50 AI Prompts for Project Managers

Copy-Paste Prompts for Planning, Risk, Reporting and Multi-Agent Workflows

How to Use This Prompt Pack

Most project managers who are disappointed with AI are asking it vague questions. "Write a status report" produces a generic template. "Act as a PMO lead, use these 14 task updates, produce a RAG report under 300 words for a CFO audience" produces something you can send. Every prompt in this pack follows the same five-part anatomy so that you get usable output the first time.

The 5-part prompt anatomy

Element	What it does	Example
1. Role	Sets the expertise and perspective the AI should adopt	Act as a PMP-certified project manager with 10 years in IT delivery.
2. Context	Gives the facts the AI cannot know	We are implementing a CRM for a 200-person sales team; go-live is 30 September.
3. Task	States exactly what you want done	Identify the 10 most likely delivery risks.
4. Constraints	Sets limits on scope, tone, length or method	Use a 1–5 scale; focus on integration and data migration; no generic risks.

5. Output format	Defines the shape of the answer	Return a table: Risk | Probability | Impact | Score | Owner | Response.

Filling in the [VARIABLES]

Text in gold square brackets is a placeholder. Replace it with your project's details before you run the prompt. The more specific you are, the better the output.

	Prompt text
Before (vague)	Create a risk register for my project.
After (specific)	Act as a risk manager. Create a risk register for a CRM implementation for a 200-person sales team, integrating with SAP and Outlook, with a fixed go-live of 30 September and a vendor-led data migration. Return 10 risks in a table with probability, impact, owner role and response.

Five habits that improve every result

- **Paste real data:** Task lists, meeting notes and status updates beat descriptions of them.

- **Ask for a table:** Tables are easier to check, copy into tools and compare between versions.
- **Iterate, don't restart:** Follow up with "make it shorter", "add owners" or "challenge your own assumptions".
- **Ask it to flag gaps:** End with "List any information you need that I did not provide."
- **Review before you send:** AI drafts; the project manager decides. Check every date, number and name.

> **Protect your project data**
>
> Never paste confidential client data, personal information, contract values or credentials into a public AI tool. Use your organisation's approved AI platform, anonymise names ("Vendor A", "Stakeholder 3") and follow your data-handling policy.

What's inside

Section	Focus	Prompts

A	Project Initiation and Planning	A1–A8
B	Scheduling and Dependencies	B1–B7
C	Risk Identification and Management	C1–C8
D	Resource Allocation	D1–D6
E	Progress Tracking and Reporting	E1–E8
F	Stakeholder Communication	F1–F7
G	Multi-Agent Orchestration	G1–G6
Bonus	Prompt chaining workflow and quality-check prompts	Q1–Q5

A. Project Initiation and Planning

A strong plan prevents most of the problems you will otherwise spend months fixing. These prompts help you move from a rough idea to a structured charter, scope and work breakdown in hours rather than days, while surfacing the questions you should ask before anything is baselined.

8 prompts in this section

A1 Draft a Project Charter

Use when: You have a project idea and need a first-draft charter for sponsor review.

Act as a senior project manager. Using the details below, draft a one-page project charter with these sections: purpose, SMART objectives, in-scope and out-of-scope items, key deliverables, high-level milestones, assumptions, constraints, success criteria and approval sign-off. Project details: **[PROJECT NAME, SPONSOR, BUSINESS PROBLEM, BUDGET, TARGET DATE]**. Finish by listing any information you need that I did not provide.

You get: A structured one-page charter plus a list of missing inputs.

Pro tip: Run it twice, once "as the sponsor" and once "as a sceptical finance director", to stress-test the objectives.

A2 Turn Requirements into a WBS ★HERO PROMPT

Use when: You have a requirements document and need a work breakdown structure.

Act as a PMP-certified planner. Break the following requirements into a work breakdown structure up to 3 levels deep, using numbering 1.0, 1.1, 1.1.1. Keep each work package between 40 and 80

hours of effort and flag any requirement that is too unclear to estimate. Requirements: **[PASTE REQUIREMENTS]**. Return a table: WBS ID | Work package | Description | Estimated effort (hours).

You get: A numbered WBS table with effort estimates and flagged ambiguities.

Pro tip: Follow up with "Which work packages are most likely to be underestimated, and why?"

▸ **SAMPLE OUTPUT** *CRM implementation, first rows*

WBS ID	Work package	Description	Effort (h)
2.0	Data Migration	Move customer and opportunity data from legacy system	—
2.1	Data audit	Profile 48k legacy records for duplicates and gaps	60
2.2	Mapping rules	Map legacy fields to CRM objects and picklists	50
2.3	Trial migration	Load 10% sample, validate counts and relationships	70
⚠Flag	Requirement R-14	"Migrate historical data" has no date range; cannot estimate	

A3 Generate Scope Clarification Questions

Use when: Before baselining scope, to uncover hidden expectations.

Review this scope statement and list 15 clarifying questions I should ask before baselining scope.

Group them by stakeholder type: business owner, technical team, legal/compliance and end users.

For each question, explain in one line what risk it reduces. Scope statement: **[PASTE SCOPE]**.

You get: 15 grouped questions, each linked to a specific risk.

Pro tip: Turn the answers into an "Agreed scope decisions" log and attach it to the charter.

A4 Map Stakeholders and Engagement

Use when: At project start, or when new stakeholders appear.

Identify the likely stakeholders for a **[PROJECT TYPE]** project in a **[INDUSTRY]** organisation of **[SIZE]**. For each stakeholder, give their interest (High/Medium/Low), influence (High/Medium/Low), main concerns and a recommended engagement strategy. Return a table, then group the stakeholders into a power-interest grid: manage closely, keep satisfied, keep informed, monitor.

You get: A stakeholder register and a power-interest grouping.

Pro tip: Add the real names afterwards inside your own systems, not in the AI tool.

A5 Build an Assumptions and Constraints Log

Use when: When the plan rests on things nobody has written down.

From the project description below, extract every explicit and implicit assumption and constraint. For each, state its category (time, budget, resources, technology, vendor, regulatory), what happens if it proves false, and how to validate it. Project description: **[PASTE DESCRIPTION]**.

You get: A categorised log with validation actions.

Pro tip: Implicit assumptions are the dangerous ones. Ask the AI to mark which ones were not stated in your text.

A6 Define Success Metrics and KPIs

Use when: When objectives are vague ("improve efficiency").

Convert these project objectives into measurable KPIs: **[OBJECTIVES]**. For each KPI provide: definition, formula, baseline data source, target, measurement frequency and owner role. Include at least one leading indicator and one lagging indicator per objective.

You get: A KPI table that can go straight into the charter or dashboard.

Pro tip: Ask "Which KPI could be gamed, and how would we detect that?"

A7 Agile: Split an Epic into User Stories

Use when: Backlog refinement for agile or hybrid projects.

Act as an experienced product owner. Split this epic into user stories in the format "As a **[user]**, I want **[goal]**, so that **[benefit]**". For each story, write 3 acceptance criteria in Given/When/Then format and suggest a story-point estimate on the Fibonacci scale. Keep each story small enough to finish in one sprint. Epic: **[PASTE EPIC]**.

You get: Sprint-ready user stories with acceptance criteria and estimates.

Pro tip: Have the team re-estimate in planning poker. Use AI estimates only as a starting anchor.

A8 Apply Lessons Learned to a New Plan

Use when: Starting a project similar to one your organisation has done before.

Here are lessons learned from previous similar projects: **[PASTE LESSONS]**. My new project is: **[DESCRIPTION]**. Identify the 5 lessons most relevant to this project and turn each one into a specific planning action, an owner role and the plan section it should change.

You get: Five concrete, owned actions instead of a forgotten lessons file.

Pro tip: Re-run this prompt at each phase gate with that phase's lessons.

B. Scheduling and Dependencies

Schedules break when dependencies are hidden and delays ripple further than expected. These prompts help you map dependencies, estimate with ranges instead of guesses, and recalculate quickly when something slips, which is exactly the job a scheduling agent performs in a multi-agent workflow.

7 prompts in this section

B1 Map Task Dependencies

Use when: You have a task list but no clear sequence.

Analyse this task list and identify dependencies between tasks. Classify each as Finish-to-Start, Start-to-Start or Finish-to-Finish, and explain the reason in a few words. Then identify which chain of tasks is most likely to be the critical path. Tasks: **[TASK | DURATION | OWNER]**.

You get: A dependency table and a likely critical path.

Pro tip: Ask it to flag any dependency that relies on an external party; these are your schedule risks.

B2 Recalculate After a Delay ★HERO PROMPT

Use when: A task has slipped and you need the impact before the stakeholder call.

Task **[TASK NAME]** is delayed by **[X]** working days. Here is the schedule: **[TASK | DURATION | DEPENDENCIES | START | END]**. Recalculate the dates of every affected task, show the new project end date, state whether the critical path has changed, and suggest 3 recovery options (fast-tracking, crashing, scope trade-off) with the pros and cons of each. Do not change tasks that are not affected.

You get: Updated dates, critical path impact and recovery options.

Pro tip: This mirrors the scheduling agent in the blog's CRM example: the AI recommends, the PM approves.

▸ **SAMPLE OUTPUT** *CRM project, API integration delayed 6 days*

> Affected tasks: Integration testing moves from 4 Aug to 12 Aug; UAT moves from 18 Aug to 26 Aug; Go-live readiness review moves from 8 Sep to 16 Sep.
>
> Project end date: 30 Sep → 8 Oct (+6 working days). The critical path is unchanged and still runs through integration.
>
> Option 1, fast-track: Start UAT test-script preparation in parallel with integration testing. Recovers 3 days; moderate rework risk.
>
> Option 2, crash: Add one integration developer for 2 weeks. Recovers 4 days; cost about 1 extra FTE.
>
> Option 3, scope trade-off: Move Outlook calendar sync to release 1.1. Recovers 6 days; needs sponsor approval.

B3 Create a Milestone Plan

Use when: Executives want a one-page view, not a 300-line Gantt chart.

Create a milestone plan for this project with 8 to 12 milestones. For each milestone give the target date, clear completion criteria (what must be true to call it done), the accountable role and the main dependency. Project summary and dates: **[PASTE]**.

You get: A one-page milestone table with objective completion criteria.

Pro tip: Completion criteria stop "90% done" reporting. Insist on them.

B4 Three-Point (PERT) Estimates

Use when: Estimates are single guesses and keep being wrong.

For each task below, provide optimistic (O), most likely (M) and pessimistic (P) durations with a one-line reason for the pessimistic case. Then calculate the PERT estimate (O + 4M + P) / 6 and the standard deviation (P − O) / 6. Tasks: **[TASK LIST WITH CONTEXT]**. Return a table and the total expected duration for the critical path.

You get: A PERT table with expected durations and uncertainty per task.

Pro tip: Large standard deviations show where to add buffers or do a spike first.

B5 Agile: Draft a Sprint Plan

Use when: Planning releases across several sprints.

Our team velocity is **[X]** story points per sprint and sprints are **[LENGTH]** weeks. Here is the prioritised backlog: **[STORY | POINTS | PRIORITY | DEPENDENCIES]**. Allocate stories into sprints without exceeding 85% of velocity, respecting dependencies and priority. Show the sprint goal for each sprint and the expected release date.

You get: A sprint-by-sprint plan with goals and a forecast release date.

Pro tip: Keeping 15% headroom protects the plan from unplanned work and bugs.

B6 Review a Schedule for Weak Spots

Use when: Before baselining the schedule or presenting it.

Review this schedule as a sceptical PMO auditor. Identify unrealistic durations, missing tasks (testing, approvals, training, cut-over), resource conflicts, tasks with no owner and dependencies that look wrong. Rank the issues by impact. Schedule: **[PASTE]**.

You get: A ranked list of schedule weaknesses to fix before baselining.

Pro tip: Missing approval and review time is the most common gap AI will catch.

B7 Check a Schedule Against Leave and Holidays

Use when: Before committing dates near holiday periods.

Check this schedule against the team leave and public holidays listed below. Flag every task where the assigned person is unavailable, estimate the delay, and suggest a reassignment or date change. Schedule: **[PASTE]**. Leave and holidays: **[LIST]**.

You get: A conflict list with suggested fixes.

Pro tip: Always supply the holiday list yourself; do not rely on the AI to know regional holidays.

C. Risk Identification and Management

AI is particularly good at the breadth of risk thinking: generating categories and scenarios you might not consider under time pressure. Your job is depth: judging which risks are real for this project and owning the response. These prompts cover identification, assessment, monitoring and the new risks introduced by AI agents themselves.

8 prompts in this section

C1 Build a Risk Register from a Brief ★HERO PROMPT

Use when: Kick-off, or when no proper risk register exists.

> Act as a project risk manager. Based on this project brief, identify 10 risks across technical, schedule, resource, vendor, security and stakeholder categories. Write each risk as "If **[cause]**, then **[event]**, resulting in **[impact]**". For each, give probability (1–5), impact (1–5), score (P × I), owner role, response strategy (avoid, mitigate, transfer or accept), a specific response action and an early-warning trigger. Brief: **[PASTE BRIEF]**. Return a table sorted by score.

You get: A sorted, ready-to-use risk register.

Pro tip: Early-warning triggers are what turn a register into a monitoring tool. Never skip them.

▸ **SAMPLE OUTPUT** *CRM project, top 3 risks*

Risk statement	P	I	Score	Response
If legacy data has duplicates, then migration validation fails, resulting in a delayed go-live	4	5	20	Mitigate: run data audit in week 2; trigger = >5% duplicates in sample
If SAP API access is approved late, then integration build stalls, resulting in a 2–3 week slip	3	5	15	Mitigate: raise access request at kick-off; trigger = no approval by week 3
If sales reps see no benefit, then adoption stays low, resulting in missed ROI	3	4	12	Mitigate: involve 5 champion reps in UAT; trigger = <60% log-ins in week 1

C2 Run an AI Pre-Mortem

Use when: Before approving the plan, to surface uncomfortable risks.

> Imagine it is **[DATE 6 MONTHS AFTER GO-LIVE]** and this project has clearly failed. Write the 8 most plausible reasons it failed, as specific stories rather than generic categories. Then, for each reason, suggest one action we could take now to prevent it. Project summary: **[PASTE]**.

You get: Eight concrete failure stories with preventive actions.

Pro tip: Share the output in a team workshop; people speak more freely about a "fictional" failure.

C3 Assess the Impact of a New Issue

Use when: Something just happened and you need to know how serious it is.

A new issue has occurred: **[DESCRIBE ISSUE]**. Given the project context below, assess its impact on scope, schedule, cost, quality and stakeholders (none, low, medium or high, with a reason). Recommend whether to escalate, and draft a 3-sentence escalation note if needed. Context: **[PROJECT STATUS, KEY DATES, BUDGET]**.

You get: A structured impact assessment and a ready escalation note.

Pro tip: Use the same format every time so leadership learns to read it quickly.

C4 Convert Assumptions into Risks

Use when: After building the assumptions log (A5).

Here is our assumptions log: **[PASTE]**. For each assumption, write the risk that exists if it is false, rate probability and impact (1–5), and propose a validation action with a deadline. Highlight the 3 assumptions we should validate first.

You get: Assumptions turned into risks with validation deadlines.

Pro tip: Revisit after each phase; validated assumptions can be closed.

C5 Assess Vendor and Third-Party Risk

Use when: Your project depends on vendors, contractors or SaaS providers.

Assess delivery risk for the following vendor dependencies: **[VENDOR | DELIVERABLE | DUE DATE | CONTRACT TERMS]**. For each, identify risks to delivery, quality, data security and lock-in, the contract levers available, and a contingency plan.

You get: A vendor risk table with contingencies and contractual levers.

Pro tip: Ask "What would we do if this vendor disappeared tomorrow?" for your top vendor.

C6 Identify Integration Risks

Use when: Technology projects that connect several systems.

Act as a solution architect. For a project integrating **[SYSTEM A]** with **[SYSTEMS B, C]**, list the likely integration risks: data mapping, authentication, rate limits, error handling, performance, environments and testing. For each, suggest a test or design check that would detect the problem early.

You get: Integration risks paired with early detection checks.

Pro tip: Share with your technical lead; it makes an excellent design-review checklist.

C7 Update the Risk Register Weekly

Use when: Weekly risk reviews take too long.

Here is last week's risk register: **[PASTE REGISTER]**. Here are this week's project updates: **[PASTE UPDATES]**. Identify which risks have increased, decreased, materialised into issues or can be closed, and any new risks the updates suggest. Explain each change in one line.

You get: A change summary for your weekly risk review.

Pro tip: This is how a risk monitoring agent works in a multi-agent setup: compare, flag, explain.

C8 Identify Risks of Using AI Agents

Use when: Before introducing AI agents into project delivery.

We plan to use AI agents for **[LIST AGENT ROLES, e.g., scheduling, reporting, risk monitoring]**. Identify the risks this introduces, including communication failures between agents, conflicting recommendations, outdated data, data exposure, uncontrolled or repeated actions and unclear accountability. For each risk, propose a control (approval checkpoint, access limit, execution log or human review).

You get: An AI-specific risk register with controls.

Pro tip: Use it as the governance section of your AI pilot proposal.

D. Resource Allocation

Resource problems are often visible in the data long before they become burnout or missed deadlines. These prompts analyse workloads, skills and competing demands, and propose options. Final allocation decisions stay with you, because they involve people, priorities and context that a model cannot see.

6 prompts in this section

D1 Detect Over-Allocation ★HERO PROMPT

Use when: Some team members seem overloaded, but the picture is unclear.

> Here is our resource data: **[NAME/ROLE | SKILLS | WEEKLY CAPACITY (HOURS) | ASSIGNED TASKS AND HOURS PER WEEK]**. Calculate each person's utilisation for the next **[N]** weeks. Flag anyone above 100% or below 60%. Suggest rebalancing moves that match tasks to people with the right skills, and show the utilisation after the changes.

You get: A utilisation table, flagged risks and a rebalanced plan.

Pro tip: Use role labels (Dev 1, BA 2) instead of names in external AI tools.

▸ **SAMPLE OUTPUT** *CRM project, weeks 6–8*

Role	Capacity	Assigned	Utilisation	Suggested move
Integration Dev 1	40 h	56 h	140% ⚠	Move error-handling task (12 h) to Dev 2
Integration Dev 2	40 h	26 h	65%	Takes error-handling task → 95%

Business Analyst	40 h	44 h	110% ⚠	Shift UAT script review (6 h) to QA Lead
QA Lead	40 h	30 h	75%	Takes UAT script review → 90%

D2 Run a Skills Gap Analysis

Use when: Planning a project that needs skills the team may lack.

> Here are the skills this project requires: **[LIST WITH LEVEL NEEDED]**. Here are our team members' skills: **[ROLE | SKILLS | LEVEL]**. Identify the gaps, rate their severity, and recommend for each gap whether to train, hire, contract or reassign, with an estimated lead time.

You get: A gap table with sourcing recommendations.

Pro tip: Training gaps need lead time. Run this before the schedule is fixed.

D3 Create a RACI Matrix

Use when: Roles and responsibilities are unclear.

Create a RACI matrix for the following deliverables and roles. Deliverables: **[LIST]**. Roles: **[LIST]**. Make sure each deliverable has exactly one Accountable role, and flag any deliverable with no Responsible role or too many Consulted roles. Return a table.

You get: A validated RACI matrix with warnings.

Pro tip: If you use AI agents, add each agent as a column. Agents can be Responsible but never Accountable.

D4 Build a Phase-by-Phase Resource Plan

Use when: Estimating staffing for a new project or proposal.

For a **[PROJECT TYPE]** with these phases and durations: **[PHASES]**, estimate the roles and FTE needed in each phase, the ramp-up and ramp-down points, and the peak staffing period. State your assumptions clearly. Return a table of phases × roles with FTE values.

You get: A staffing profile across the project lifecycle.

Pro tip: Compare AI estimates with your organisation's historical data before using them in a budget.

D5 Prioritise Shared Resources Across Projects

Use when: Two or more projects compete for the same people.

Projects **[A, B, C]** all need **[SHARED ROLE/PERSON]** in **[TIME PERIOD]**. Here is each project's priority, deadline, business value and the hours requested: **[PASTE]**. Recommend an allocation using these criteria in order: **[YOUR CRITERIA]**. Show the trade-offs and the impact on each project, and draft a short note for the portfolio steering group.

You get: A transparent allocation recommendation with trade-offs.

Pro tip: Agreeing the criteria first makes the final decision far less political.

D6 Plan Onboarding for a New Team Member

Use when: Someone joins mid-project and needs to contribute quickly.

Create a 2-week onboarding plan for a **[ROLE]** joining our project in the **[PHASE]** phase. Include the documents to read, people to meet, systems to access, a first small task and checkpoints at day 3 and day 10. Project context: **[SUMMARY]**.

You get: A day-by-day onboarding plan.

Pro tip: Request system access before day 1; it is the most common onboarding delay.

E. Progress Tracking and Reporting

Reporting is one of the most time-consuming parts of project management and one of the easiest to delegate to AI. Feed these prompts your raw updates and get structured, audience-ready reports. Spend the time you save on what the report is for: decisions and conversations.

8 prompts in this section

E1 Write a RAG Status Report ★HERO PROMPT

Use when: Weekly status reporting from scattered updates.

> Act as a PMO lead. Using the task updates below, write a weekly status report under 300 words. Include: overall RAG status with a one-line rationale; RAG for schedule, budget, scope and quality; key accomplishments; plan for next week; top 3 risks or issues; and decisions needed from the steering group. Updates: **[PASTE TASK UPDATES, BUDGET FIGURES, MILESTONE DATES]**. Use plain business language for a **[AUDIENCE]**.

You get: A complete, concise status report ready for review.

Pro tip: Define your RAG rules first (e.g., Amber = up to 5 days late) and include them in the prompt.

▸ **SAMPLE OUTPUT** *CRM project, week 9*

Overall: AMBER. Integration is 6 days behind; go-live is still achievable with the recovery option approved last week.

Schedule: Amber | Budget: Green (48% spent, 52% complete) | Scope: Green | Quality: Green

Accomplished: Trial migration of 4,800 records passed with 99.2% accuracy; 3 of 5 champion reps onboarded.

Next week: Complete SAP order sync; start integration testing; finalise UAT scripts.

Decision needed: Approve moving Outlook calendar sync to release 1.1 (recovers 6 days).

E2 Create a 5-Bullet Executive Summary

Use when: Senior leaders need the story in 30 seconds.

Summarise this project report for a senior executive in exactly 5 bullets: status, the biggest achievement, the biggest risk, the financial position and the one decision or support needed. No jargon, no more than 20 words per bullet. Report: **[PASTE]**.

You get: A 30-second executive summary.

Pro tip: Lead with the decision you need when the meeting is short.

E3 Interpret Earned Value Metrics

Use when: You have EVM numbers and need to explain them.

Given BAC = **[X]**, PV = **[X]**, EV = **[X]** and AC = **[X]**, calculate SV, CV, SPI, CPI and EAC (using BAC / CPI). Show each formula and result, explain in plain English what each figure means for this project, and state whether the project is likely to finish on time and on budget if current performance continues.

You get: Calculated EVM metrics with a plain-language interpretation.

Pro tip: Always double-check AI arithmetic with a spreadsheet before reporting numbers.

E4 Explain Variances Against the Plan

Use when: Actuals differ from the baseline and you need to explain why.

Compare planned and actual milestone dates and costs below: **[MILESTONE | PLANNED DATE | ACTUAL/FORECAST DATE | PLANNED COST | ACTUAL COST]**. For each variance, suggest likely causes based on these notes: **[PROJECT NOTES]**, classify it as one-off or systemic, and recommend a corrective action.

You get: A variance analysis with causes and corrective actions.

Pro tip: Systemic variances (like repeated estimation errors) matter more than one-off ones.

E5 Agile: Summarise a Sprint

Use when: Sprint review and stakeholder updates.

Summarise this sprint for stakeholders: committed vs completed story points, sprint goal met (yes/no and why), completed features in business language, carry-over items and reasons, and blockers needing help. Sprint data: **[PASTE BOARD EXPORT OR NOTES]**.

You get: A stakeholder-friendly sprint summary.

Pro tip: Add "compare with the last 3 sprints" to spot velocity trends.

E6 Write a Dashboard Narrative

Use when: Dashboards show numbers but nobody explains them.

Here are this month's dashboard metrics: **[METRIC | TARGET | ACTUAL | LAST MONTH]**. Write a short narrative (under 150 words) that highlights the 3 most important movements, explains what they mean for delivery and states what action, if any, is needed.

You get: A short narrative to sit beside your dashboard.

Pro tip: Ask it to avoid restating every number; insight is the point.

E7 Draft a Project Closure Report

Use when: The project is ending and the closure report is due.

Draft a project closure report with these sections: objectives vs outcomes, final scope, schedule and budget performance, benefits realised and pending, open items and their owners, handover to operations, and key lessons learned. Source material: **[PASTE CHARTER, FINAL STATUS, METRICS]**.

You get: A structured closure report draft.

Pro tip: Benefits often appear after closure. Add a dated benefits-review plan.

E8 Turn Retro Notes into Lessons Learned

Use when: After retrospectives or phase reviews.

> Here are raw notes from our retrospective: **[PASTE NOTES]**. Group them into themes, separate what worked from what to improve, and turn each improvement into a lesson with a recommended action for future projects. Remove names and keep the tone blame-free.

You get: A clean, themed lessons-learned record.

Pro tip: Feed these into prompt A8 at the start of your next project.

F. Stakeholder Communication

Project managers spend most of their time communicating. These prompts help you capture decisions accurately, adapt messages to different audiences and handle difficult conversations, such as delivering bad news, with clarity and confidence.

7 prompts in this section

F1 Meeting Notes to Decisions, Actions and Follow-Up ★HERO PROMPT

Use when: Right after a meeting, while it is fresh.

> From these meeting notes or transcript, extract: (1) decisions made, (2) action items with owner and due date, (3) open questions and (4) risks raised. Then draft a short follow-up email to attendees summarising the decisions and actions. Mark any action without a clear owner or date as "TO CONFIRM". Notes: **[PASTE NOTES OR TRANSCRIPT]**.

You get: Structured meeting outputs and a ready-to-send follow-up email.

Pro tip: The "TO CONFIRM" rule stops unowned actions from disappearing.

▸ **SAMPLE OUTPUT** *CRM steering meeting*

Type	Item	Owner	Due
Decision	Outlook calendar sync moves to release 1.1	Sponsor	—
Action	Update release plan and notify sales leadership	PM	14 Aug
Action	Confirm extra integration developer budget	Finance BP	TO CONFIRM
Open question	Do regional managers need separate dashboards?	BA	Next meeting

F2 Deliver Bad News Clearly

Use when: You must report a delay, overspend or failed milestone.

Draft an email to **[STAKEHOLDER ROLE]** explaining that **[BAD NEWS]**. Structure it as: what happened, impact, root cause in one sentence, the recovery plan with dates and what we need from them. Tone: transparent, calm and solution-focused, with no excuses or blame. Keep it under 200 words.

You get: A professional bad-news email that leads with solutions.

Pro tip: Send bad news early. A short, honest email builds more trust than a polished late one.

F3 Tailor One Update for Three Audiences

Use when: The same news needs to reach executives, the team and the client.

Rewrite this project update for three audiences: (1) an executive sponsor (business impact, 80 words), (2) the delivery team (actions and priorities, 120 words) and (3) the client (progress and next steps, positive but accurate, 100 words). Update: **[PASTE]**.

You get: Three versions of the same update, each fit for its audience.

Pro tip: Check the client version carefully for anything internal that should not be shared.

F4 Summarise a Change Request

Use when: Preparing a change request for the change control board.

Summarise this change request for the change control board: description, reason, impact on scope, schedule, cost, quality and risk, the options (approve, reject, defer) with consequences, and a recommendation. Change details: **[PASTE]**. Keep it to one page.

You get: A one-page decision-ready change summary.

Pro tip: Always include "defer"; it is often the most practical option.

F5 Prepare for a Resistant Stakeholder

Use when: A key stakeholder is pushing back.

A **[STAKEHOLDER ROLE]** is resisting **[CHANGE OR DECISION]** because **[THEIR STATED CONCERNS]**. Suggest the likely underlying concerns, 5 talking points that address them, 3 questions to understand their position better and a possible compromise. Keep the tone respectful and collaborative.

You get: A preparation sheet for a difficult conversation.

Pro tip: Use the questions first. Listening usually resolves more than talking points.

F6 Create a Focused Meeting Agenda

Use when: Meetings run long and end without decisions.

Create a **[DURATION]**-minute agenda for a **[MEETING TYPE]** with **[ATTENDEES]**. The meeting must achieve: **[OUTCOMES]**. For each item give the time, owner, purpose (inform, discuss or decide) and any pre-reading. Put decision items first.

You get: A timed, outcome-driven agenda.

Pro tip: Labelling items "decide" makes it clear when a meeting has succeeded.

F7 Agile: Design a Retrospective

Use when: Retrospectives feel repetitive and produce few changes.

Design a **[DURATION]**-minute retrospective for a team that **[CONTEXT, e.g., just missed a sprint goal / is in its 10th sprint]**. Choose a suitable format, give the facilitation steps with timings, 5 prompting questions, and a method to agree no more than 2 improvement actions.

You get: A ready-to-run retro plan.

Pro tip: Rotate formats every few sprints to keep the conversation honest.

G. Multi-Agent Orchestration

In a multi-agent setup, specialised AI agents handle planning, scheduling, risk, reporting and quality, while an orchestration layer coordinates them and the project manager approves key decisions. You do not need to code to design these systems well. These prompts help you define agent roles, handoffs and controls that your technical team or platform can then implement.

6 prompts in this section

G1 Write a System Prompt for a Project Agent ★HERO PROMPT

Use when: Defining the role of one agent in a multi-agent workflow.

Write a system prompt for an AI **[AGENT ROLE, e.g., planning agent]** in a multi-agent project delivery workflow. Include: its role and goal; the inputs it receives and from which agent; the outputs it must produce and their exact format (JSON); the data and tools it may access; actions it must never take; when it must escalate to the project manager; and how it hands off to the **[NEXT AGENT]**. Project context: **[SUMMARY]**.

You get: A complete, implementable agent definition.

Pro tip: "Actions it must never take" is the most important section. Be explicit.

▸ **SAMPLE OUTPUT** *Planning agent, extract*

Role: You are the Planning Agent for the CRM implementation project. You convert approved requirements into a task breakdown.

Inputs: Approved requirements from the PM; templates from the PMO library.

Outputs: JSON list of tasks {id, name, description, effort_hours, dependencies, open_questions}, sent to the Scheduling Agent.

Never: Change approved scope, assign named people, contact stakeholders or modify the project plan in the PM tool.

Escalate to the PM when: A requirement cannot be estimated, conflicts with another requirement, or implies new scope.

G2 Design an Orchestration Workflow

Use when: Planning how several agents should work together.

Design a multi-agent workflow for **[PROCESS, e.g., weekly status reporting]**. Available agents: **[LIST]**. Define the sequence of steps, the trigger for each step, the data passed between agents, where the human approval checkpoints are, what happens if an agent fails or times out, and what is logged. Recommend a centralised, decentralised or hierarchical orchestration model and explain why.

You get: An end-to-end workflow design with checkpoints and failure handling.

Pro tip: For enterprise projects, centralised or hierarchical models are usually easier to govern.

G3 Define the Handoff Message Format

Use when: Agents are passing incomplete or inconsistent information.

Define a standard JSON message format for handoffs between project agents. It must include sender, receiver, timestamp, task ID, payload, data sources used, confidence level, assumptions made and whether human review is required. Provide a schema and one filled example for a handoff from **[AGENT A]** to **[AGENT B]**.

You get: A handoff schema and an example message.

Pro tip: Requiring "assumptions made" in every handoff makes errors far easier to trace.

G4 Set Conflict Resolution Rules

Use when: Two agents give conflicting recommendations.

Our **[AGENT A]** recommends **[X]** while our **[AGENT B]** recommends **[Y]**. Propose rules for resolving conflicts like this: a priority order between agents, the information each must provide to justify its recommendation, the conditions under which the conflict goes to the project manager, and how the final decision is recorded.

You get: Clear, documented conflict-resolution rules.

Pro tip: Any conflict involving cost, scope or customer impact should always go to a human.

G5 Build an Agent Approval Matrix

Use when: Deciding what agents can do without asking.

For these agent actions: **[LIST, e.g., update task status, change dates, reassign tasks, send stakeholder emails, raise risks]**, classify each as: autonomous, autonomous with notification, requires PM approval or never allowed. Explain the reason for each classification based on reversibility, impact and audience.

You get: An approval matrix that defines agent autonomy.

Pro tip: Start strict. Loosen controls only after the pilot shows reliable performance.

G6 Plan an Agent Pilot and Its Tests

Use when: Before rolling agents out to live projects.

Create a 4-week pilot plan for using AI agents in **[PROCESS]** on **[PROJECT]**. Include objectives, scope, 10 test scenarios (including failure cases such as missing data, conflicting updates and duplicate

actions), success metrics (accuracy, time saved, error rate, user satisfaction) and go/no-go criteria for wider rollout.

You get: A pilot plan with test scenarios and go/no-go criteria.

Pro tip: Measure the baseline time spent on the process before the pilot, or you cannot prove the value.

Bonus: The Prompt-Chaining Workflow

Individual prompts save time. Chained prompts, where the output of one becomes the input of the next, recreate a simple multi-agent workflow without any technical setup. Here is how the CRM implementation project from the blog runs through four prompts from this pack.

Step	Prompt	Input	Output passed on
1. Plan	A2: Requirements to WBS	Requirements document	Task list with effort estimates
2. Schedule	B1 + B4: Dependencies and PERT	Task list from step 1	Sequenced schedule with critical path

3. Assess risk	C1: Risk register	Brief + schedule from step 2	Scored risks with triggers
4. Report	E1: RAG status report	Schedule, risks and weekly updates	Steering-ready status report

When something changes, re-run only the affected link. For example, when integration slipped by six days, prompt B2 recalculated the schedule, C7 updated the risk register and E1 regenerated the report, all within one conversation.

Chaining rules of thumb

- Use one conversation per chain so the AI keeps the context.
- Start each step with "Using the output above…" to make the handoff explicit.
- Review each output before passing it on; errors compound down the chain.

- Keep a final human checkpoint before anything leaves the project team.

Bonus: 5 Quality-Check Prompts

AI output can look confident and still be wrong. Run these prompts on any important output before you rely on it. They are the human-in-the-loop checkpoint in prompt form.

Q1 Check Facts Against the Source

Use when: Before sending any AI-generated report.

> Compare the output above with the source data I provided. List every date, number, name and claim in the output that does not appear in, or contradicts, the source. Mark each as "supported", "not in source" or "contradicts source".

You get: A fact-check table that exposes invented details.

Pro tip: Run this on every status report until you trust the pattern.

Q2 Check for Missing Dependencies

Use when: After generating a WBS or schedule.

Review the plan above. List tasks that are commonly needed in a **[PROJECT TYPE]** but are missing (for example approvals, testing, training, data migration, cut-over, hypercare), and dependencies that appear to be missing or incorrect.

You get: A gap list for the plan.

Pro tip: Combine with B6 for a full schedule health check.

Q3 Detect Conflicting Recommendations

Use when: When several prompts or agents have produced outputs.

Compare these outputs: **[OUTPUT 1]**, **[OUTPUT 2]**, **[OUTPUT 3]**. Identify any conflicting recommendations, dates or assumptions, explain the conflict and state what information would resolve it.

You get: A list of conflicts and how to resolve them.

Pro tip: This is the manual version of the conflict rules designed in G4.

Q4 Executive Tone Check

Use when: Before sending anything to senior stakeholders.

> Review this message for a **[AUDIENCE]**. Flag jargon, vague statements, missing asks, defensive language and anything that could be misread. Then provide a revised version under **[N]** words.

You get: A tightened, audience-appropriate message.

Pro tip: Ask it to state the main point in one sentence first; if it cannot, the message is unclear.

Q5 Surface Hidden Assumptions

Use when: Before acting on any AI recommendation.

> List every assumption you made in the output above that was not stated in my input. For each, rate how much the recommendation would change if the assumption were wrong (low, medium or high).

You get: A transparent list of the AI's assumptions and their sensitivity.

Pro tip: High-sensitivity assumptions are the ones to validate with your team.

TAKE THE NEXT STEP

Lead AI-Driven Projects with Confidence

GSDC Generative AI in Project Management Certification

Prompts are a starting point. To use AI responsibly across a whole project, from planning to closure, you need a structured understanding of what generative AI can and cannot do, and how to keep human oversight, accountability and governance in place. GSDC's Generative AI in Project Management Certification helps professionals understand how generative AI supports planning, risk identification, stakeholder communication, scheduling and reporting.

How the certification connects to this pack

Prompt section	Certification focus area
A. Initiation and Planning	Using generative AI to support project planning

B. Scheduling and Dependencies	AI-assisted scheduling and data-driven decisions
C. Risk Identification	Identifying and evaluating project risks with AI
D. Resource Allocation	Improving productivity and automating repetitive tasks
E. Tracking and Reporting	AI-supported progress tracking and reporting
F. Stakeholder Communication	Using AI to improve stakeholder communication
G. Multi-Agent Orchestration	Integrating AI into workflows with human oversight and governance

Who it's for

- PMPs and certified project managers who want to stay relevant in AI-driven environments
- Programme managers, PMO leads and delivery managers adopting AI tools
- Scrum masters and agile leads working with AI-assisted planning and reporting

- Professionals preparing to manage projects that use AI agents

What you gain

- **Practical AI skills:** Apply generative AI across every stage of project delivery.
- **Responsible adoption:** Integrate AI while keeping oversight, accountability and governance.
- **Productivity:** Reduce time spent on reporting and repetitive administration.
- **Career edge:** A credential that shows you can lead projects in an AI-enabled workplace.

GSDC
Global Skill Development Council

CERTIFICATION IN GENERATIVE AI IN PROJECT MANAGEMENT

Stand out with the Certified Generative AI in Project Management program. Personlized learning path for you



ABOUT GSDC CERTIFICATION

LIFETIME VALIDITY

GSDC Certification is an globally accreditted certification with lifetime validity.

EBOOK

Extensive and exclusive Ebook created by world's experts to help you with understanding core concepts.

CREATED BY EXPERTS

GSDC certifications are created and authored by world's leading experts in the field.

LEARNING MATERIALS

Get access to learning materials such as videos, ebooks, templates, and practice exams, which will help you clear the certification exam.

LEARNING OBJECTIVE

- **Upskill in GenAI applications tailored for project managers.**
- **Learn directly from industry SMEs with real-world project insights.**
- **Build practical skills for planning, execution, and optimization.**

Enroll now with the code **LEARN20** To avail **20%** discount

Enroll Now

www.gsdcouncil.org