# Agentic AI Architecture and Forward Deployed Engineering

Customized training for senior engineers and architects | Trainer: Rajinikanth Vadla | WhatsApp: +91 9100028801 | November 2026 batch

## About this course

This course is for experienced people, 12 years and above, who want to move into senior AI roles. That means AI architect, forward deployed engineer, AI advisory and consulting, or building your own AI product as a founder.

At this level, knowing how to build an agent is only half the job. You also need to sit with a client, find the right use case, design something that is secure and can pass their security review, show the business value, and take it from a POC to production. This course covers both sides.

We work on one client style project from start to end. Every week we add one piece to it, the same way it happens in a real engagement. Hands-on coding is part of every class, because in FDE and architect roles you are expected to build, not only draw diagrams.

This is a customized training. With a small batch I can adjust the depth and the examples based on your background, your industry and the roles you are targeting.

## Course details

| | |
|---|---|
| **Training type** | Customized training, depth and examples adjusted to the batch |
| **Who it is for** | Architects, senior engineers, tech leads, consultants and founders with 12+ years of experience |
| **Duration** | 8 to 10 weeks |
| **Classes** | 3 classes per week, 2 hours each |
| **1:1 sessions** | 30 minutes every week with each student, for design review, doubts and career direction |
| **Mode** | Online, live classes. Recording shared after every class |
| **Batch size** | 5 to 6 students |
| **Course fee** | INR 70,000 per student |
| **Support after course** | 15 to 20 days of doubt support |

## Who should join

- Solution and enterprise architects moving into AI architecture
- Senior engineers and tech leads who want forward deployed engineer (FDE) roles
- Consultants who want to run AI advisory and delivery projects for clients
- Founders and product leaders building AI products or AI services
- People targeting roles like AI Architect, Principal AI Engineer, Head of AI or AI Practice Lead

You should be able to read and write Python. You do not need prior AI experience. Your experience with systems, cloud, data and clients is what we build on.

Rajinikanth Vadla

## How the classes run

Each class has a short concept part, then a live build, then you apply it to your own client project. Many classes end with a design discussion where we compare options and trade-offs, like we would in an architecture review.

Security is covered in two full weeks, and we also do a short security check on whatever we build every week. Clients ask about security first, so you should be able to answer those questions with confidence.

## Your project: a client engagement

Each of you takes a different client case and runs it like a real FDE or consulting engagement. Some cases:

- Bank: agent that helps the loan operations team review applications
- Hospital: agent that answers staff questions from policies and patient workflow documents
- SaaS company: support agent connected to tickets, CRM and product docs
- Manufacturing: agent that reads machine alerts and suggests maintenance steps
- IT services company: incident response agent for their cloud platform
- Your own industry or startup idea, if we agree on it in week 1

At the end you will have:

- A discovery note and a business case with expected ROI
- An architecture document with the key design decisions
- A threat model and security review of your agent
- A working agent with MCP tools, RAG, memory and human approval, deployed on the cloud
- An evaluation report that works as acceptance criteria for the client
- A short executive presentation, the same way you would present to a client CTO

## Week wise syllabus

The plan below is for 10 weeks. If the batch moves faster, we finish in 8 weeks by combining weeks 1 and 2, and weeks 8 and 9.

**Week 1: Agentic AI for architects**

- How LLMs work, in the depth an architect needs: tokens, context, reasoning models
- Choosing models: cost, latency, quality, open source vs closed, hosting options
- Where agents work well and where they fail
- Workflows vs agents, and how much autonomy to give
- Building a raw agent loop with tool calling in plain Python

**Hands-on:** Pick your client case. Build a first agent in plain Python with 2 or 3 tools.

Rajinikanth Vadla

**Week 2: Agent design patterns and LangGraph**

- ReAct, planner and executor, reflection
- LangGraph: state, nodes, routing, checkpoints
- Human in the loop and approval steps
- Error handling, retries, timeouts and fallbacks
- Writing architecture decision records (ADRs)

**Hands-on:** Rebuild your agent in LangGraph with a human approval step. Write your first ADR.

**Week 3: Tools, MCP and enterprise integration**

- Designing tools that models can use reliably
- MCP: host, client and server, building your own MCP server
- Connecting to enterprise systems like CRM, ERP, ticketing and databases
- A2A protocol and agent skills
- API gateways and integration patterns used in large companies

**Hands-on:** Your agent works through your own MCP server connected to your client's systems (mocked).

**Week 4: Enterprise RAG and memory**

- RAG architecture: ingestion, chunking, embeddings, vector databases
- Hybrid search, re-ranking and agentic RAG
- Permission aware retrieval, so users only see data they are allowed to see
- Short term and long term memory
- Data quality and keeping knowledge fresh

**Hands-on:** Add a knowledge base from your client's documents with citations and access control.

**Week 5: Multi agent systems and reference architectures**

- Supervisor, handoff and hierarchical patterns
- When multi agent is worth it and when it is not
- Reference architectures on AWS Bedrock, Azure AI Foundry and Google Vertex AI
- Build vs buy, and choosing a platform for a client
- Scaling, latency and cost planning

**Hands-on:** Split your agent into specialist agents. Draw the target architecture for your client's cloud.

**Week 6: AI security - threats and attacks**

- OWASP Top 10 for LLM applications and agentic AI threats
- Threat modeling an agent system
- Prompt injection, direct and indirect (through documents, emails and web pages)
- Tool misuse, excessive agency and data leakage
- MCP security: tool poisoning, malicious servers, token handling
- Red teaming with Promptfoo, Garak and PyRIT

**Hands-on:** Write a threat model for your agent. Red team each other's agents and record what breaks.

Rajinikanth Vadla

**Week 7: AI security - secure design and governance**

- Identity and access for agents: OAuth, scoped tokens, least privilege
- Sandboxing code execution and isolating tools
- Guardrails for input and output, PII masking, secrets management
- Audit logs and traceability of every agent action
- Supply chain risk: models, packages and third party MCP servers
- Governance and compliance: NIST AI RMF, ISO 42001, EU AI Act, India DPDP Act

**Hands-on:** Fix the issues from week 6 and prepare a security review document a client CISO would accept.

**Week 8: Evaluation, observability and LLMOps**

- Evals as acceptance criteria for a client
- Golden datasets, LLM as a judge, checking tool calls and full runs
- Tracing with Langfuse and OpenTelemetry
- Cost tracking, caching and model routing
- Running evals in CI/CD

**Hands-on:** Eval suite running in GitHub Actions and a dashboard showing quality, cost and latency.

**Week 9: Deployment and forward deployed engineering**

- Deploying with FastAPI, Docker and CI/CD on the cloud
- Working inside client environments: VPCs, private networking, on-prem limits
- How an FDE works: discovery workshops, scoping, POC to production
- Use case prioritisation and building the business case and ROI
- Writing a statement of work, estimating and pricing an engagement
- Founder view: picking a wedge, AI product moats, AI services vs AI products

**Hands-on:** Deploy your agent. Write a discovery note, business case and a one page SOW for your client.

**Week 10: Client presentation and career**

- Presenting to executives: what a CTO, CISO and CFO each want to hear
- Handling objections on security, cost and accuracy
- Final project review as a client panel (I play the client)
- Mock interviews for AI architect and FDE roles, including system design and case rounds
- Positioning yourself: LinkedIn, personal brand and finding consulting clients

**Hands-on:** Present your full engagement to the client panel. Final architecture, security review and demo.

## Tools we will use

Python, Claude and OpenAI APIs, open source models with Ollama, LangGraph, MCP, Chroma / pgvector / Qdrant, Langfuse, OpenTelemetry, Promptfoo, Garak, PyRIT, guardrail libraries, FastAPI, Docker, GitHub Actions, and AWS Bedrock, Azure AI Foundry or Google Vertex AI for deployment.

Rajinikanth Vadla

## Assessment and certificate

There is no written exam. I will look at your engagement documents, your working agent, your security review and your final presentation to the client panel. You get the certificate after the final presentation.

## Career support

- 2 mock interviews for each student, for AI architect or FDE roles, with feedback
- Resume and LinkedIn review, positioning you for senior AI roles
- Guidance on moving into consulting or starting an AI services business
- Your engagement documents can be used as case studies when you talk to clients or employers

## Few things to note

- If you miss a class, please inform me before. Watch the recording and we can clear doubts in the 1:1.
- Class timings can be changed with 24 hours notice if all of us agree.
- You will use your own API keys and cloud account. We will use cheaper models while building, so the cost stays small.
- Client cases use sample or mock data. Please do not bring your company's confidential data into class.
- The project and documents are yours. You can use them in your portfolio.

For any questions, message me directly on WhatsApp: **+91 9100028801**

**Rajinikanth Vadla**

Rajinikanth Vadla