

# CRACK THE AI ENGINEERING INTERVIEW — 2025 COMPLETE GUIDE

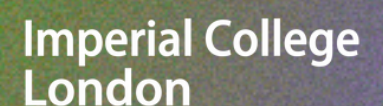
QUESTIONS, ANSWERS & INSIDER FRAMEWORKS TO LAND YOUR DREAM ROLE



Manu Jayawardana  
(Exited Founder of Rise AI)



Aki Wijesundara, PhD  
(Ex-Google AI Accelerator)



This guide distills the essential technical, practical, and behavioral knowledge needed to become a top-tier AI Engineer in 2025.

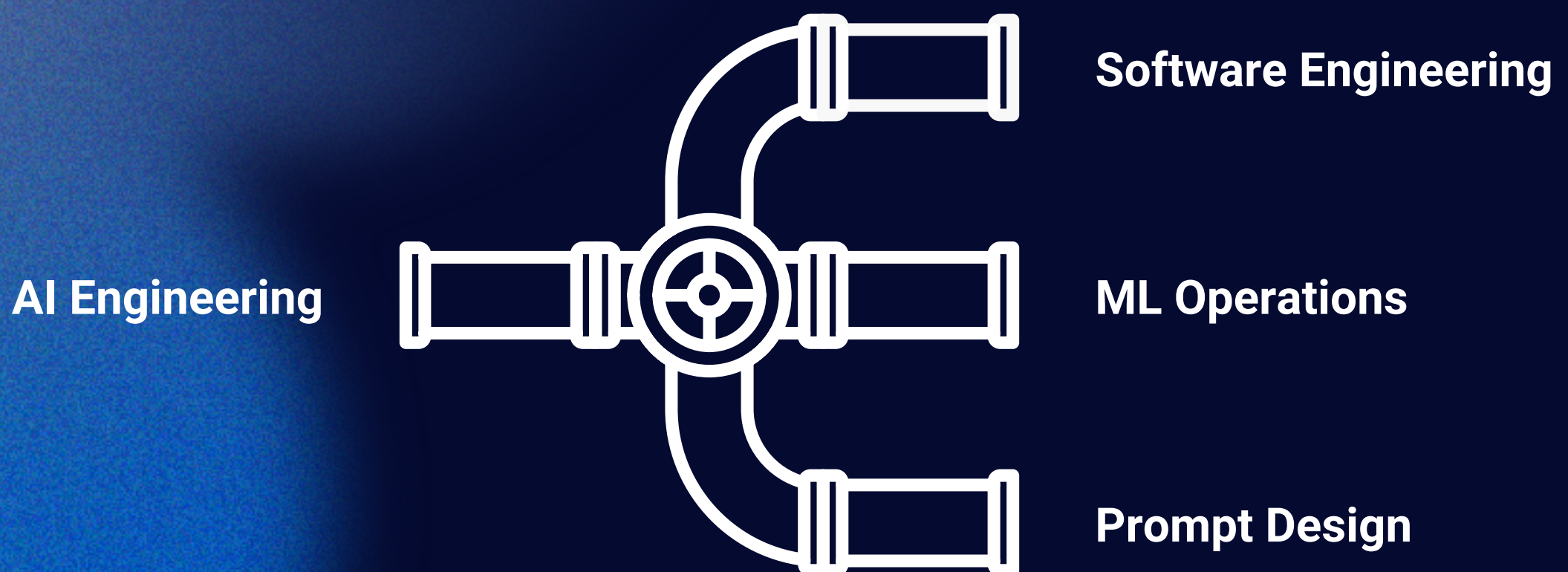


## INTRODUCTION

AI engineering is one of the fastest-growing roles in 2025, blending coding, system design, and applied reasoning. This guide covers how to prepare, what to expect in interviews, and how to stand out as a candidate who can ship real systems.

An AI Engineer sits at the intersection of software engineering, ML operations, and prompt design — bridging models, infrastructure, and user experience.

## Exploring the Dimensions of AI Engineering





# 1. CORE SKILL MAP

Employers test across seven domains:

- **Programming** — Python, TypeScript, async APIs, debugging, software patterns
- **LLM APIs & Prompting** — LangChain, OpenAI, Anthropic, structured prompts.
- **RAG Pipelines** — Embeddings, chunking, retrieval flow, caching.
- **Agents & Tools** — Function-calling, orchestration logic, planning.
- **Deployment & Infrastructure** — Docker, GPUs, FastAPI, Render, Kubernetes.
- **Observability** — Logging, tracing, LangFuse, cost optimization.
- **Collaboration & Product Sense** — Agile, GitHub workflows, prioritization, communication.

## AI Development Skills



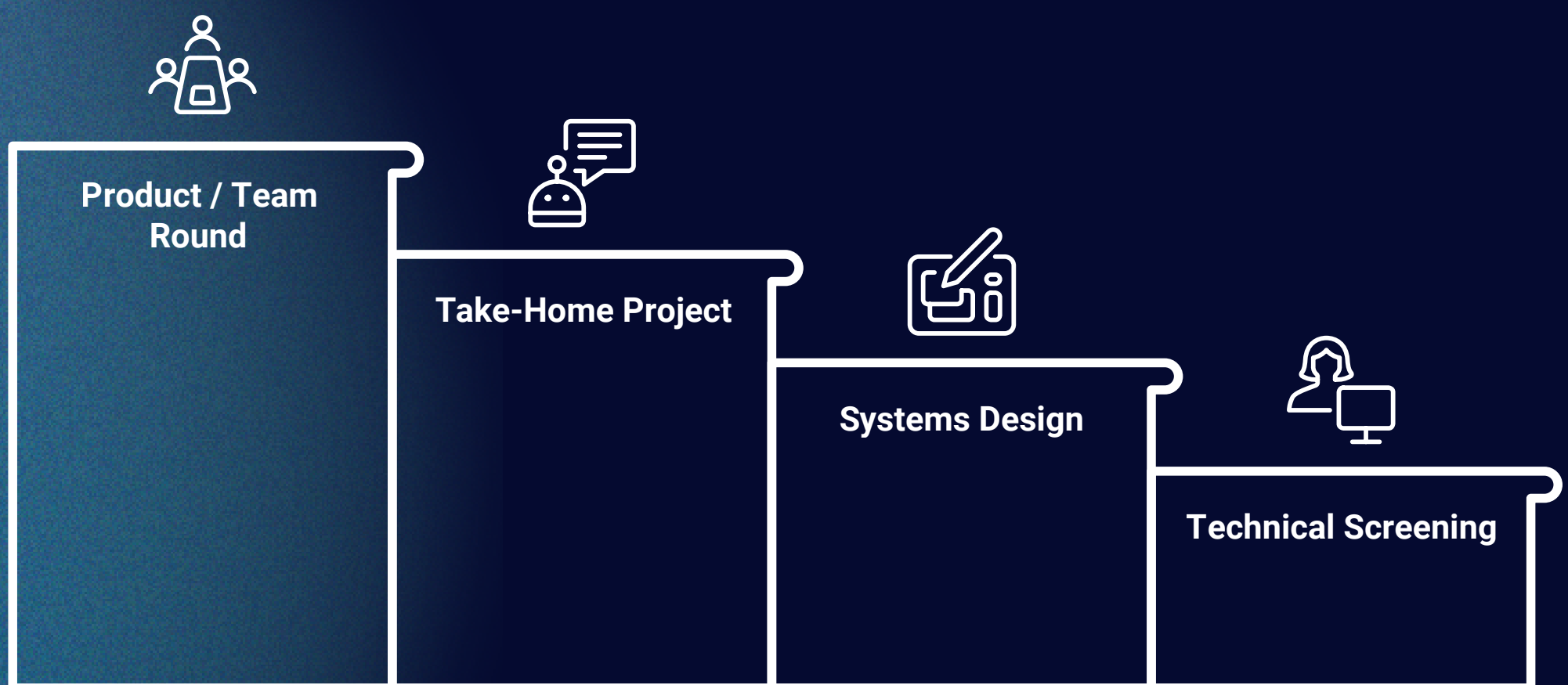
**Pro Tip:** Most interviews focus on applied implementation rather than ML theory — emphasize building reliable systems.



## 2. INTERVIEW FORMATS

- **Technical Screening** — 30 min coding test and quick API reasoning.
- **Systems Design** — 45–60 min session to design an AI application end-to-end.
- **Take-Home Project** — Mini RAG or chatbot assignment to showcase applied ability.
- **Product / Team Round** — Assess collaboration, clarity, and design trade-offs.

### AI Interview Process



Each stage measures different strengths: code quality, reasoning, scalability, and communication.



### 3. EXAMPLE TECHNICAL QUESTIONS

#### A. Coding & Infrastructure

Q: Difference between multiprocessing and multithreading?

A: Multiprocessing uses multiple CPU cores; avoids GIL bottlenecks.

Q: How to containerize a LangChain app?

A: Use Dockerfile, expose port 8000, run Uvicorn.

Q: Role of GPUs in inference vs training?

A: Inference = memory bandwidth; Training = gradient computation.

#### B. LLMs & Prompt Engineering

Q: Why use few-shot prompting?

A: Improves structure and consistency with minimal examples.

Q: How to evaluate prompt quality?

A: Use eval datasets and metrics for accuracy, tone, and stability.

Q: Example of chain-of-thought prompt?

A: "Let's reason step-by-step before answering."



## C. Retrieval & RAG

Q: What's chunk size and overlap?

A: Segmentation of text for retrieval granularity.

Q: FAISS vs Qdrant?

A: FAISS = local prototype; Qdrant = scalable cloud DB.

Q: How to cache embedding queries?

A: Use Redis or SQLite with hash keys.

## D. Agents & Tools

Q: What is a tool call?

A: A structured function execution invoked by a model.

Q: Multi-agent collaboration pattern?

A: Planner-Executor or Supervisor-Worker architectures.

Q: What are MCP Tools?

A: Memory, Control, Planning modules for modular orchestration

## E. Deployment & Observability

Q: What is LangFuse?

A: Open-source LLM observability and trace tool.

Q: How to store secrets?

A: Use environment variables or secret managers, never commit keys.

Q: How to scale an API?

A: Containerize → Load balance → Monitor → Auto-scale.



## 4. SYSTEMS DESIGN DEEP DIVE

**Example Prompt:** “Design an AI document-search assistant for a law firm.”

Expected framework:

1. Ingest → Chunk → Embed → Store in Qdrant.
2. Query → Embed → Retrieve → Context → LLM → Answer.
3. Add caching, observability, and authentication.



Follow-Up: Discuss cost control, data privacy, and feedback loops.



## 5. BEHAVIORAL & CULTURAL FIT

### Sample Questions:

- Tell me about a project where you built an AI feature end-to-end.
- How do you handle rapid framework changes?
- Describe explaining model limits to non-technical stakeholders.
- When have you prioritized shipping over perfection?

### Frameworks:

- STAR — Situation|Task|Action|Result.
- Failure → Learning → Iteration loop.



## 6. TAKE-HOME PROJECT EXAMPLE

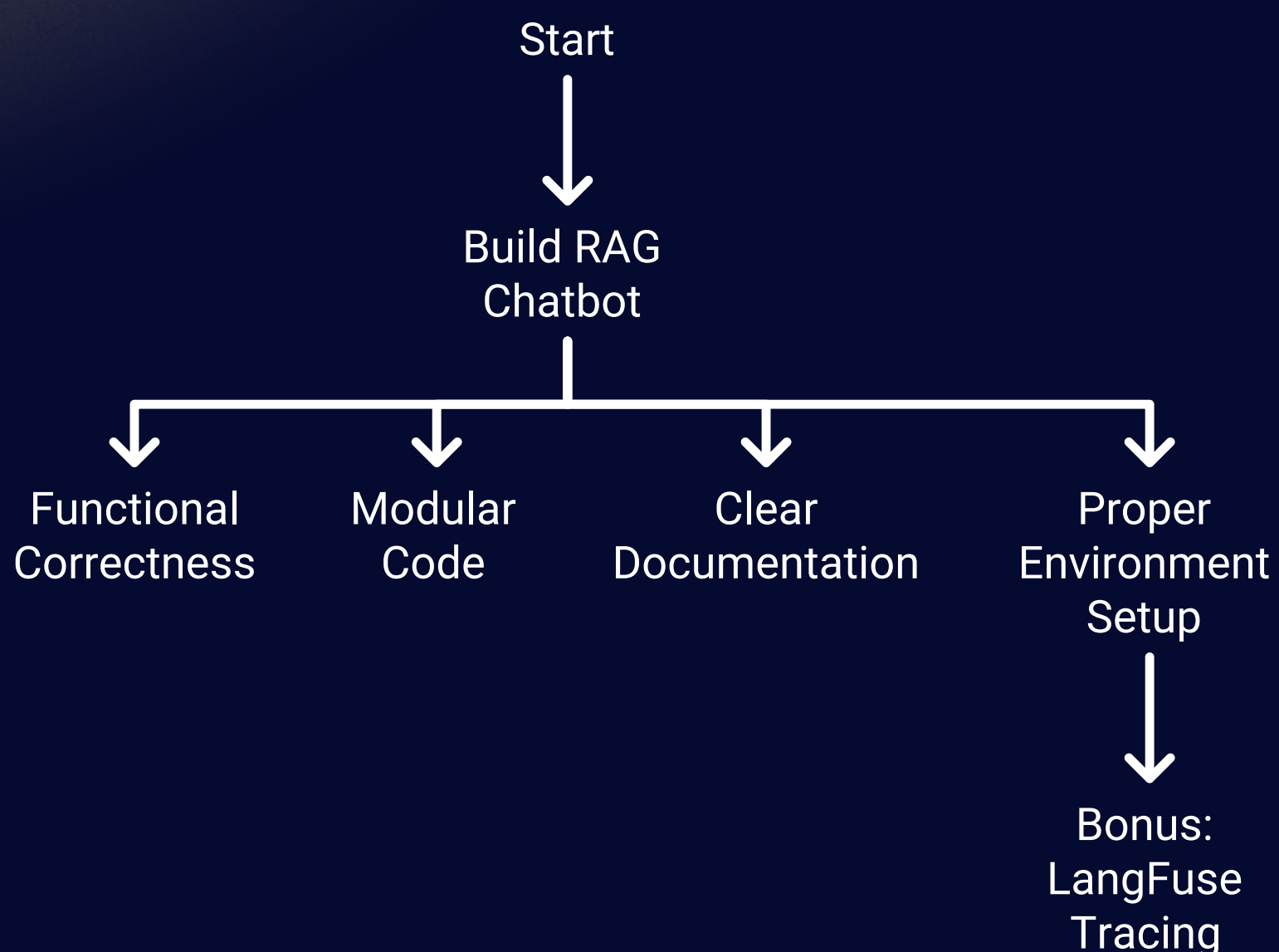
**Task: Build a RAG chatbot using LangChain + Chroma**

**+ public PDFs.**

Evaluation Criteria:

- Functional correctness of retrieval.
- Modular code and clear documentation.
- Proper environment setup.
- Bonus: Include LangFuse tracing.

### RAG Chatbot Development Process





## 7. RECRUITER & HIRING MANAGER INSIGHTS

- Teams value builders who understand cost, latency, and reliability.
- Strong Git hygiene and clear documentation are key.
- Demonstrate awareness of privacy and bias considerations.
- Show ability to ship production systems, not just prototypes.

**Insider Tip:** Hiring managers now test applied reasoning more than ML math.

## 8. 30-DAY PREPARATION PLAN

- **Week 1** — Revisit Python, FastAPI, and Docker fundamentals.
- **Week 2** — Build two mini-projects (Chatbot + RAG pipeline).
- **Week 3** — Deploy one project on Render or Fly.io.
- **Week 4** — Do mock interviews and finalize GitHub portfolio.



## 9. SALARY BENCHMARKS 2025

- **Junior AI Engineer** — \$90–120 K (US) | £60–80 K (UK)
- **Mid-level Engineer** — \$130–180 K (US) | £85–120 K (UK)
- **Senior Engineer** — \$200 K+ (US) | £130–180 K (UK)
- **AI Infra / MLOps** — \$180–250 K (US) | £120–160 K (UK)

Remote and hybrid roles often match major-city salary ranges due to global competition.

## 10. PORTFOLIO & LINKEDIN OPTIMIZATION

- Pin two AI projects (RAG + Agent) on GitHub.
- Include Dockerfiles and clear READMEs.
- Write one technical post (e.g. 'How I Built a LangChain App').
- Add 'AI Engineering Bootcamp @ Maven' to your education section.



## Conclusion & Next Steps

AI engineering interviews reward practical builders who can reason about design, scale, and reliability.

To move from preparation to mastery, join the [Maven AI Engineering Bootcamp](#) by Aki & Manu — where you'll actually build, deploy, and document the systems interviewers ask about.

[CLICK HERE TO LEARN MORE](#)



**BRING THIS INTO THE BOOTCAMP  
WE'LL HELP YOU HARDEN IT FOR PRODUCTION.**

BOOK A FREE 15-MIN STRATEGY CALL:

[SNAPDRUM.COM/CALL](https://snapdrum.com/call)