AI PROMPTING PLAYBOOK

For Product Managers

15 Ready-to-Use Prompts for Your Daily Work

Most product managers ask ChatGPT vague questions and get generic answers.

This playbook gives you specific, battle-tested prompts that actually work.

No theory. Just copy, customize, and use.

Created by ACM Pro | acmpro.in

How to Use This Playbook

Each prompt follows the same structure:

Use Case: When to use this prompt
The Prompt: Copy-paste template (customize the [BRACKETED] parts)
Pro Tip: How to get better results

Golden Rule: Specificity beats vagueness. The more context you give, the better the output.

1. User Story Refinement

Use Case: You have a rough user story that needs detail and acceptance criteria

```
I'm a Product Manager for [PRODUCT TYPE, e.g., "a B2B SaaS project management tool"]. I have this user story: "[YOUR STORY]" Help me refine it by: 1. Rewriting it in proper "As a [user], I want [goal], so that [benefit]" format 2. Adding 3-5 specific acceptance criteria 3. Identifying potential edge cases I should consider 4. Suggesting any missing context or requirements Context about our users: [ADD 1-2 SENTENCES ABOUT YOUR USERS]
```

Pro Tip: Always include your product type and user context. 'B2B SaaS for enterprise' gets different suggestions than 'mobile app for consumers.'

2. Feature Prioritization Analysis

Use Case: You have multiple feature requests and need help thinking through prioritization

```
I need to prioritize these features for Q[X]: [LIST YOUR FEATURES, e.g.: 1. Advanced filtering on dashboard 2. Mobile app dark mode 3. Bulk export functionality 4. Integration with Salesforce] My constraints: - Team capacity: [X developers for Y weeks] - Business goal: [e.g., "increase user retention by 15%"] - User feedback: [e.g., "top complaint is slow load times"] For each feature, analyze: 1. Estimated impact on our business goal (High/Medium/Low) 2. Estimated effort (Small/Medium/Large) 3. Dependencies or risks 4. Recommendation on priority order with reasoning
```

Pro Tip: Include your actual constraints (team size, timeline, business goals). Generic prioritization frameworks are useless without context.

3. Stakeholder Status Update

Use Case: You need to write a status update for executives or non-technical stakeholders

```
I need to write a status update for [STAKEHOLDER, e.g., "our CEO" or "the executive team"]. Project: [PROJECT NAME] Audience: [WHO THEY ARE and WHAT THEY CARE ABOUT] Current status: [BRIEF SUMMARY] Key achievements this period: [LIST 2-3] Current blockers: [LIST ANY] Upcoming milestones: [LIST 2-3] Write a concise status update (200-300 words) that: 1. Leads with the most important information 2. Frames technical issues in business terms 3. Is clear about risks without being alarmist 4. Ends with a specific ask or next step if needed
```

Pro Tip: Specify what your stakeholder cares about. CFOs want cost/ROI, CEOs want strategic impact, tech leads want architecture decisions.

4. Quick Competitive Feature Analysis

Use Case: You need to understand how competitors handle a specific feature

```
I'm researching how competitors handle [SPECIFIC FEATURE, e.g., "onboarding flow" or "pricing tiers"]. Our product: [YOUR PRODUCT] Competitors to analyze: [LIST 3-5 COMPETITORS] What I want to learn: [e.g., "best practices for free trial conversion" or "how they structure their pricing"] Create a comparison that shows: 1. How each competitor implements this feature 2. Strengths and weaknesses of each approach 3. What we can learn or avoid 4. Recommendation for our implementation Note: Base this on your general knowledge of these products. I'll verify details.
```

Pro Tip: AI can give you a starting framework, but always verify against actual competitor products. Use this for brainstorming, not as final research.

5. Requirements Clarification Questions

Use Case: A stakeholder gave you vague requirements and you need to ask better questions

```
A stakeholder requested: "[THEIR VAGUE REQUEST]" Our product: [PRODUCT TYPE]
Stakeholder: [THEIR ROLE/DEPARTMENT] Generate 10-15 clarifying questions I should
ask to properly scope this request, covering: 1. User impact and use cases 2.
Success criteria 3. Technical constraints 4. Timeline and dependencies 5. Edge cases
and error handling Format as questions I can directly ask in our next meeting.
```

Pro Tip: Use these questions as a starting point in your stakeholder meeting. Don't just send them via email - have a conversation.

6. Sprint Planning Preparation

Use Case: You need to prepare stories and talking points for upcoming sprint planning

```
I'm preparing for sprint planning. Here's our backlog for next sprint: [PASTE YOUR
TOP 10-15 BACKLOG ITEMS] Team capacity: [X story points or Y developer-days] Sprint
goal: [WHAT WE WANT TO ACHIEVE] Help me: 1. Identify stories that should be broken
down further 2. Spot dependencies between stories 3. Suggest a logical order to
tackle these 4. Draft 2-3 talking points to explain the sprint goal to the team 5.
Identify any risks or unknowns we should discuss
```

Pro Tip: Review the AI's suggestions with your tech lead before sprint planning. They'll catch technical dependencies AI might miss.

7. User Interview Insights Extraction

Use Case: You've done user interviews and need to find patterns in the feedback

```
I conducted [X] user interviews about [TOPIC]. Here are my raw notes: [PASTE YOUR INTERVIEW NOTES] Analyze these notes and identify: 1. Top 3-5 recurring pain points 2. Common feature requests or needs 3. Surprising insights or patterns 4. Quotes that strongly represent user sentiment 5. Recommended next steps based on this feedback Format as a summary I can share with my team.
```

Pro Tip: Don't rely solely on AI analysis. Read through your notes yourself first, then use AI to validate your thinking or spot patterns you missed.

8. Explain Technical Concept Simply

Use Case: Engineers explained something technical and you need to communicate it to non-technical stakeholders

```
The engineering team explained this technical concept to me: "[TECHNICAL EXPLANATION FROM ENGINEERS]" I need to explain this to [AUDIENCE, e.g., "our sales team" or "executives"]. Rewrite this explanation: 1. Using simple, non-technical language 2. With an analogy that makes it relatable 3. Focusing on business impact, not technical details 4. In 3-4 sentences maximum Also suggest one diagram or visual that would help explain this.
```

Pro Tip: Test your simplified explanation on the engineers first. Make sure you didn't lose important nuance while simplifying.

9. Roadmap Communication

Use Case: You need to communicate your roadmap to different audiences

```
Here's our product roadmap for the next quarter: [LIST YOUR PLANNED
FEATURES/INITIATIVES] Create three versions of this roadmap summary: 1. For
customers: Focus on benefits and value 2. For sales team: Focus on competitive
advantage and talking points 3. For internal team: Focus on strategy and business
rationale Each version should be 150-200 words.
```

Pro Tip: Different audiences care about different things. Customers want outcomes, sales wants ammunition, internal teams want the 'why.'

10. Structured Meeting Agenda

Use Case: You need to run an effective meeting and want a clear agenda

```
I'm running a meeting about: [TOPIC] Attendees: [LIST ROLES/NAMES] Goal: [WHAT WE
NEED TO DECIDE OR ACCOMPLISH] Time available: [DURATION] Key discussion points:
[LIST 3-5 TOPICS] Create a structured meeting agenda with: 1. Time allocation for
each topic 2. Pre-work or prep attendees should do 3. Decision points that need
resolution 4. Suggested discussion format for contentious topics 5. Clear next
steps/action items template
```

Pro Tip: Send the agenda 24 hours before the meeting. People who show up prepared make meetings 10x more productive.

11. A/B Test Results Interpretation

Use Case: You ran an A/B test and need help interpreting results

```
We ran an A/B test: Feature tested: [WHAT YOU TESTED] Variant A (control): [DESCRIPTION] Variant B (treatment): [DESCRIPTION] Results: - Sample size: [X users per variant] - Metric tracked: [e.g., "conversion rate"] - Variant A result: [NUMBER] - Variant B result: [NUMBER] - Statistical significance: [IF YOU KNOW IT] Help me: 1. Determine if this result is meaningful 2. Identify what might explain the difference 3. Recommend whether to ship variant B 4. Suggest follow-up tests or investigations
```

Pro Tip: AI can help interpret, but always discuss results with your data analyst. They'll catch statistical issues AI might miss.

12. Feature Risk Assessment

Use Case: You're about to ship a feature and want to think through potential risks

```
We're about to ship this feature: [DESCRIBE THE FEATURE] User base: [WHO USES THIS] Impact: [HOW MANY USERS AFFECTED] Technical changes: [HIGH-LEVEL OVERVIEW] Identify potential risks in these categories: 1. User experience risks (confusion, frustration) 2. Technical risks (performance, bugs, edge cases) 3. Business risks (churn, support load, revenue impact) 4. Reputation risks (negative feedback, PR issues) For each risk, suggest a mitigation strategy.
```

Pro Tip: Do this exercise with your team, not alone. Different people spot different risks based on their experience.

13. Customer Feedback Analysis

Use Case: You have a pile of customer feedback and need to categorize and prioritize

```
Here's customer feedback from the last [TIMEFRAME]: [PASTE 10-20 PIECES OF FEEDBACK] Analyze this feedback and: 1. Group into themes/categories 2. Identify top 3 most urgent issues 3. Separate feature requests from bug reports from confusion 4. Highlight any feedback that contradicts other feedback 5. Recommend which items should go into backlog vs. need immediate attention
```

Pro Tip: Look for feedback patterns across customer segments. Enterprise customers often have different needs than SMBs.

14. User-Facing Release Notes

Use Case: You're shipping features and need clear, customer-friendly release notes

```
We're releasing these changes: [LIST TECHNICAL CHANGES OR JIRA TICKETS] Target audience: [e.g., "B2B customers, mostly non-technical"] Tone: [e.g., "professional but friendly"] Write release notes that: 1. Explain each change in user benefit language (not tech specs) 2. Use short, scannable formatting 3. Call out any changes that require user action 4. Are 300 words or less 5. End with a link to detailed documentation if needed
```

Pro Tip: Test your release notes on a customer success person. If they can't explain it to a customer, rewrite it.

15. Sprint Retrospective Analysis

Use Case: You want to analyze patterns across multiple sprint retrospectives

```
Here are notes from our last [X] sprint retrospectives: [PASTE YOUR RETRO NOTES]
Analyze these and identify: 1. Recurring problems or bottlenecks 2. Team strengths
we should leverage more 3. Experiments we tried - which worked, which didn't 4.
Systemic issues vs. one-time problems 5. 3 concrete action items to improve our
process Frame these as talking points for our next retro.
```

Pro Tip: Share this analysis with your team before the retro. It helps everyone come prepared with solutions, not just complaints.

Making These Prompts Work for You

1. Customize, Don't Copy-Paste Blindly
Replace all [BRACKETED] sections with your actual context. The more specific you are, the better the output.

2. Iterate on the Output
AI rarely gets it perfect on the first try. Ask follow-up questions like "make this more concise" or "focus more on the business impact".

3. Verify Everything
AI can hallucinate facts, especially about competitors or technical details. Always verify before using in decisions.

4. Combine with Human Judgment
These prompts help you think faster, not replace your thinking. Your experience and context matter more than AI output.

5. Track What Works
Keep a note of which prompts save you the most time. Refine them based on what works for YOUR specific situation.

6. Share What You Learn
If you modify a prompt and it works brilliantly, share it with your team. Collective learning beats individual optimization.

Remember: AI is a tool, not a replacement for your expertise as a PM.

Use these prompts to handle routine thinking faster, so you can spend more time on strategic decisions, customer conversations, and the messy human problems that AI can't solve.

Good luck!

Created by ACM Pro
Making AI practical for project & product management
acmpro.in | contact@acmpro.in