



## Gemini CLI for Developers – Intermediate

- 1 Day Course
- Hands-on and Lecture

### Course Overview

This course dives deep into enterprise-grade workflows using Gemini CLI. Learners master million-token context ingestion for whole-repository analysis, integrate multimodal UI diagrams and PDFs into terminal workflows, build self-correcting test-driven development (TDD) loops, and script headless AI automation steps inside Git hooks and CI/CD runners.

### Who Should Attend

- Intermediate to Advanced Developers
- Systems Architects
- Technical Leads

### What You'll Learn

- Exploit Gemini's 1M+ token context window for full-repository reasoning and architecture mapping.
- Perform multimodal analysis in terminal workflows using local images, diagrams, and PDFs.
- Build automated testing loops where Gemini CLI fixes failing test suites autonomously.
- Manage token optimization, context compaction, and cost efficiency in long sessions.
- Implement structured JSON schema generation for automated pipeline steps.

### Outline

#### 1. Massive Context Window Strategy & Repo Mapping

- Leveraging 1M+ token windows for large codebase ingestion
- Token usage analytics, context window limits, and cost monitoring
- Summarizing legacy codebases, monolithic repos, and multi-file dependencies
- Context compaction techniques and session state preservation
- Hands-on: Ingesting an Entire Legacy Repository for Structural Analysis
- Hands-on: Generating Architectural Diagrams and Onboarding Guides from Code

#### 2. Multimodal Terminal Workflows & Media Analysis

- Feeding local assets to Gemini CLI: Mockups, architecture diagrams, and PDFs
- Extracting UI/UX specs from design images directly into frontend code
- Analyzing diagnostic charts, log dumps, and visual error reports
- Combining text context files (@file) with visual assets
- Hands-on: Translating UI Mockups (PNG/SVG) into Functional Code
- Hands-on: Diagnosing Network Logs and Performance Screenshots via CLI

### 3. Autonomous Test-Driven Development Loops

- Setting up autonomous TDD cycles with Gemini CLI
- Configuring test execution scripts and error log capturing
- Self-correcting code loops: Reading stack traces, patching code, and re-running tests
- Managing non-deterministic output and preventing endless modification loops
- Hands-on: Building an Automated Test-Fix-Refactor Loop
- Hands-on: Automated Migration of Legacy Test Suites to Modern Frameworks

### 4. Non-Interactive Scripting & Headless Automation

- Running Gemini CLI in headless/CI mode using pipe inputs
- Integrating Gemini CLI into Git hooks and CI/CD runners (GitHub Actions / GitLab CI)
- Generating structured JSON outputs for down-stream build scripts
- Enforcing security compliance and credential sanitization in automated pipelines
- Hands-on: Scripting Headless Code Quality Audits via CLI Pipes
- Hands-on: Building a Git Pre-Commit Hook for Automated AI PR Reviews

### Prerequisites

- Completion of Gemini CLI for Developers – Beginner (ai-gem-1d1) or equivalent hands-on experience.
- Familiarity with Git branching, pull requests, and CI/CD pipelines.
- Comfortable working with JSON, REST APIs, and command-line diagnostics.

### Next Courses

- ai-gem-3d1