AI PRODUCT MANAGEMENT 2 · DETAILED COURSE

Agile, Scrum and Product Delivery

How teams plan, build and ship iteratively, and the tools PMs use to coordinate delivery.

Chapter 06 of 26 · 23 topics · Edition 2026 · skillnavora.in

CHAPTER 06

Agile, Scrum and Product Delivery

How teams plan, build and ship iteratively, and the tools PMs use to coordinate delivery.

Agile Fundamentals

Iterative delivery, frequent feedback and adapting to change over following a fixed plan.

Agile delivers work in short iterations, gathers feedback quickly and adapts plans to what is learned. It replaces long fixed plans with frequent checkpoints. The PM's job is to keep the backlog ordered by outcome value.

Agile Manifesto

Four values and twelve principles favouring individuals, working software, collaboration and responding to change.

The manifesto values individuals and interactions, working software, customer collaboration and responding to change. The twelve principles add practical guidance such as frequent delivery and sustainable pace. Use the values as a test for practices, not as a rulebook.

Scrum Framework

Fixed-length sprints with defined roles, events and artefacts to deliver increments.

Scrum organises work into fixed-length sprints, with defined roles, events and artefacts. Each sprint produces an increment of working product. Teams adopt Scrum to create a predictable rhythm of planning, review and improvement.

Kanban Methodology

A continuous flow board that limits work in progress and optimises throughput.

Kanban visualises work on a board, limits work in progress and measures flow. It suits support and operations work with unpredictable arrivals. Limits prevent teams from starting too much and finishing nothing.

Scrum Roles and Responsibilities

Product Owner, Scrum Master and Developers, each accountable for a distinct part of the work.

The product owner orders the backlog and maximises value. The Scrum master removes impediments and coaches the process. Developers build the increment. Each role has a distinct accountability, and blurring them is a common failure.

Product Owner Responsibilities

Maximising product value by managing and ordering the backlog and accepting work.

The product owner manages the backlog, writes clear items, accepts work against criteria and represents customers to the team. A product owner who is absent from daily decisions cannot steer the team well.

Sprint Planning

The team selects backlog items and plans how to deliver them within the sprint goal.

The team selects backlog items and agrees a sprint goal. Planning answers what can be delivered and how. A good sprint goal describes the outcome, so the team can adapt scope without losing purpose.

Daily Stand-ups

A 15-minute sync on progress toward the sprint goal and blockers.

The daily stand-up is a 15-minute check on progress toward the sprint goal and on blockers. It is not a status report to the manager. Use it to remove blockers quickly, for example by escalating a waiting API decision the same day.

Sprint Reviews

Demonstrating the increment to stakeholders and gathering feedback.

The sprint review demonstrates the working increment to stakeholders and gathers feedback. It is the best place to show real software, not slides. Feedback here should change the backlog order.

Sprint Retrospectives

The team reflects on how it worked and agrees small improvements.

The retrospective examines how the team worked and agrees small, specific improvements. Keep one or two actions per sprint and track whether they helped. Retrospectives fail when they produce long lists nobody follows.

Backlog Grooming

Older term for refinement: keeping the backlog clear, sized and ordered.

Backlog grooming is the older term for refinement: keeping items clear, sized and ordered. Groom continuously rather than in one large session, so the team is never starting from scratch.

Story Points

A relative unit of effort and complexity for sizing stories, not hours.

Story points measure relative size and complexity, not hours. A story of 5 points should feel about five times the effort of a 1-point story. Use points for forecasting, and do not compare teams by points.

Estimation Techniques

Planning poker, t-shirt sizing, affinity estimation and reference-class estimates to forecast work.

Planning poker, t-shirt sizing and affinity estimation provide quick estimates. Reference-class estimates compare work to similar past items. Estimates are forecasts with error bars, so communicate ranges.

Velocity and Burndown Charts

Velocity is points completed per sprint; burndown shows remaining work over time. Use for forecasting, not for judging people.

Velocity is the points completed per sprint, and burndown shows remaining work over time. They forecast delivery. Using them to judge individuals damages trust and distorts estimates.

Release Planning

Sequencing sprints into releases with dates, scope, dependencies and risk buffers; staged rollout and rollback plans belong here.

Release planning sequences sprints into releases with scope, dates, dependencies and risk buffers. Staged rollouts such as 5%, 25% and 100% of users, with rollback plans, belong here. A release is ready when its go/no-go criteria are met.

Dependency Management

Identifying and tracking work that relies on other teams, systems or vendors, and resolving early.

A dependency is work that relies on another team, system or vendor. List dependencies early, agree dates with those teams and track them in each sprint review. Hidden dependencies cause most missed release dates.

Risk and Issue Management

Risks are possible problems to mitigate; issues are problems that have occurred and need action.

A risk is a possible problem you can prevent or reduce. An issue is a problem that has already happened and needs action. Keep a short risk register with owners, likelihood, impact and the mitigation date.

Cross-Functional Team Collaboration

Product, design, engineering, data, QA and go-to-market working toward one goal with shared context.

Product, design, engineering, data, QA and go-to-market work toward one goal with shared context. Shared goals and regular demos prevent the handoff delays of separate silos.

Engineering Collaboration

Working with engineers on trade-offs, feasibility and scope, treating them as partners from discovery onward.

Involve engineers from discovery onward. They know the cost of options and can suggest simpler solutions. Treat trade-offs on scope and quality as joint decisions.

Product Delivery Lifecycle

Idea to discovery, design, build, test, release, measure and iterate.

The lifecycle runs from idea to discovery, design, build, test, release, measurement and iteration. Measurement closes the loop: the results decide whether to invest more, change direction or stop.

Jira

Issue and sprint tracking tool for backlogs, boards, workflows and reports.

Jira tracks issues, sprints, boards and workflows. Keep the workflow simple, with few statuses, and use a clear naming and labelling convention. Reports are useful only if the data is kept current.

Confluence

Team wiki used for specs, decision records and knowledge sharing, integrated with Jira.

Confluence is a team wiki for specifications, decision records and knowledge. Link pages to Jira issues so context stays connected to the work. Archive old pages so people can find current ones.

Linear and Azure DevOps

Linear is a fast, opinionated issue tracker; Azure DevOps combines boards, repos and pipelines for engineering teams.

Linear is a fast, opinionated issue tracker popular with product teams. Azure DevOps combines boards, repositories and pipelines for engineering organisations. Choose the tool that fits how your engineering team already works.

Summary of terms

Term	Meaning
Agile Fundamentals	Iterative delivery, frequent feedback and adapting to change over following a fixed plan.
Agile Manifesto	Four values and twelve principles favouring individuals, working software, collaboration and responding to change.
Scrum Framework	Fixed-length sprints with defined roles, events and artefacts to deliver increments.
Kanban Methodology	A continuous flow board that limits work in progress and optimises throughput.
Scrum Roles and Responsibilities	Product Owner, Scrum Master and Developers, each accountable for a distinct part of the work.
Product Owner Responsibilities	Maximising product value by managing and ordering the backlog and accepting work.
Sprint Planning	The team selects backlog items and plans how to deliver them within the sprint goal.
Daily Stand-ups	A 15-minute sync on progress toward the sprint goal and blockers.
Sprint Reviews	Demonstrating the increment to stakeholders and gathering feedback.
Sprint Retrospectives	The team reflects on how it worked and agrees small improvements.
Backlog Grooming	Older term for refinement: keeping the backlog clear, sized and ordered.
Story Points	A relative unit of effort and complexity for sizing stories, not hours.
Estimation Techniques	Planning poker, t-shirt sizing, affinity estimation and reference-class estimates to forecast work.
Velocity and Burndown Charts	Velocity is points completed per sprint; burndown shows remaining work over time. Use for forecasting, not for judging people.

Release Planning	Sequencing sprints into releases with dates, scope, dependencies and risk buffers; staged rollout and rollback plans belong here.
Dependency Management	Identifying and tracking work that relies on other teams, systems or vendors, and resolving early.
Risk and Issue Management	Risks are possible problems to mitigate; issues are problems that have occurred and need action.
Cross-Functional Team Collaboration	Product, design, engineering, data, QA and go-to-market working toward one goal with shared context.
Engineering Collaboration	Working with engineers on trade-offs, feasibility and scope, treating them as partners from discovery onward.
Product Delivery Lifecycle	Idea to discovery, design, build, test, release, measure and iterate.
Jira	Issue and sprint tracking tool for backlogs, boards, workflows and reports.
Confluence	Team wiki used for specs, decision records and knowledge sharing, integrated with Jira.
Linear and Azure DevOps	Linear is a fast, opinionated issue tracker; Azure DevOps combines boards, repos and pipelines for engineering teams.

Practice exercise: Break one epic into stories small enough for a sprint, estimate them with planning poker and draft the sprint goal.

Self-check

1. Explain Agile Fundamentals in two sentences to a non-technical stakeholder, then say what you would measure for it.
2. Explain Story Points in two sentences to a non-technical stakeholder, then say what you would measure for it.
3. Explain Linear and Azure DevOps in two sentences to a non-technical stakeholder, then say what you would measure for it.

Related SkillNavora material

- 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