

SKILLNAVORA · AI PRODUCT MANAGEMENT & GENAI
LAUNCHPAD

Platform & API Product Management

Manage platforms and APIs as products: developer experience, API design decisions, versioning and limits, ecosystems and partners, platform metrics, and AI/MCP platforms.

Edition 2026 · skillnavora.in

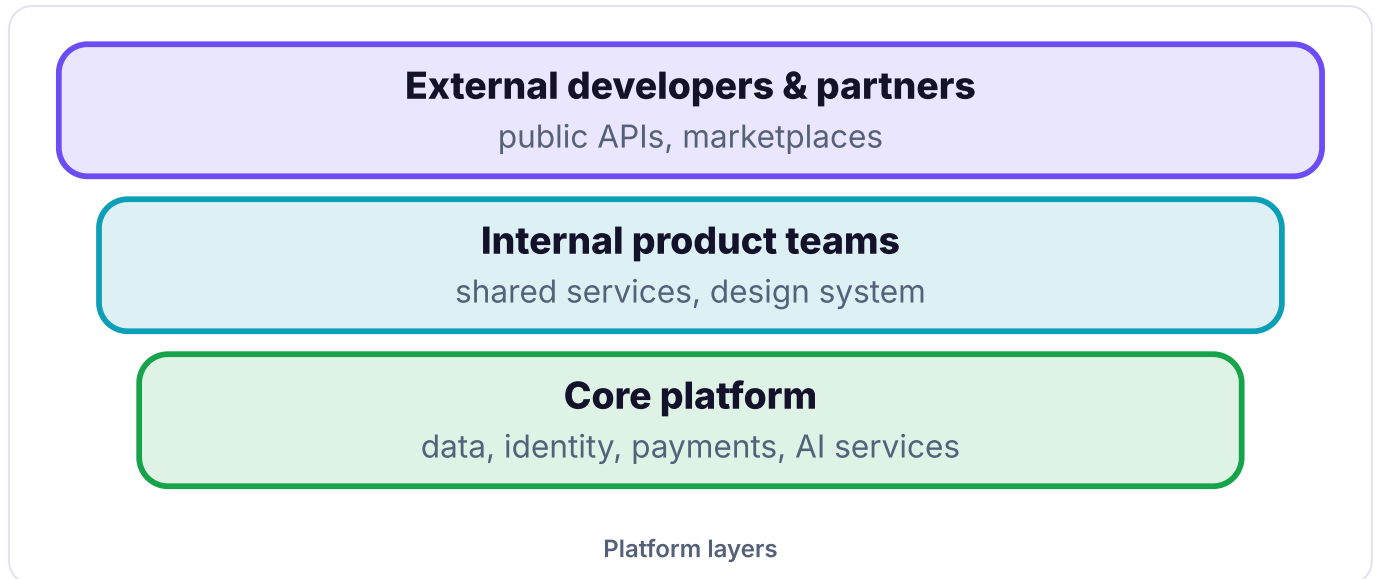
How to use this guide

Each class has goals, concepts with Indian examples, key takeaways and a practice exercise, from basic to advanced. Finish with the final project to build something you can show in interviews.

	Class	Level
Class 1	Platforms and APIs as products	Basic
Class 2	Developer experience (DX)	Basic
Class 3	API design decisions for PMs	Intermediate
Class 4	Ecosystems and partners	Intermediate
Class 5	Platform metrics	Advanced
Class 6	AI platforms and final project	Advanced
Toolkit: templates, cheat sheet, glossary, 5 interview questions, further reading		

Platforms and APIs as products

Understand platform types and users.



Examples: UPI/payment APIs (Razorpay, Stripe), maps APIs, WhatsApp Business API, internal identity or AI platforms, MCP servers that expose your product to AI assistants.

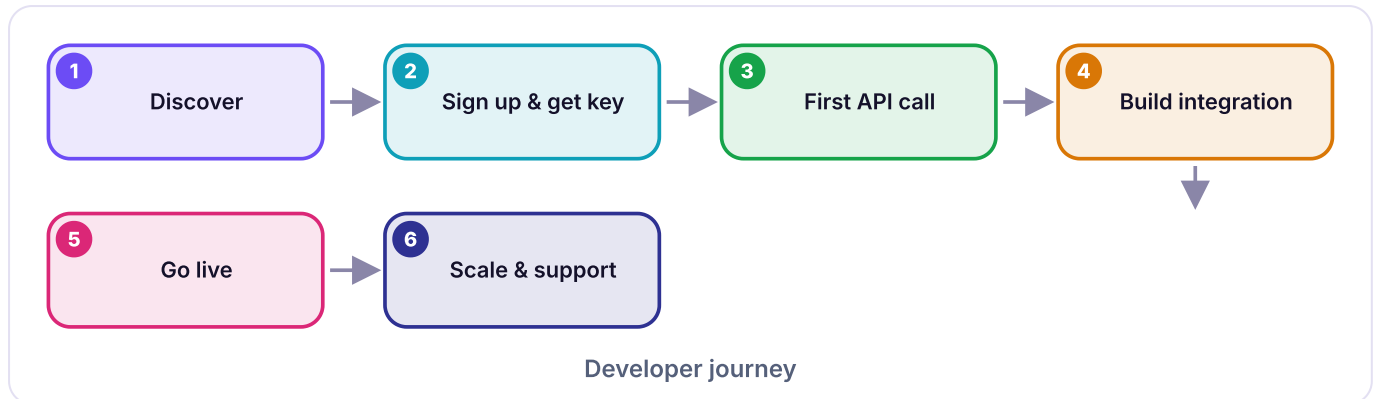
Key takeaways

- Developers are the users.
- Internal platforms are products too.
- AI assistants are a new channel (MCP).

Practice exercise: Map the platform users and their jobs for a payments company.

Developer experience (DX)

Make developers successful fast.



- Time to first successful call (aim for minutes).
- Clear docs, quickstarts, SDKs, sandbox keys, Postman collections.
- Helpful errors, status page, changelog, community support.

Key takeaways

- Measure time to first call.
- Docs and sandbox are the onboarding.
- Errors should explain the fix.

Practice exercise: Audit a public API's onboarding and list 5 improvements.

API design decisions for PMs

Decide what matters in API contracts.

- Resources and endpoints that match developer jobs; consistent naming.
- Authentication (API keys, OAuth), permissions and scopes.
- Versioning and deprecation policy (e.g. 12 months notice).
- Rate limits and quotas; idempotency keys for payments.
- Webhooks for events; pagination; error formats.

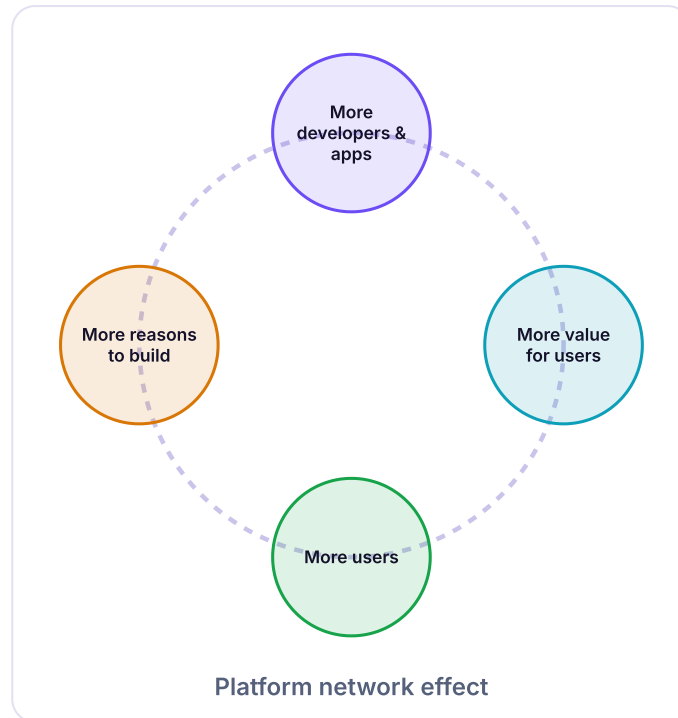
Key takeaways

- Contracts are promises: version carefully.
- Rate limits protect reliability.
- Webhooks make integrations real-time.

Practice exercise: Write the API requirements for a 'refund' endpoint.

Ecosystems and partners

Grow a platform through others.



- Partner tiers, certification, marketplace listing, revenue share.
- Governance: app review, security, data access rules.

Key takeaways

- Platforms grow through network effects.
- Partners need clear rules and incentives.
- Review apps for security.

Practice exercise: Design a partner program for a SaaS marketplace.

Platform metrics

Measure platform health.

Metric	Why
Active developers / apps	Adoption
Time to first call / to production	DX quality
API calls, success rate, latency	Usage and reliability
Revenue via platform	Business value
Support tickets per integration	Friction

Key takeaways

- Mix adoption, DX, reliability and revenue.
- Track success rate and latency.
- Use tickets to find friction.

Practice exercise: Build a platform dashboard spec.

AI platforms and final project

Apply platform thinking to AI.

- Internal AI platforms: shared model access, evals, guardrails, prompt/skill libraries, cost tracking.
- Exposing your product via MCP so AI assistants can use it.

Final project

Write a platform PRD for a public API or internal AI platform: users, developer journey, API requirements, limits and versioning, partner rules, metrics.

Key takeaways

- AI platforms centralise models, evals and guardrails.
- MCP is a new distribution channel.
- Treat developers like customers.

Practice exercise: Complete the platform PRD.

Templates

Platform PRD extras

Field	Fill in
Developer personas	
Jobs & use cases	
Time-to-first-call target	
Endpoints / events	
Auth & scopes	
Limits & versioning	
Partner rules	
Metrics	

Cheat sheet

Term / tool	Meaning / use
DX	Developer experience
SDK	Software development kit
Webhook	Event notification sent to your URL
Rate limit	Max requests per time
Deprecation	Planned retirement of an API version

Glossary

Idempotency key	Prevents duplicate actions on retries
OAuth	Standard for delegated access

5 interview questions with answers

Q1. What does a platform PM do?

Builds products other teams or developers build on, optimising developer success, reliability and ecosystem growth.

Q2. Key DX metric?

Time to first successful API call, plus time to production.

Q3. How do you version an API?

Explicit versions, backward compatibility, deprecation notices, migration guides and changelogs.

Q4. Platform metrics?

Active developers/apps, success rate, latency, calls, revenue, support friction.

Q5. What is an AI platform team?

Provides shared model access, evals, guardrails, prompt/skill libraries and cost controls to product teams.

Further reading and tools

- Stripe API docs (example of great DX): <https://docs.stripe.com/api>
- Razorpay docs: <https://razorpay.com/docs/>
- Postman learning center: <https://learning.postman.com/>
- Model Context Protocol: <https://modelcontextprotocol.io/>

© SkillNavora · skillnavora.in · Part of the AI Product Management and GenAI Launchpad programs. Practise with the free AI Tutor, AI PM Toolkit and Practice Lab on skillnavora.in.