GenAI Engineering Mastermind: 2-Day Intensive to Master LLM APIs, Prompt Engineering, and Multi-Agent Systems

Course Overview

This accelerated 2-day mastermind (12+ hours of live sessions) equips engineers, developers, and tech enthusiasts with practical skills to build production-grade generative AI applications. From API integration and advanced prompting to agentic workflows and multi-agent collaboration, you'll gain hands-on experience using the latest 2025 tools and frameworks. Suitable for beginners with basic coding knowledge or experienced developers advancing into agentic AI.

- **Target Audience**: Developers, software engineers, and AI builders seeking to integrate GenAI into real applications.

- **Prerequisites**: Basic Python knowledge; familiarity with APIs is helpful.

- **Format**: Live interactive sessions with code demos, pair programming, Q&A, and group debugging; recordings and code repos provided.

- **Outcomes**: Build deployable conversational agents, automated workflows, and a multi-agent MVP; understand production considerations like cost, safety, and scaling. GitHub portfolio template included.

- **Resources**: Shared code repository, prompt templates, API keys guide (free tiers), and private community for ongoing support. Bonus: Ethical agent design guidelines and 2025 trends overview.

Day 1: LLM Foundations, APIs, and Advanced Prompt Engineering

Session 1: Introduction to LLMs, OpenAI APIs & Prompt Engineering (Morning, ~6 hours)

What Will You Learn?

1. Deep dive into LLM architecture: Transformers, attention mechanisms, scaling laws, and capabilities of 2025 models (e.g., GPT-5.2, Claude 3.5, Grok).
2. Hands-on with modern APIs: OpenAI Responses API/AgentKit, Anthropic tools, Groq for speed.
3. Advanced prompt engineering: Chain-of-thought, few-shot, scaffolding, structured outputs (JSON), confidence calibration, and adversarial testing.
4. Building conversational agents: Tool-calling, streaming responses, memory management.
5. Optimization techniques: Temperature/top-p control, cost-aware prompting, multi-model routing.
6. Ethical tip: Implementing guardrails for safety and bias mitigation.

Tools (Updated for December 2025)

- OpenAI API (GPT-5.2, Responses API, AgentKit)
- Anthropic Claude API
- Groq/Grok for inference
- LangChain basics for chaining

Estimated Time: 6 hours live + 2 hours assignments.

Assignments

1. **Set up API access and build a basic agent**:
    - Step 1: Obtain API keys (OpenAI/Anthropic free tiers) and test completions with GPT-5.2.
    - Step 2: Implement tool-calling (e.g., weather/search function).
    - Step 3: Add streaming and error handling.

    - Expected Outcome: Deployed simple chatbot (e.g., via Streamlit). Share repo link.
    - Tips: Use Responses API for built-in tools.
2. **Master advanced prompting**:
    - Step 1: Experiment with 5 techniques (e.g., CoT for math, few-shot for classification).
    - Step 2: Optimize for consistency across models; measure with evals.
    - Step 3: Build a reusable prompt template library.
    - Expected Outcome: Notebook with comparisons and best practices report.

Day 2: Agentic Workflows and Multi-Agent Systems

Session 2: Advanced AI Applications & Automation (Morning, ~3 hours)

What Will You Learn?

1. Agentic AI architecture: Reasoning loops, tool integration, planning/reflection.
2. Designing multi-step workflows: Self-improving loops, human-in-the-loop.
3. Scaling agents: Cost monitoring, rate limits, fallback strategies.
4. Frameworks overview: LangGraph for stateful flows, CrewAI for role-based.
5. Building autonomous applications: Research agents, data processing pipelines.

Tools

- LangChain/LangGraph
- CrewAI/AutoGen
- LangSmith for tracing

Estimated Time: 3 hours live + 2 hours assignments.

Assignments

1. **Build a self-improving agent workflow**:
    - Step 1: Use LangGraph to create a reasoning + tool-calling loop.
    - Step 2: Add reflection (e.g., critique output and retry).
    - Step 3: Integrate external tools (e.g., web search).
    - Expected Outcome: Deployed agent handling multi-step tasks (e.g., research summary).

Session 3: Multi-Agent AI Systems (Afternoon, ~3 hours)

What Will You Learn?

1. Blueprint for multi-agent teams: Role assignment, communication protocols.
2. Coordinating collaboration: Hierarchical vs. peer-to-peer.
3. Role-based agents: Ideator, coder, tester, deployer.
4. Real-world patterns: Productivity boosts in dev, ops, research.
5. Production considerations: Observability, security, evals.
6. Capstone: Build and deploy a multi-agent MVP.

Tools

- LangGraph/CrewAI for orchestration
- AutoGen alternatives
- OpenAI AgentKit for quick multi-agent

Estimated Time: 3 hours live + 2 hours capstone.

Assignments & Capstone

1. **Develop a multi-agent MVP**:
    - Step 1: Define roles (e.g., planner, executor, reviewer).
    - Step 2: Implement collaboration (e.g., build a simple app from spec).
    - Step 3: Add monitoring and test coordination.
    - Expected Outcome: Live multi-agent demo (e.g., automated code generation pipeline). Share video/repo.
2. **Group discussion & optimization**:
    - Step 1: Present MVP in breakout.
    - Step 2: Iterate based on feedback (cost/safety).
    - Expected Outcome: Final polished project for portfolio.

What You'll Walk Away With

- Proficiency in 2025 LLM APIs and prompt mastery for reliable outputs.
- Deployable single and multi-agent systems.

- Hands-on experience with top frameworks (LangGraph, CrewAI).
- A production-ready GenAI MVP in your portfolio.
- Skills to build/scaling agentic applications autonomously.

This enhanced 2-day program incorporates December 2025 updates (e.g., GPT-5.2, Responses API/AgentKit, LangGraph dominance) for relevance, adds detailed hands-on assignments, and extends depth with modern techniques/tools while maintaining the 12-hour structure. Paste into Word for editing!