AI PRODUCT MANAGEMENT 2 · DETAILED COURSE

AI Product Architecture and Development

How AI features are built, served and operated, and how a PM makes build, buy and partner decisions.

Chapter 16 of 26 · 26 topics · Edition 2026 · skillnavora.in

CHAPTER 16

AI Product Architecture and Development

How AI features are built, served and operated, and how a PM makes build, buy and partner decisions.

AI Product Architecture Fundamentals

The layers of an AI product: data, models, orchestration, application, monitoring and governance.

An AI product has layers: data, models, orchestration, application, monitoring and governance. A weakness in any layer limits the whole product, so review all layers, not only the model.

AI Application Architecture

How the interface, backend, model calls, data stores and tools fit together.

The interface, backend, model calls, data stores and tools fit together through defined APIs. Draw this picture early so engineers and PMs agree on boundaries.

LLM Application Architecture

Prompt assembly, retrieval, model call, tool execution, validation and response handling.

An LLM request usually assembles a prompt, retrieves context, calls the model, runs any tools, validates the output and returns the response. Each step adds latency and failure points.

Model Integration

Connecting models to the product through APIs or SDKs, with timeouts, retries and fallbacks.

Connect models through APIs or SDKs with timeouts, retries and fallbacks. Plan for provider outages, because a model dependency is an external service.

AI APIs

Hosted endpoints for models, speech, vision and embeddings, billed per use.

Hosted endpoints provide models for text, speech, vision and embeddings, billed per use. They remove infrastructure work but add cost per request and dependence on the provider.

Model Hosting

Running models on a provider's servers or your own infrastructure.

Hosting runs models on a provider's servers or your own infrastructure. Self-hosting gives control over data and cost at scale, but needs machine learning operations skills.

Model Serving

Systems that load models and answer requests efficiently with batching and scaling.

Serving systems load models and answer requests efficiently, using batching and scaling. Serving is where latency and cost targets are met or missed.

Inference Endpoints

Network endpoints that return predictions or generations on request.

Endpoints return predictions or generations on request. Monitor their latency, errors and cost separately from the rest of the application.

GPU Fundamentals

Parallel processors used to train and run models; memory size and availability drive cost and capacity.

GPUs run the parallel calculations behind training and inference. Memory size and availability drive both cost and capacity, so plan capacity with engineers.

AI Data Pipelines

Automated flows that collect, clean, label and deliver data for training, retrieval and monitoring.

Pipelines collect, clean, label and deliver data for training, retrieval and monitoring. Pipelines run continuously, so their failures can silently degrade quality.

ETL and ELT

Extract-transform-load transforms before loading; ELT loads first and transforms in the warehouse.

ETL transforms data before loading it into a store. ELT loads raw data first and transforms it in the warehouse. Modern analytics stacks often prefer ELT.

AI Application Databases

Operational stores for users, sessions, prompts, outputs and feedback.

Operational stores hold users, sessions, prompts, outputs and feedback. Feedback stored here feeds later evaluation and improvement.

Vector Database Integration

Embedding content, indexing it and querying by similarity within the application.

Embed content, index it and query by similarity inside the application. Keep the index in sync with source content and respect permissions at query time.

AI Orchestration Layers

Code or frameworks that sequence model calls, retrieval and tools.

Code or frameworks sequence model calls, retrieval and tools. A clear orchestration layer makes behaviour easier to test and change.

Microservices for AI

Separating model, retrieval and business services so each scales and deploys independently.

Separating model, retrieval and business services lets each scale and deploy independently. The cost is more network calls and more operational work.

Cloud AI Infrastructure

Managed platforms for training, hosting and monitoring models, such as Bedrock, Azure AI Foundry and Vertex AI.

Managed platforms for training, hosting and monitoring models include Amazon Bedrock, Azure AI Foundry and Google Vertex AI. They reduce operational work and tie you to a provider's services.

AI Application Scalability

Handling rising load with queues, caching, rate limits, batching and autoscaling.

Scale with queues, caching, rate limits, batching and autoscaling. Rate limits protect the product and the provider budget during traffic spikes.

Model Deployment

Releasing a model or prompt change to production with testing, staged rollout and rollback.

Release a model or prompt change to production with testing, staged rollout and rollback. Treat it like any software release with monitoring after each stage.

Model Versioning

Recording model, prompt and data versions so results are reproducible and regressions traceable.

Record model, prompt and data versions so results can be reproduced and regressions traced. Without versions, you cannot explain why behaviour changed.

MLOps

Practices for training, deploying, monitoring and retraining machine-learning models reliably.

MLOps covers training, deploying, monitoring and retraining machine learning models reliably. It applies software engineering discipline to models.

LLMOps

MLOps adapted to LLM apps: prompt and eval management, tracing, cost control and guardrails.

LLMOps adapts MLOps to LLM applications, adding prompt and evaluation management, tracing, cost control and guardrails.

CI/CD for AI

Automated tests and eval gates that run on every change to code, prompts or models.

Automated tests and evaluation gates run on every change to code, prompts or models. A change that lowers the quality score is blocked before release.

AI System Monitoring

Tracking quality, latency, errors, cost, drift and safety signals in production.

Track quality, latency, errors, cost, drift and safety signals in production. Set alerts on changes, not only on outages.

AI Application Reliability

Keeping the product working despite model outages and variance, via fallbacks, timeouts and degraded modes.

Keep the product working despite model outages and variable outputs, using fallbacks, timeouts and degraded modes. A degraded mode might show search results when the AI answer is unavailable.

Production AI Architecture

A complete design including security, observability, evaluation, cost controls and operational runbooks.

A production design includes security, observability, evaluation, cost controls and operational runbooks. Runbooks say what to do when something goes wrong at 2 a.m.

Build vs Buy vs Partner Decisions

Build where it differentiates you; buy for commodity capability; partner for speed or expertise. Weigh cost, control, time and risk.

Build where the capability differentiates you, buy commodity capability, and partner for speed or expertise. Weigh cost, control over data, time to market and risk.

Summary of terms

Term	Meaning
AI Product Architecture Fundamentals	The layers of an AI product: data, models, orchestration, application, monitoring and governance.
AI Application Architecture	How the interface, backend, model calls, data stores and tools fit together.
LLM Application Architecture	Prompt assembly, retrieval, model call, tool execution, validation and response handling.
Model Integration	Connecting models to the product through APIs or SDKs, with timeouts, retries and fallbacks.
AI APIs	Hosted endpoints for models, speech, vision and embeddings, billed per use.
Model Hosting	Running models on a provider's servers or your own infrastructure.
Model Serving	Systems that load models and answer requests efficiently with batching and scaling.
Inference Endpoints	Network endpoints that return predictions or generations on request.
GPU Fundamentals	Parallel processors used to train and run models; memory size and availability drive cost and capacity.
AI Data Pipelines	Automated flows that collect, clean, label and deliver data for training, retrieval and monitoring.
ETL and ELT	Extract-transform-load transforms before loading; ELT loads first and transforms in the warehouse.
AI Application Databases	Operational stores for users, sessions, prompts, outputs and feedback.
Vector Database Integration	Embedding content, indexing it and querying by similarity within the application.

AI Orchestration Layers	Code or frameworks that sequence model calls, retrieval and tools.
Microservices for AI	Separating model, retrieval and business services so each scales and deploys independently.
Cloud AI Infrastructure	Managed platforms for training, hosting and monitoring models, such as Bedrock, Azure AI Foundry and Vertex AI.
AI Application Scalability	Handling rising load with queues, caching, rate limits, batching and autoscaling.
Model Deployment	Releasing a model or prompt change to production with testing, staged rollout and rollback.
Model Versioning	Recording model, prompt and data versions so results are reproducible and regressions traceable.
MLOps	Practices for training, deploying, monitoring and retraining machine-learning models reliably.
LLMOps	MLOps adapted to LLM apps: prompt and eval management, tracing, cost control and guardrails.
CI/CD for AI	Automated tests and eval gates that run on every change to code, prompts or models.
AI System Monitoring	Tracking quality, latency, errors, cost, drift and safety signals in production.
AI Application Reliability	Keeping the product working despite model outages and variance, via fallbacks, timeouts and degraded modes.
Production AI Architecture	A complete design including security, observability, evaluation, cost controls and operational runbooks.
Build vs Buy vs Partner Decisions	Build where it differentiates you; buy for commodity capability; partner for speed or expertise. Weigh cost, control, time and risk.

Practice exercise: Draw the architecture of an AI feature with model, retrieval, database, monitoring and fallback. Write the build, buy or partner decision for each part.

Self-check

1. Explain AI Product Architecture Fundamentals in two sentences to a non-technical stakeholder, then say what you would measure for it.
2. Explain AI Orchestration Layers in two sentences to a non-technical stakeholder, then say what you would measure for it.
3. Explain Build vs Buy vs Partner Decisions in two sentences to a non-technical stakeholder, then say what you would measure for it.

Related SkillNavora material

- System Design and Databases for PMs : Architecture, databases and trade-offs.
- AI Tutor : Ask questions on any topic.
- Study Library : Every lesson, class and PDF by programme.

© SkillNavora · skillnavora.in · Facts and tools change; check official sites before relying on a figure.

ORIGINAL CONTENT

Original AI Product Management content

The AI Product Management programme already on SkillNavora, included unchanged so the full course is in one place.

Quick links Study Library (all material) Learning sites for PM AI builder tools (how-to) Webhooks & Playwright MCP Webhook URLs and keys (start here) AI Skills Academy AI Skills Map Context engineering (PDF) AI skills & templates (PDF) Orchestration (PDF) PRD library (120 ideas) AI PRD Writer AI PM Toolkit AI Evals (PDF) Browser automation (PDF) AI second brain (PDF) PM handbook Top 100 PM questions AI product sense Q&A (86) PM practice test Practice Lab Tool tutorials AI Agents & MCP for PMs

PRD documents

PRD library: template, framing guide and 120 PRD ideas

Learn how top PMs frame a PRD, download the free template, and turn any of 120 ideas across 15 industries into a full PRD with AI.

Open the PRD library ✨ AI PRD Writer PRD template (Word) AI Evals for PMs (PDF)

Context window

Context window: what a model can see at once

The context window is the most tokens a model can consider in one request. It covers your prompt, any documents you paste, the conversation so far and the reply. Anything outside it is invisible to the model. For PMs: a longer window fits more material but costs more and can slow responses; put the most important facts near the instructions; use retrieval (RAG) rather than pasting whole libraries; and test answers on long inputs, where quality can drop.

Context engineering course Context windows in the AI PM 2 course

Tokens and context in the Grokbot course

AI-First Product Management

Resources for Product Managers

The PM Master Handbook, AI and product tools with their official websites, interview practice, a frameworks cheat sheet, a PRD template and free courses.

Open the PM handbook library: all 68 PDFs ↗ PM Glossary Handbook Tokens, Pricing and Tokenization Vibe Coding addendum

PM Master Handbook by SkillNavora
23 modules and 620+ pages: 12 Product Core modules, an 11-class AI Tools Lab, frameworks, case studies, projects and interview prep.

Open the interactive handbook ↗ Complete handbook PDF (620+ pages, 23 MB) ↗

Product Core modules

1Fundamentals of Product Management*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
2Product Strategy and Strategic Thinking*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
3Problem Discovery and Market Research*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
4Problem Framing and Prioritization*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
5UX and Design Collaboration*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
6Effective Communication and Stakeholder Management*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
7Product Analytics and Metrics*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
8Product Launch, Growth, and Monetization*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
9Tech 101 and System Design for PMs*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
10LLMs, RAGs, and AI Foundations*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
11Vibe Coding for Product Managers*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides
12Agentic AI and AI Systems*Handbook chapter, detailed PDF, handbook and class slides*PDF Detailed PDF Handbook Class Slides

AI Tools Lab

13AI Tools Landscape and Your Starter StackPDF ↗ 14Claude for Daily PM WorkPDF ↗ 15Claude Code: Basic to AdvancedPDF ↗ 16Lovable: Basic to AdvancedPDF ↗ 17Replit and Other AI App BuildersPDF ↗ 18Figma and AI Design ToolsPDF ↗ 19Notion, PRD Tools and AI DocumentationPDF ↗ 20Automation with Zapier, Make and n8nPDF ↗ 21AI Chatbots for BusinessPDF ↗ 22Voice Agents for BusinessesPDF ↗ 23Learning Paths, Courses and CommunitiesPDF ↗

Business AI Stack (chapters 24 to 31)

24The business AI stack: tools and where to learn themPDF ↗ 25Next.js: customer-facing dashboardsPDF ↗ 26Zoho CRM: managing customers and leadsPDF ↗ 27WhatsApp Business Platform: customer messagingPDF ↗ 28Deployment with Vercel and AWSPDF ↗ 29Jira: tracking product and development workPDF ↗ 30Meta Business Suite, social media marketing and Google AdsPDF ↗ 31A one-page website that generates leadsPDF ↗

Complete AI tool guides (15)

1n8n: workflows, Gmail and an AI reply agentPDF ↗ 2Make: scenarios, routers and lead routingPDF ↗ 3Zapier: invoices from Gmail to Drive and a sheetPDF ↗ 4Webhooks: how apps talk in real timePDF ↗ 5Replit: one app for the web and a phonePDF ↗ 6Lovable: a job application trackerPDF ↗ 7Figma: components, auto layout and handoffPDF ↗ 8Claude Code: an AI coding agent in your projectPDF ↗ 9MCP: connecting AI assistants to your toolsPDF ↗ 10LLMs: how they work and how to call themPDF ↗ 11RAG: answers from your own documentsPDF ↗ 12AI agents: a job-search agent with approvalsPDF

↗ 13Agentic AI: planning, guardrails and teamsPDF ↗ 14Generative AI: uses, limits and a weekly reportPDF ↗ 15Other AI tools: choosing one per jobPDF ↗

Complete AI tool guides (one PDF, 30 pages) ↗

Libraries

★Framework Library*29 frameworks with worked examples and templates*PDF ↗ ★Case Study Library*12 product case studies*PDF ↗ ★Practical Projects and Portfolio*5 end-to-end portfolio projects*PDF ↗ ★Interview Handbook*100+ PM interview questions with model answers*PDF ↗ ★AI Tools Lab*Every AI tools class with step-by-step labs*PDF ↗

PM interview practice (free, on the original sites)

Hand-picked public question banks and guides from other PM communities. Each link opens the original page in a new tab.

PM frameworks cheat sheet

Framework	Use it to	How it works
RICE	Prioritise a backlog	Reach × Impact × Confidence ÷ Effort. Score each idea, then sanity-check the ranking.
Kano	Decide which features delight	Classify features as Must-be, Performance, Attractive, Indifferent or Reverse using a short survey.
MoSCoW	Scope a release	Must have, Should have, Could have, Won't have (this time).
Jobs-to-be-done	Understand motivation	When [situation], I want to [motivation], so I can [outcome].
AARRR (Pirate metrics)	Map the growth funnel	Acquisition, Activation, Retention, Referral, Revenue.
HEART	Measure user experience	Happiness, Engagement, Adoption, Retention, Task success, each with goals, signals and metrics.
North Star metric	Align the team	One metric that captures the value users get, supported by input metrics the team can move.
Five whys	Find the root cause	Ask 'why?' repeatedly until you reach a cause you can act on.
CIRCLES	Answer product design questions	Comprehend, Identify users, Report needs, Cut through prioritisation, List solutions, Evaluate trade-offs, Summarise.

PRD template

PRD: [Feature name] Owner: [Name] · Status: Draft · Last updated: [Date] 1. Problem Who has the problem, what it is, and the evidence (data, interviews, tickets). 2. Goal and success metrics Primary metric (with baseline and target), secondary metrics, guardrails. 3. Users and use cases Target segment, top 3 user stories: "As a [user], I want [action] so that [benefit]." 4. Solution What we'll build (link designs), what's out of scope, open questions. 5. AI behaviour (if applicable) Inputs, model, prompt/approach, what it must never do, human review, evaluation set and pass criteria, cost per request. 6. Launch plan Experiment design, rollout stages, support and comms, rollback criteria. 7. Risks and dependencies Technical, legal/privacy, operational; owners and mitigations.

Free courses