

Build Agentic Apps Bootcamp

A hands-on 2-day sprint — Syllabus & Resources

Duration	2 days
Daily hours	9:00 – 18:00
Format	Live sessions + hands-on labs
Level	Intermediate builders

Learning outcomes

- Design agent loops with planning, tool use, and memory
- Implement function calling and structured outputs reliably
- Build a RAG pipeline with citations and freshness controls
- Add guardrails against prompt injection and unsafe outputs
- Write evals that catch regressions before users do
- Deploy an agent behind a real API with observability

Day 1 — Foundations & your first agent

09:00 – 10:00 — Agentic AI, demystified

What separates an agent from a chatbot: planning, tool use, memory, and autonomy loops.

10:00 – 11:30 — LLM foundations for builders

Tokens, context windows, structured outputs, function calling, and cost/latency trade-offs.

11:30 – 13:00 — Prompt engineering that scales

System prompts, few-shot patterns, chain-of-thought, and prompt versioning.

14:00 – 15:30 — Tools & function calling (lab)

Wire your first agent to real tools — web search, calculators, and a custom API.

15:30 – 17:00 — RAG for grounded answers (lab)

Chunking, embeddings, vector search, and retrieval-augmented generation with citations.

17:00 – 18:00 — Day 1 wrap + Q&A;

Review builds, debug common failure modes, and preview Day 2.

Day 2 — Memory, safety & shipping

09:00 – 10:30 — Agent architectures

ReAct, plan-and-execute, multi-agent, and when to use each pattern.

10:30 – 12:00 — Memory & state (lab)

Short-term context, long-term vector memory, and per-user personalization.

12:00 – 13:00 — Guardrails & safety

Prompt injection defenses, output validation, PII handling, and human-in-the-loop.

14:00 – 15:30 — Evaluation & observability

Build an eval suite, trace runs, track regressions, and score with LLM-as-judge.

15:30 – 17:00 — Ship it (capstone)

Deploy your agent behind an API + minimal UI, with auth, rate limits, and logging.

17:00 – 18:00 — Demos & next steps

Everyone demos, peer feedback, and a roadmap for taking your agent to production.

Pre-work checklist

- Install Node.js 20+ or Python 3.11+
- Set up VS Code with the recommended extensions
- Obtain an OpenAI, Anthropic, or Lovable AI Gateway API key
- Read: 'A Survey of Large Language Models' (skim sections 1–3)
- Skim: LangChain and LlamaIndex quickstart docs

Resources bundle

The resources ZIP includes starter repos, prompt templates, an eval harness skeleton, and a curated reading list.

- starter/agent-node — Node.js starter with function calling
- starter/agent-python — Python starter with LangChain
- prompts/ — reusable system prompt templates
- evals/ — LLM-as-judge harness + example dataset
- reading-list.md — curated papers, blog posts, and videos