AI ENGINEERING

Build production-ready AI apps with Hands-on training

Course Modules

1. Foundation (Week 1)

Python for AI + First LLM Calls

Key Topics:

- Git & GitHub from day one
- Python environment setup (venv, pip, .env files)
- OpenAI and Anthropic SDK setup
- Role-based messaging (system, user, assistant)
- Prompt engineering fundamentals (zero-shot, few-shot, chain-of-thought)
- Model comparison: GPT-4o vs Claude 3.5 Sonnet vs Gemini
- API keys, rate limits, billing awareness
- **Deliverable:** CLI Q&A bot

2. Chatbot Skills (Week 2)

Memory, Streaming & Building a Chat UI

Key Topics:

- Multi-turn conversation history
- Session memory vs long-term memory
- System prompt engineering (personas, tone, guardrails)
- Streaming responses with server-sent events
- Build a polished Streamlit chat UI
- Streamlit Cloud deployment (live URL)
- Professional code structure (separating LLM logic from UI)
- **Deliverable:** Live deployed chatbot with public URL

3. RAG Systems (Week 3)
Single-doc & Multi-document RAG Pipelines

Key Topics:

- Why RAG: context limits, knowledge cutoffs, reducing hallucinations
- Embeddings demystified (text → vectors)
- Vector databases: ChromaDB (local) and Pinecone (cloud)
- Document loaders for PDFs, Word docs, web pages
- Chunking strategies (size, overlap, quality)
- Full RAG pipeline: ingest → embed → store → retrieve → generate
- Multi-document RAG with source attribution
- Evaluating RAG quality (faithfulness, relevance)
- **Deliverable:** PDF Q&A system with citations

4. Tool Use & Agents (Week 4)
Function Calling + the ReAct Agent Loop

Key Topics:

- Function calling fundamentals
- Defining tools in JSON schema
- Calling real external APIs from LLM loops
- Handling tool results and continued reasoning
- Multi-tool assistant (weather, calculator, web search)
- ReAct agent pattern from scratch
- LangChain agents — when to use a framework
- Error handling and graceful recovery
- **Deliverable:** Tool-using AI assistant

5. Multi-Agent Systems (Week 5)
Multi-Agent Systems with CrewAI

Key Topics:

- Why multi-agent: parallelism, specialisation, scale
- Agent roles and personas (Researcher, Writer, Critic, Planner)
- CrewAI framework (Crew, Agent, Task, Process)
- Inter-agent communication and handoffs
- Tool assignment per agent
- Hierarchical process and Manager agents
- Handling agent failures and retries

- Real-world use cases (research, content, data analysis)
- **Deliverable:** Multi-agent research system

6. Advanced RAG (Week 6)

Multi-RAG, Hybrid Search & Re-ranking

Key Topics:

- Limitations of basic RAG
- Hybrid search (dense + sparse BM25)
- Re-ranking with Cohere Rerank and FlashRank
- Multi-RAG routing across vector stores
- Parent-child chunking strategy
- Hypothetical Document Embeddings (HyDE)
- Self-querying retrieval
- RAG evaluation with RAGAS metrics
- **Deliverable:** Enterprise Q&A system

7. Production & Deployment (Week 7)

FastAPI Backend + Docker + Cloud Deployment

Key Topics:

- FastAPI fundamentals (routes, async, Pydantic)
- Wrapping LLM/RAG logic behind REST APIs
- API key authentication middleware
- Background tasks for long-running calls
- Auto-generated Swagger docs
- Docker (Dockerfile, build, run)
- Environment variables and secrets management
- Deploy to Railway / Render
- Logging and Sentry error tracking
- **Deliverable:** Live deployed AI API on the cloud

8. Capstone Project (Week 8)

Build, Demo, Deploy

Key Topics:

- Project scoping and architecture design
- Core build: LLM/agent/RAG logic + API + UI
- Integration, testing, end-to-end QA
- Cloud deployment with public URL
- Demo Day — 5-minute live demo
- GitHub repo polish (README, architecture diagram)
- CV & LinkedIn presentation
- Roadmap for AI specialisation after the course
- **Capstone Tracks:** SaaS micro-tool · Freelance project · Portfolio project
- **Deliverable:** Deployed AI product + GitHub repo + recorded demo