and availability concerns.
- Simulating node Out-of-Rotation recommendations and human approval.
- Generating incident summaries, post-mortems, and shift handoff reports.
- Hands-on exercise: run the complete simulator against the local open-source 5G Core.
- Hands-on exercise: demonstrate human-in-the-loop validation for node Out-of-Rotation decisions.
- Hands-on exercise: generate a polished shift handoff report and executive incident summary.
- Strategic discussion of Claude Code and Microsoft 365 Copilot enterprise collaboration.

Prerequisites

- Strong 5G NOC skills.