AI ENGINEER READY · STUDENT ROADMAP

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

Your 6-Project Roadmap

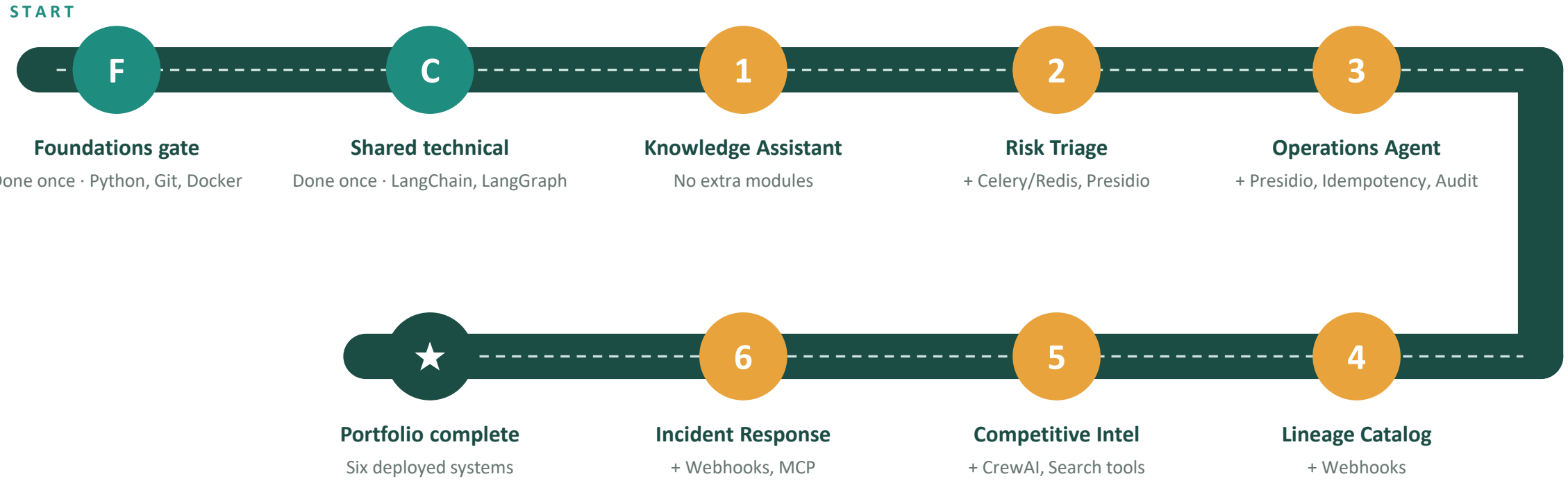
What you build in each project — and the exact path to get there, stage by stage.

1 **Foundations gate** → 2 **Shared technical** → 3 **Six projects** → 4 **Portfolio**

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

One Road: Foundations to Six Deployed Systems

Join any week · learn at your own pace · bring every doubt to the 3-hour Sunday live



! **ROADMAP RULE:** The Foundations gate and shared technical are done once. After that, each project only adds its own specialist modules.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

Foundations Gate: Everyone Starts Here

1 Python for AI Engineering

Pydantic, async, logging and testing — Python the way AI systems need it.

2 Git & GitHub Workflow

Branches, pull requests, and keeping API keys out of your history.

3 Docker & Compose

Package any app so it runs the same everywhere, plus multi-service stacks.

4 FastAPI

Wrap your AI in a clean, typed API that other systems can call.

5 LLM Fundamentals

Tokens, context windows, embeddings — and the three levers of cost.

6 Prompt Engineering

Clear instructions, few-shot examples, grounding, and fixing a prompt that drifts.

! **REMEMBER THIS:** Six modules, done once — tech or non-tech, everyone starts here, so nobody is left behind.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

Shared Technical: Learn Once, Reuse Six Times

01
Streamlit
Chat UIs, streaming, file upload

02
PostgreSQL
Schemas, SQLAlchemy, migrations

03
Structured Output
Pydantic schemas, repair-retry

04
LangChain Core
Models, LCEL, loaders, splitters

05
LangGraph Core
State, nodes, routing, checkpoints

06
System Design for AI
Timeouts, retries, rate limits

07
Evaluation
Faithfulness, golden sets, LLM-as-judge

08
LangSmith Tracing
Read traces, run eval datasets

09
CI/CD
GitHub Actions: test, build, protect

10
Azure Deployment
Container Apps, scaling, logs

! **REMEMBER THIS:** Every project uses this core — so each new project gets faster, not harder.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

1 Enterprise Knowledge Assistant

Ask a question in plain English and get an answer grounded in your company's own documents — with citations.

YOUR DATA `Confluence` `PDFs` `SharePoint` `Drive` `Slack threads`

AGENT 1

Ingest & Chunk

- Load every source format
- Split semantically, not by character count
- Keep source + page metadata

→

AGENT 2

Embed & Index

- Generate embeddings
- Store in a vector DB
- Build a hybrid keyword + vector index

→

AGENT 3

Retrieve & Rank

- Hybrid search over the index
- Rerank for actual relevance
- Assemble the context window

→

AGENT 4

Answer & Cite

- Generate a grounded answer
- Attach the source citation
- Flag it when confidence is low

WHAT YOU SHIP **Cited answer · Source links · Confidence score**

PROVES RAG · chunking · reranking · evaluation

! **WHY START HERE:** Every company has documents nobody can find. This is the most requested internal AI build right now.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

1 Enterprise Knowledge Assistant: Your Roadmap

1 — 2 — 3 — 4 — 5

STEP 1 · REQUIRED, DONE ONCE

Foundations gate

- Python for AI
- Git & GitHub
- Docker & Compose
- FastAPI
- LLM fundamentals
- Prompting

STEP 2 · REQUIRED, DONE ONCE

Shared technical

- Streamlit
- PostgreSQL
- Structured output
- LangChain core
- LangGraph core
- System design for AI
- Evaluation
- LangSmith tracing
- CI/CD
- Azure deployment

STEP 3 · THIS PROJECT

Specialist modules

None — you go straight into the build.

STEP 4 · PROJECT 1 BUILD

Build it, checkpoint by checkpoint

1. RAG, from first principles
2. Chunk, embed & store
3. Retrieval that actually works
4. Make it agentic
5. Evaluate, secure & deploy

STEP 5

Ship it

- Cited answer
- Source links
- Confidence score
- Deployed on Azure

! **MILESTONE:** A deployed assistant that answers from your documents, cites the page, and refuses when it doesn't know.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

2 Support Risk Triage

Read every incoming ticket, score how likely it is to blow up, and put the dangerous ones in front of a human first.

YOUR DATA

- Zendesk tickets
- Email threads
- CSAT scores
- Account tier

AGENT 1

Ingest & Normalise

- Pull tickets on a schedule
- Clean and de-duplicate text
- Attach the account context

→

AGENT 2

Classify & Score

- Detect intent and severity
- Score churn risk 0–100
- Tag the product area

→

AGENT 3

Detect Signals

- Sentiment dropping over a thread
- Repeat contacts on one issue
- SLA breach approaching

→

AGENT 4

Route & Alert

- Assign the right queue
- Notify the account owner
- Escalate above a threshold

WHAT YOU SHIP **Risk score per ticket · Priority queue · Escalation alerts**

PROVES Classification · scoring · workflow automation

! **WHY THIS SELLS:** It saves money on day one and the ROI is easy to prove — the easiest project to get a business to pay for.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

2 Support Risk Triage: Your Roadmap

1 — 2 — 3 — 4 — 5

STEP 1 · REQUIRED, DONE ONCE

Foundations gate

- Python for AI
- Git & GitHub
- Docker & Compose
- FastAPI
- LLM fundamentals
- Prompting

STEP 2 · REQUIRED, DONE ONCE

Shared technical

- Streamlit
- PostgreSQL
- Structured output
- LangChain core
- LangGraph core
- System design for AI
- Evaluation
- LangSmith tracing
- CI/CD
- Azure deployment

STEP 3 · THIS PROJECT

Specialist modules

- Celery & Redis
- PII with Presidio

STEP 4 · PROJECT 2 BUILD

Build it, checkpoint by checkpoint

1. Design the classifier
2. Rules first, model second
3. Confidence & cost
4. Context & a learning loop
5. Measure, monitor & deploy

STEP 5

Ship it

- Risk score per ticket
- Priority queue
- Escalation alerts
- Deployed on Azure

! **MILESTONE:** A triage system that scores every ticket, knows when to abstain, and reports its cost in rupees, not just accuracy.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

3 Support Operations Agent

Not just triage — this one drafts the actual resolution, checks itself, and hands off to a human when it isn't sure.

YOUR DATA `Live ticket` `KB articles` `Past resolutions` `Product docs`

AGENT 1

Understand Request

- Extract the real intent
- Pull out entities and IDs
- Spot what's missing from the ask

→

AGENT 2

Search & Gather

- RAG over the knowledge base
- Find how this was solved before
- Check current product docs

→

AGENT 3

Draft Resolution

- Compose the reply
- Cite the source it used
- Suggest the next actions

→

AGENT 4

Verify & Hand Off

- Check it against policy
- Gate on a confidence threshold
- Route to a human when unsure

WHAT YOU SHIP **Draft reply · Suggested actions · Human-in-the-loop gate**

PROVES Multi-step agents · tool use · guardrails

! **REMEMBER THIS:** This is the jump from a system that answers to a system that acts — exactly what an Agentic AI Engineer is hired for.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

3 Support Operations Agent: Your Roadmap

1 — 2 — 3 — 4 — 5

STEP 1 · REQUIRED, DONE ONCE

Foundations gate

- Python for AI
- Git & GitHub
- Docker & Compose
- FastAPI
- LLM fundamentals
- Prompting

STEP 2 · REQUIRED, DONE ONCE

Shared technical

- Streamlit
- PostgreSQL
- Structured output
- LangChain core
- LangGraph core
- System design for AI
- Evaluation
- LangSmith tracing
- CI/CD
- Azure deployment

STEP 3 · THIS PROJECT

Specialist modules

- PII with Presidio
- Idempotency
- Audit logging

STEP 4 · PROJECT 3 BUILD

Build it, checkpoint by checkpoint

1. Safe tool design
2. Human-in-the-loop
3. Policy & limits
4. Defend against attacks
5. Evaluate & deploy

STEP 5

Ship it

- Draft reply
- Suggested actions
- Human-in-the-loop gate
- Deployed on Azure

! **MILESTONE:** An agent that drafts real resolutions but can never take a risky action without a human saying yes.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

4 Data Lineage Catalog

Read the SQL nobody documented, work out what feeds what, and answer “what breaks if I change this column?”

YOUR DATA

- `SQL scripts`
- `dbt models`
- `ETL pipelines`
- `Warehouse metadata`

AGENT 1

Parse Code

- Read SQL and dbt models
- Extract tables and columns
- Identify each transform step

→

AGENT 2

Map Dependencies

- Build the upstream graph
- Build the downstream graph
- Resolve aliases and CTEs

→

AGENT 3

Enrich & Describe

- Auto-document each table
- Infer the owner and freshness
- Flag orphaned and stale tables

→

AGENT 4

Answer Impact Qs

- “What breaks if I drop this?”
- Trace a column end to end
- Rank blast radius

WHAT YOU SHIP **Lineage graph · Searchable catalog · Impact analysis**

PROVES Code parsing · graphs · domain depth

! **WHY IT STANDS OUT:** Almost nobody builds this in a portfolio. It shows you understand real data platforms, not just chatbots.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

4 Data Lineage Catalog: Your Roadmap

1 — 2 — 3 — 4 — 5

STEP 1 · REQUIRED, DONE ONCE

Foundations gate

- Python for AI
- Git & GitHub
- Docker & Compose
- FastAPI
- LLM fundamentals
- Prompting

STEP 2 · REQUIRED, DONE ONCE

Shared technical

- Streamlit
- PostgreSQL
- Structured output
- LangChain core
- LangGraph core
- System design for AI
- Evaluation
- LangSmith tracing
- CI/CD
- Azure deployment

STEP 3 · THIS PROJECT

Specialist modules

- Webhooks & events

STEP 4 · PROJECT 4 BUILD

Build it, checkpoint by checkpoint

1. Understand lineage
2. Parse before you prompt
3. Cut cost, survive limits
4. Keep data safe
5. Graph, validate & deploy

STEP 5

Ship it

- Lineage graph
- Searchable catalog
- Impact analysis
- Deployed on Azure

! **MILESTONE:** A catalog that answers “what breaks if I change this?” — with a before-and-after cost table to prove it's affordable.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

5 Competitive Intelligence

Watch what your competitors publish, work out how they position themselves, and find the gap you can own.

YOUR DATA

- Competitor sites
- Pricing pages
- Review sites
- Job posts
- News

AGENT 1

Collect Sources

- Crawl on a schedule
- De-duplicate and timestamp
- Snapshot what changed

→

AGENT 2

Extract Claims

- Pull positioning and messaging
- Capture pricing and packaging
- Identify the target segment

→

AGENT 3

Compare & Find Gaps

- Diff against your own position
- Spot claims nobody is making
- Track shifts over time

→

AGENT 4

Generate Battlecard

- Where you win, where you lose
- Objection handling lines
- Alert on a real change

WHAT YOU SHIP Battlecard · Positioning gap list · Change alerts

PROVES Web agents · scheduling · structured extraction

! **WHY IT'S EASY TO SELL:** The buyer is marketing or sales, not engineering — and non-technical buyers approve budget faster.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

5 Competitive Intelligence: Your Roadmap

1 — 2 — 3 — 4 — 5

STEP 1 · REQUIRED, DONE ONCE

Foundations gate

- Python for AI
- Git & GitHub
- Docker & Compose
- FastAPI
- LLM fundamentals
- Prompting

STEP 2 · REQUIRED, DONE ONCE

Shared technical

- Streamlit
- PostgreSQL
- Structured output
- LangChain core
- LangGraph core
- System design for AI
- Evaluation
- LangSmith tracing
- CI/CD
- Azure deployment

STEP 3 · THIS PROJECT

Specialist modules

- CrewAI
- Web-search tools

STEP 4 · PROJECT 5 BUILD

Build it, checkpoint by checkpoint

1. When multi-agent is worth it
2. Fan-out & failure
3. Plan approval & cost control
4. Trustworthy research
5. Evaluate & deploy

STEP 5

Ship it

- Battlecard
- Positioning gap list
- Change alerts
- Deployed on Azure

! **MILESTONE:** A research crew that writes a sourced battlecard — and a report that honestly says when the simpler version won.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

6 Incident Response Assistant

When production breaks at 2am, gather the signals, rank the likely causes, and draft the postmortem afterwards.

YOUR DATA

`Alerts` `Logs` `Metrics` `Deploy history` `Runbooks`

AGENT 1

Gather Signals

- Pull the firing alerts
- Fetch recent deploys
- Collect error-rate spikes

→

AGENT 2

Correlate

- Link symptoms across services
- Build the incident timeline
- Separate cause from effect

→

AGENT 3

Diagnose

- Rank the probable causes
- Pull the matching runbook
- Suggest the next check to run

→

AGENT 4

Draft Postmortem

- Write the timeline
- State the root cause
- List the action items

WHAT YOU SHIP **Probable root cause · Runbook steps · Draft postmortem**

PROVES Correlation · reasoning · production judgement

! **REMEMBER THIS:** This is the one that convinces a hiring manager you've actually run something in production, not just built a demo.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

6 Incident Response Assistant: Your Roadmap

1 — 2 — 3 — 4 — 5

STEP 1 · REQUIRED, DONE ONCE

Foundations gate

- Python for AI
- Git & GitHub
- Docker & Compose
- FastAPI
- LLM fundamentals
- Prompting

STEP 2 · REQUIRED, DONE ONCE

Shared technical

- Streamlit
- PostgreSQL
- Structured output
- LangChain core
- LangGraph core
- System design for AI
- Evaluation
- LangSmith tracing
- CI/CD
- Azure deployment (optional)

STEP 3 · THIS PROJECT

Specialist modules

- Webhooks & events
- MCP

STEP 4 · PROJECT 6 BUILD

Build it, checkpoint by checkpoint

1. Frame the 2am problem
2. Isolated agents via MCP
3. Correlate & diagnose
4. Fail honestly
5. Test, evaluate & deploy

STEP 5

Ship it

- Probable root cause
- Runbook steps
- Draft postmortem
- Runs locally; Azure optional

! **MILESTONE:** An assistant that investigates like a calm senior engineer — ranked causes, honest gaps, and a postmortem draft.

LEARN WITH SARVESH
COURSES. CLARITY. CONFIDENCE.

Your Six Projects, Side by Side

#	Project	New modules	You ship	Proves
1	**Enterprise Knowledge Assistant**	—	Cited answer · Source links · Confidence score	RAG · chunking · reranking · evaluation
2	**Support Risk Triage**	Celery & Redis, PII with Presidio	Risk score per ticket · Priority queue · Escalation alerts	Classification · scoring · workflow automation
3	**Support Operations Agent**	PII with Presidio, Idempotency, Audit logging	Draft reply · Suggested actions · Human-in-the-loop gate	Multi-step agents · tool use · guardrails
4	**Data Lineage Catalog**	Webhooks & events	Lineage graph · Searchable catalog · Impact analysis	Code parsing · graphs · domain depth
5	**Competitive Intelligence**	CrewAI, Web-search tools	Battlecard · Positioning gap list · Change alerts	Web agents · scheduling · structured extraction
6	**Incident Response Assistant**	Webhooks & events, MCP	Probable root cause · Runbook steps · Draft postmortem	Correlation · reasoning · production judgement

! **GOOD TO KNOW:** A new project starts every 6 weeks and every one comes round again — so you can join any week.

LEARN WITH SARVESH

COURSES. CLARITY. CONFIDENCE.

Ready to Start Your Road?

Join any week at **ai-ready.learnwithsarvesh.com**

Questions? WhatsApp **+91 99655 88321**

Learn with Sarvesh · Courses. Clarity. Confidence.