Summary

PMP Exam Master Cheat Sheet

---

1. Process Groups & Knowledge Areas

Process Groups (5)

1. Initiating – Define scope, identify stakeholders, Project Charter.
2. Planning – Create PMP, baselines, define WBS, risk planning.
3. Executing – Direct/manage work, acquire/manage team, implement quality.
4. Monitoring & Controlling – Track performance, control scope/schedule/cost/risk.
5. Closing – Validate deliverables, release resources, lessons learned.

Knowledge Areas (10)

"I Saw Six Cats Quickly Running Carrying Really Pretty Socks"

- I – Integration
- S – Scope
- S – Schedule
- C – Cost
- Q – Quality
- R – Resource
- C – Communications
- R – Risk
- P – Procurement
- S – Stakeholder

Stakeholder & Communication Management

- Identify all stakeholders early and maintain engagement (Stakeholder Engagement Assessment Matrix).
- Tailor communication methods based on audience (executives ≠ developers) : Interactive, Push/Pull communication
- Use active listening and empathy to manage conflicts.

Exam tip: Always engage stakeholders continuously, especially in agile environments.

---

2. PMP Formulas (Must Memorize!)

Earned Value Management (EVM)

- BAC (Budget at Completion)
- EV (Earned Value) = % Complete × BAC
- PV (Planned Value) = Planned % × BAC

- AC (Actual Cost) = Spent so far
- CPI (Cost Performance Index) = EV ÷ AC
- SPI (Schedule Performance Index) = EV ÷ PV
- CV (Cost Variance) = EV − AC
- SV (Schedule Variance) = EV − PV
- EAC (Estimate at Completion)
    - If typical: EAC=BAC ÷ CPI
    - If atypical: EAC=AC + (BAC−EV)
    - If considering both CPI & SPI: EAC=AC +(BAC−EV) ÷ (CPI×SPI)
- ETC (Estimate to Complete) = EAC − AC
- VAC (Variance at Completion) = BAC – EAC
- To-Complete Performance Index (TCPI)
    - TCPI (to meet BAC) = (BAC – EV) ÷ (BAC – AC)
    - TCPI (to meet EAC) = (BAC – EV) ÷ (EAC – AC)

Critical Path & Float

- Float (Slack) = LS − ES = LF – EF

Where:

- ES (Early Start): Earliest time an activity can begin, based on predecessors.
- EF (Early Finish): Earliest time an activity can finish = ES + Duration.
- LS (Late Start): Latest time an activity can begin without delaying the project.
- LF (Late Finish): Latest time an activity can finish without delaying the project.

Meaning of the Formula:

- LS − ES: Difference between the latest you can start and the earliest you can start.
- LF − EF: Difference between the latest you can finish and the earliest you can finish.
- Both give the same result = Float.

Types of Float:

- Total Float: Time an activity can slip without delaying the project end date.
- Free Float: Time an activity can slip without delaying the start of its successor.
- Critical Path → Path with 0 float.

Estimating

- Triangular = (O + M + P)/3
- (Expected Duration (TE)) -> Beta (PERT) = (O + 4M + P)/6
- Standard Deviation (**σ** or SD) -> **σ** (SD) = (P − O)/6
- Channels = n(n−1)/2

Team Development (Tuckman's Model)

- Forming → Storming → Norming → Performing → Adjourning
- Conflict is normal; focus on resolution, not escalation.
- Use coaching, mentoring, and empowerment to improve performance.

◇ 3. Collect Requirements

Technique	When to Use	Example
Interviews	When you need detailed, individual input from stakeholders.	PM interviews marketing lead to define campaign requirements.
Focus Groups	When you want diverse stakeholder input quickly.	Gather customers + users to discuss features of a new mobile app.
Facilitated Workshops (e.g., JAD)	When you need cross-functional alignment.	Business, IT, and compliance meet to define system requirements.
Brainstorming	For idea generation, especially early in project.	Team brainstorms features for a new e-commerce site.
Questionnaires / Surveys	When you need input from large groups.	Online survey sent to 1,000 potential customers.
Prototypes	When requirements are unclear → show model, refine through feedback.	Create wireframes for a new web platform.
Observation ("Job Shadowing")	When you want to see how users actually work.	PM shadows warehouse staff to define inventory system requirements.
Benchmarking	When you want to compare practices with industry peers.	Compare logistics practices with leading competitors.
Document Analysis	When historical/legacy systems exist.	Analyze current process manuals to extract requirements.

✅ Exam Tip:

- If requirements are unclear → Prototypes.
- If users can't articulate needs → Observation.
- If lots of stakeholders → Surveys.
- If cross-functional alignment needed → Facilitated workshops.

---

4. Estimating Techniques

Technique	How It Works	Accuracy	When to Use	Pros	Cons
Analogous (Top-down)	Uses historical data from similar past projects	-25% to +75% (least accurate)	Early stage, little detail available	Quick, low effort	Not precise, assumes history is relevant
Parametric	Uses statistical models (e.g., cost per unit, productivity rates)	-10% to +25% (moderate accuracy)	When reliable data and scalable parameters exist	More accurate than analogous, fast	Limited by quality of data and assumptions
Three-Point (PERT)	Combines Optimistic (O), Most Likely (M), Pessimistic (P) → weighted avg: (O+4M+P)/6(O + 4M + P)/6	-15% to +20% (better than parametric/analogous)	When uncertainty/risk exists, need realistic forecast	Accounts for risk/uncertainty, provides range	Requires expert input, can be subjective
Bottom-Up	Estimates each work package, then rolls up	-5% to +10% (most accurate)	When WBS is detailed & scope well-defined	Highest accuracy, detailed	Time-consuming, resource-intensive

5. Risk & Change Management

- Always manage proactively, not reactively.
- For change requests → Follow the Integrated Change Control process.
- For risks → Identify, analyze, respond, monitor (use risk register).

Exam tip: If scope changes → never skip the change control process.

- EMV (Expected Monetary Value): Probability × Impact
- Decision Tree Analysis → Sum of EMVs

Risk Responses

🔹 Threat Responses (Negative Risks)

👉 "AMTA" → Avoid, Mitigate, Transfer, Accept

Think: *"Avoid My Terrible Accident"* 🚗💥

- Avoid → eliminate risk.
- Mitigate → reduce risk.
- Transfer → give to someone else.
- Accept → live with it.

🔹 Opportunity Responses (Positive Risks)

👉 "EESA" → Exploit, Enhance, Share, Accept

Think: *"Enjoy Every Smart Advantage"* 🌟

- Exploit → guarantee it happens.
- Enhance → increase chance/impact.
- Share → partner to maximize.
- Accept → take it if it comes.

Strategy	Type of Risk	When to Use	How It Works	Example
Avoid *(AMTA)*	Negative (Threat)	Use when the risk cannot be accepted and you want to eliminate it completely.	Change the project plan, scope, or approach to remove the threat.	Changing the supplier to avoid a high-risk vendor.
Mitigate *(AMTA)*	Negative (Threat)	Use when you cannot remove the risk but can reduce its probability or impact.	Add safeguards, quality checks, redundancies, or training.	Adding extra testing to reduce defects.
Transfer *(AMTA)*	Negative (Threat)	Use when you cannot manage the risk directly but can shift the responsibility to a third party.	Use contracts, warranties, or insurance.	Purchasing cyber insurance to handle potential data breaches.
Accept *(AMTA & EESA)*	Negative (Threat) & Positive (Opportunity)	Use when the cost or effort to manage the risk is higher than its impact.	Do nothing proactively but monitor the risk.	Accepting minor shipment delays.
Exploit *(EESA)*	Positive (Opportunity)	Use when you want to ensure the opportunity happens.	Assign extra resources or modify plans to guarantee success.	Adding a dedicated team to secure a key client deal.
Enhance *(EESA)*	Positive (Opportunity)	Use when you want to increase the likelihood or impact of an opportunity.	Improve processes, add resources, or optimize schedules.	Offering early delivery discounts to attract more customers.
Share *(EESA)*	Positive (Opportunity)	Use when the opportunity can be maximized by partnering with others.	Collaborate with another organization or stakeholder.	Partnering with a vendor to co-develop a product.

Project Selection

- Benefit Cost Ratio (BCR): Benefits ÷ Cost

- NPV (Net Present Value): Higher is better
- IRR (Internal Rate of Return): Higher is better
- Payback Period: Shorter is better

Metric	Definition	Formula	Decision Rule	PMP Recommendation	Pros	Cons
NPV *(Net Present Value)*	Measures the value added by the project today, considering the time value of money.	$NPV = \sum_{t=1}^{n} \frac{R_t}{(1+i)^t} - C$	- NPV > 0 → Accept ✅ - NPV < 0 → Reject ❌	Choose the project with the highest positive NPV.	- Considers time value of money - Shows actual monetary value added	- Sensitive to discount rate - Requires cashflow estimates
IRR *(Internal Rate of Return)*	The discount rate at which NPV = 0; represents the project's expected annual return.	Solve: $0 = \sum_{t=1}^{n} \frac{R_t}{(1+IRR)^t} - C$	- IRR > Cost of Capital → Accept ✅ - IRR < Cost of Capital → Reject ❌	Choose the project with the highest IRR if risks are similar.	- Easy to interpret as a % return - Useful for ranking projects	- Complex to calculate - Misleading when projects have non-normal cashflows
ROI *(Return on Investment)*	Measures the profitability percentage relative to investment cost.	$ROI = \frac{\text{Total Benefit} - \text{Cost}}{\text{Cost}} \times 100$	Higher ROI = Better project ✅	Use when comparing profitability quickly; not as robust as NPV/IRR.	- Simple to calculate - Intuitive for stakeholders	- Ignores time value of money - Can be misleading if project duration differs
Payback Period	Time required to recover the initial investment.	$\text{Payback Period} = \frac{\text{Initial Investment}}{\text{Annual Cash Inflow}}$	Shorter Payback Period = Better ✅	Use for risk-sensitive projects or when liquidity is critical.	- Simple and quick - Good for risk analysis	- Ignores time value of money - Ignores benefits after payback

6. Conflict Resolution (PMI Order of Preference)

1. Collaborating/Problem Solving → Win-Win ✅
2. Compromising → Partial win/lose
3. Smoothing → Focus on agreements
4. Forcing → Win-lose
5. Withdrawal/Avoidance → Not addressing

---

7. Leadership & Management Styles

- Servant Leadership → Key for Agile (support teams, remove blockers).
- Transformational → Inspire and empower teams.
- Transactional → Reward/punish based on performance.
- Autocratic → PM makes all decisions
- Laissez-faire → Team-driven
- Situational Leadership → Adapt style based on team maturity.

Exam tip: Prefer servant leadership in adaptive/agile projects.

---

8. Procurement Types

- Fixed Price (FP): Risk on seller
- Cost Reimbursable (CR): Risk on buyer
- Time & Material (T&M): Hybrid

---

9. PMP Project Management Approaches

Approach	When to Use	Pros	Cons
Predictive (Waterfall)	- Clear, stable requirements- Compliance/regulatory projects- Construction, manufacturing	- Detailed planning upfront- Easy to track progress vs. plan- Works well with fixed scope & budget	- Inflexible to change- Risk of late discovery of issues- Customer sees product late
Iterative	- Scope is evolving but you want early feedback on partial solutions- R&D, prototyping	- Early delivery of working versions- Stakeholder feedback guides improvements	- More planning effort- Can feel slow if increments don't add much value
Incremental	- When you can deliver value in usable pieces (modules, features)- Software product rollouts	- Value delivered early- Reduces risk by smaller releases	- Integration challenges- May need strong release management

Approach	When to Use	Pros	Cons
Agile	- Requirements are uncertain or changing- High stakeholder involvement- Innovation-driven environments	- Very flexible- Frequent customer feedback- Prioritizes high-value features	- Needs mature, engaged team- Less effective with fixed contracts/budgets- Harder in regulated environments
Hybrid	- Large, complex projects where some parts are predictable and others are adaptive- Digital transformation, public works with IT component	- Balances stability and flexibility- Tailors to different workstreams	- More complex governance- Requires strong coordination between predictive and agile tracks

Complement: Agile vs Predictive (Key Differences)

Aspect	Predictive (Waterfall)	Agile (Adaptive)
Scope	Fixed	Evolving
Cost/Schedule	Fixed baseline	Flexible, iterative
Change Mgmt	Formal CR process	Welcomed & integrated
Role of PM	Directs, controls	Facilitator, servant leader
Documentation	Heavy	Just enough
Delivery	End of project	Incremental

Complement: Agile "controls" (in PMP / PMI-ACP language)

When PMI refers to Agile controls, they usually mean mechanisms to monitor and adjust work such as:

- Daily standups
- Burndown / burnup charts
- Velocity tracking
- WIP limits (Kanban)
- Definition of Done / Ready
- Reviews and retrospectives

These are about governance and monitoring progress.

Complement: Agile Roles Cheat Sheet

Role	Key Responsibilities	What They Decide	What They Do Not Decide
Product Owner (PO)	- Owns product vision & roadmap- Manages backlog (ordering/prioritizing)-	- What features to build- Priority of work- Accept/reject increments	- How the team delivers- Technical decisions- People management

Role	Key Responsibilities	What They Decide	What They Do Not Decide
	Ensures value delivery to stakeholders		
Scrum Master (SM)	- Servant-leader- Coaches Agile practices- Removes impediments- Facilitates ceremonies	- Team process improvements- How Agile ceremonies are run	- Product scope or priorities- Technical implementation
Agile Team (Developers/Testers/Designers)	- Self-organize to deliver increments- Estimate work- Commit to sprint goals- Ensure quality (DoD)	- How to build the solution- Technical choices- Work estimates	- What features to build- Final say on priorities
Project Manager (in Hybrid/Scaled Agile)	- Aligns Agile work with organizational strategy- Manages stakeholders outside the team- Ensures governance, budget, reporting	- Governance & compliance- Integrating Agile outputs with larger program/portfolio	- Daily team decisions- Detailed backlog management

✅ Exam Tip Rules of Thumb:

- If question is about scope, backlog, product changes → PO.
- If about team process, impediments → SM.
- If about how to deliver, technical choices → Team.
- If about budget, governance, portfolio alignment → PM (in hybrid/large orgs).

Complement: DoR vs DoD Cheat Sheet

Concept	Definition	Purpose	Example Criteria
Definition of Ready (DoR)	A checklist of conditions that a backlog item must meet before the team can pull it into a sprint.	Ensures items are clear, small, and testable before committing. Prevents half-baked work entering the sprint.	- User story well-written (INVEST)- Acceptance criteria defined- Dependencies identified- Story estimated- Business value understood
Definition of Done (DoD)	A shared understanding of what "complete" means for work delivered by the team.	Ensures increments are truly finished, potentially shippable, and meet quality standards.	- Code completed & peer reviewed- Unit & acceptance tests passed- Integrated into main branch- No critical defects open- Documentation updated

✅ Key Exam Tips:

- DoR = Entry criteria (before sprint / work starts).
- DoD = Exit criteria (after sprint / work is complete).
- If work is incomplete at sprint end → DoR was weak.
- If work is "done" but not shippable → DoD was weak.

Complement: PMP Schedule Techniques

Technique	What It Does	Purpose	Pros	Cons
Resource Leveling	Adjusts start/finish dates based on resource availability (no one is overallocated). May extend schedule.	Balances resource usage.	- Prevents burnout- Realistic plan	- Can delay project- May change critical path
Resource Smoothing	Adjusts activities within available float only → project end date does not change.	Optimizes resource usage without delaying project.	- Keeps end date fixed- Reduces peaks in resource demand	- Less flexible- Only possible if float exists
Crashing	Add extra resources (people, overtime, money) to critical path activities to shorten schedule.	To finish project sooner.	- Shortens schedule- Keeps sequence same	- Increases cost- May cause inefficiency
Fast Tracking	Perform activities in parallel that were originally sequential.	To finish earlier without adding cost.	- No direct cost increase- Can save significant time	- Increases risk- May cause rework

✅ Easy Exam Mnemonics

- Leveling = Resources drive schedule (can delay project).
- Smoothing = Adjust within float (no project delay).
- Crashing = Add resources = more cost.
- Fast tracking = Overlap activities = more risk.

---

10. Important PMP Acronyms

- BAC – Budget at Completion
- EVM – Earned Value Management
- WBS – Work Breakdown Structure
- OBS – Organizational Breakdown Structure
- RACI – Responsible, Accountable, Consulted, Informed
- MoSCoW – Must, Should, Could, Won't

---

11. Quality Control

Tool	When to Use	Pros	Cons
Scatter Diagram	To check if there's a correlation between two variables (e.g., defect rate vs. temperature).	- Simple visual correlation- Helps identify cause–effect hypotheses	- Doesn't prove causation- Hard to interpret with many variables
Affinity Diagram	To organize ideas/issues after brainstorming or collecting customer feedback.	- Groups complex ideas clearly- Great for root cause exploration	- Subjective grouping- Doesn't quantify importance

Tool	When to Use	Pros	Cons
Control Chart	To monitor process stability over time and check if a process is “in control.”	- Identifies common vs. special cause variation- Helps avoid over-adjusting processes	- Requires data collection over time- Can be misread if limits aren’t set correctly
Histogram	To see the distribution or frequency of defects or measurements.	- Easy to interpret- Highlights most frequent problems	- No info on sequence or time- Doesn’t show relationships between causes
Pareto Chart (80/20 rule)	To identify the vital few causes that contribute to most problems.	- Prioritizes issues by impact- Focuses efforts on what matters most	- Oversimplifies (not all problems fit 80/20)- Needs accurate data collection
Fishbone (Ishikawa) Diagram	To analyze root causes of a problem across categories (Machines, Methods, Materials, People).	- Structured brainstorming- Reveals multiple root causes	- Qualitative, not quantitative- Needs follow-up data to confirm causes

Complement: Test Quality Control

Practice	When to Use	Pros	Cons
Test-Driven Development (TDD)	When you want high code quality and fast defect detection. Write tests first, then code to pass them.	- Forces clear requirements - Prevents over-coding - Reduces bugs early	- Requires skilled devs - Slows initial development
Behavior-Driven Development (BDD)	When you want business + technical alignment. Uses natural language scenarios (Given–When–Then).	- Improves communication with non-tech stakeholders - Ensures features meet business needs	- Needs strong collaboration - Can be time-consuming
Acceptance Test-Driven Development (ATDD)	When you want to ensure requirements are validated before coding begins. Acceptance criteria drive the development.	- Builds shared understanding between PO, testers, and devs - Ensures features meet agreed expectations - Reduces rework	- Needs clear, upfront acceptance criteria - Can delay start if PO is not engaged
Continuous Integration (CI)	When multiple devs commit code frequently. Builds and tests automatically.	- Detects integration errors quickly - Encourages frequent commits - Supports Agile delivery pace	- Needs automation setup - Fails if tests aren’t comprehensive
Pair Programming	When you want knowledge sharing and fewer errors. Two devs work at one workstation.	- Higher code quality - Cross-training - Real-time review	- More effort/cost upfront - Can slow down if pairs don’t collaborate well

Practice	When to Use	Pros	Cons
Automated Regression Testing	When frequent changes risk breaking existing features.	- Saves time in long run - Catches recurring defects <	

12. Exam Strategy Tips

- PMI Mindset: Always choose the answer that shows → Proactive, Preventive, Collaborative, Long-term, and Formal process-oriented.
- Escalate to sponsor/stakeholders only if outside PM's authority.
- Never choose punitive actions (e.g., blaming, firing).
- Always analyze root cause before corrective action.
- Favor communication, risk management, and team empowerment.

Situation	PMP Preferred Action
Scope change requested	Go through formal change control
Stakeholder conflict	Collaborate and seek win-win
Unclear requirements	Choose agile/iterative
Fixed regulatory constraints	Choose predictive
Deliver value quickly	Choose incremental/agile
Team not performing	Coach, mentor, support

---

13. Exam Timing Tips

PMP Time Management Sheet (230 minutes, 180 questions)

Question #	Target Elapsed Time	Time Remaining	Notes
Q30	~35 minutes	~185 minutes	If you're slower, speed up a bit.
Q60	~75 minutes	~155 minutes	Break 1 checkpoint.
Q90	~115 minutes	~115 minutes	Halfway in time and questions.
Q120	~150 minutes	~80 minutes	Break 2 checkpoint.
Q150	~190 minutes	~40 minutes	Final stretch – keep pace.
Q180	~225 minutes	~5 minutes	Buffer for review / flagged Qs.

🚦 How to Use It

- Each 30-question block ≈ 35–40 minutes.
- Each 60-question block ≈ 70–75 minutes.
- Use breaks at Q60 and Q120 to reset.
- If you're >10 minutes behind at any checkpoint, stop overthinking and move faster.

📝 Quick Tips

- Mark tough questions and move on — don't burn >90 sec on one Q.
- Expect questions that may take longer, so balance with fast multiple-choice Qs.
- Breaks are optional, but recommended to clear your head.