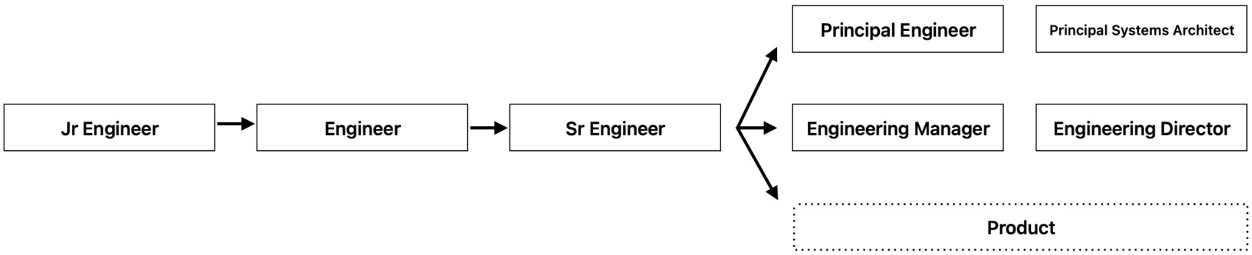
Engineering Expectations

On Apollos, engineering growth follows a clear progression. Each level **builds upon the expectations of the levels before it**. Once an engineer reaches **Senior Software Engineer**, they may continue growth along one of the following paths: **Technical Leadership, Engineering Management, or Product Leadership (not outlined here)**.



Individual Contributor Path (Technical)

Junior Software Engineer
Scope & Impact: Works on assigned tasks with close guidance. Contributes to team output while building core skills.

- Capable of working well on a team
Examples: Participates in team discussions, listens actively, collaborates on code reviews.
- Communicates effectively
Examples: Shares progress updates, asks clarifying questions, documents work clearly.
- Eager to learn
Examples: Seeks feedback, researches best practices, explores new tools.
- Gets assigned work done on time
Examples: Completes assigned tasks, asks for help early if blocked.

Software Engineer
Scope & Impact: Owns individual tasks from start to finish. Contributes reliably to team velocity and output.

Includes all expectations from Junior Software Engineer.

> **- Manages individual work assignments to completion, delivering a consistent minimum work volume independently (e.g., at least two completed Linear tasks per week).**
> *Examples:* Independently delivers features, ensures quality through testing, closes loops on reviews, maintains steady throughput.

Senior Software Engineer
Scope & Impact: Owns projects and outcomes. Supports peers through mentorship and helps ensure team success.

Includes all expectations from a Software Engineer.

> **- Manages projects to completion**
> *Examples:* Breaks down projects into clear tasks, aligns with requirements, tracks progress and ensures delivery.
> **- Helps get others unblocked**
> *Examples:* Reviews code promptly, shares debugging strategies, mentors peers.

Principal Software Engineer
Scope & Impact: Provides technical leadership within the team. Sets patterns and makes decisions that raise the quality of engineering work.

Includes all expectations from Senior Software Engineer.

- Makes good technical decisions
Examples: Chooses appropriate tools and patterns, balances trade-offs between speed and scalability.
- Does not get blocked
Examples: Finds creative solutions, escalates appropriately, leverages documentation and experts.

Principal Systems Architect
Scope & Impact: Shapes the technical direction for multiple teams. Designs systems and patterns that scale across the organization.

Includes all expectations from a Principal Software Engineer.

- Makes good technical plans and decisions at scale
Examples: Designs system architecture, anticipates scaling needs, defines standards and best practices.

Engineering Management Path

Senior Software Engineer
Scope & Impact: Entry point to management track. Demonstrates technical competence and begins influencing people and processes.

Baseline entry to the management track. Includes all expectations from Senior Software Engineer.

Engineering Manager
Scope & Impact: Leads a team of engineers. Balances delivery with people development. Directly impacts team health, velocity, and output.

Includes all expectations from Senior Software Engineer.

- Effectively leads teams of engineers
Examples: Runs effective 1:1s, sets clear goals, manages delivery timelines, fosters growth in team members.

Engineering Director
Scope & Impact: Leads multiple teams through managers. Shapes organizational strategy and ensures cross-team alignment.

Includes all expectations from an Engineering Manager.

- Effectively leads teams of managers
Examples: Coaches managers, sets organizational strategy, ensures alignment across multiple teams.