CHAPTER 14:

Titles for Programmers

Programming Management

On Web and SIP programming teams, Lead Programmers (L4) and Principal Programmers (L5) act as tech leads to Junior — Senior Programmers (L1-3). All Leads and Principals report directly to our CTO. Tech leads are responsible for:

- Reviewing a report's substantial PRs before they're shipped.
- Having a kick-off conversation with a report on every substantial project.
- Reviewing a report's on-call work and handoff after every shift.
- Providing People Ops with the key input to annual performance reviews.
- Identifying and escalating slips in performance or process to People Ops.

A tech lead on the SIP team also acts as SIP Coordinator. They're responsible for setting cycle priorities, liaising with other project teams, and maintaining general oversight of team resources.

Our Mobile programming teams (iOS and Android) have 1 manager overseeing both teams who is responsible for technical leadership and people management.

Junior Programmer (L1)

Skill

Mastery

- Proficient in basic language and framework features, but some advanced structures or features may still be unfamiliar. Occasional issues following patterns and approaches within existing code bases.

Scope

- Capable of working on tightly-scoped, routine problems.

Judgment

- Demonstrates curiosity in their craft, prioritizes their own development to better serve their team. Researches what's been done in the past and why, attempts to solve problems independently.

Tooling

- Has a rudimentary understanding of our tools (AppSignal, Buildkite, Kibana, Dash, Trek, Post Office etc).

Engagement

Ownership

- Participates in programmer on-call rotation under guidance from more senior programmers, but delivering simpler resolutions end-to-end.

Communication

- Proactively communicates what they are working on, attempts to unblock themselves but asks for help when needed. Questions mostly don't repeat.

Planning

- Prioritizes daily work against team/project needs with regular guidance. Demonstrates good time management.

Trust

- Work is thoroughly reviewed with substantial back and forth frequently needed before merging. Reads pull requests. Makes minor contributions to the improvement of system documentation.

Coachability

Feedback

- Observes and learns from the work and behavior of others. Receives constructive feedback with a growth mindset.

Programmer (L2)

Skill

Mastery
- Follows established patterns and approaches within existing code bases with ease. Writes code that is considered high quality by colleagues. Effectively and artfully applies domain design skills in normal course of work.

Scope
- Capable of working on clearly defined and scoped individual features or problems.

Judgment
- Proactively identifies and resolves problems, improves areas of the code they work on, and leaves things in better shape than they found them. Pays attention to detail.

Tooling
- Uses our most common tools in day to day work to help diagnose issues, optimize performance.

Engagement

Ownership
- Participates in programmer on-call shifts with guidance from a more senior team member. Capable of handling or triaging production issues across multiple systems with minimal assistance. Can be relied upon to lead the resolution of bugs or minor incidents related to their work or area of expertise.

Communication
- Presents pull requests that demonstrates competency with our patterns and principles, but may require oversight from senior peers. Questions to more senior coworkers are thoroughly researched and presented before posed.

Planning

- Prioritizes tasks and project work with little direction. Works confidently and effectively, persists when faced with uncertainty or ambiguity but pulls in support when needed. Identifies judo opportunities. Contributes more broadly to cycle planning within the team.

Trust

- Work is reviewed with some back and forth and the occasional need for material direction or implementation changes. Trusted to review non-critical pull requests to clarify style, readability, performance issues. Makes material contributions to documentation: features, procedures, runbooks.

Coachability

Feedback

- Receives constructive feedback with a growth mindset. Provides specific, well-considered feedback within area of expertise, when asked. Is able to disagree and commit.

Senior Programmer (L3)

Skill

Mastery

- Capable of devising new patterns and conventions within existing code bases and in greenfield development. Code remains accurate, reliable, and maintainable even when faced with challenging or complicated problems.

Scope

- Capable of taking substantial features from concept to shipping as the sole programmer (alongside a designer on the product side).

Judgment

- Has accumulated sufficient experience to be consistently sought by peers, across multiple teams, for advice. Improves existing professional standards for the team.

Tooling

- Demonstrates a comprehensive understanding of all our tools. Uses them regularly to diagnose issues quickly and calmly, even under pressure.

Engagement

Ownership

- Can recognize and raise major incidents and lead the response from inception to resolution.

Communication

- Trusted to provide actionable feedback across team pull requests in line with our principles that requires little or no oversight. Provides technical advice and suggestions during planning/scoping.

Planning

- Requires no direction to effectively manage their time, finds valuable and productive tasks to work on without prodding, and helps steer cycle planning. Identifies judo opportunities in their own work and for others. Fosters collaboration across teams.

Trust

- Work doesn't necessarily need to be reviewed, but general approach may be. Trusted to review critical or full-batch pull requests, course- corrects when required. Provides context and clarity to the engineering organization by producing and maintaining documentation / runbooks / pitches.

Coachability

Feedback

- Can provide specific, material, constructive feedback on the work of junior-mid programmers that deepens their knowledge across broad areas. Is able to disagree and commit.

Lead Programmer (L4)

Skill

Mastery

- Acts as a subject matter expert on multiple systems. Makes strategic decisions around major components.

Scope

- Capable of owning and running entire subsystems of work (Queenbee expert, Action Cable implementation). Strong track record of introducing significant improvements in system quality, performance, or scalability.

Judgment

- Has the experience, trust and respect to operate at the tech lead level. Defines professional standards. Is sought for advice across disciplines.

Tooling

- An expert in our tools. Extends and improves them over time. Can identify where tooling should be improved and oversee implementation.

Engagement

Ownership

- Can lead the resolution of any engineering problem in their discipline. Is a significant technical contributor to problem resolution; demonstrates consistent maturity in communication and demeanor under stress.

Communication

- Provides critical, actionable feedback on pull requests which ensures quality and consistency is upheld across the team. Demonstrates project management best practices. Ensures leadership is informed on team progress. Writes Heartbeats and Kickoffs.

Planning

- Can plan entire cycles for a team. Can drive the agenda, hammer scope, offer critical insight from extensive experience. Regularly challenges their team or discipline to help drive projects forward, to better deliver on team/discipline priorities.

Trust

- Major projects can be completed independently and shipped without review. Provides developing oversight to senior, mid, and juniors. Expected to review complex and high-impact pull requests across teams. Defines and maintains departmental standards by developing processes and procedures, writing style guides, recording internal screencasts.

Coachability

Feedback
- Is a formal mentor to programmers, providing specific, material, constructive feedback on work. Capable of being a technical lead on small teams. Consistently sought by colleagues for assistance with critical work or decisions. Is able to disagree and commit.

Principal Programmer (L5)

Skill

Mastery
- Invents new concepts, creates new systems, pushes the whole organization or industry forward regularly.

Scope
- Can set the strategic direction of an entire discipline, such as security, web or mobile. Consistently works on projects with high complexity. Would be the go-to person for advice on the most technically challenging problems.

Judgment
- Is seen as a leader across teams, technically and culturally. Capable of setting and directing across an entire discipline, such as SIP, Web or Mobile. Is known company-wide as a go-to expert for advice on critical work across disciplines.

Tooling
- Invents novel tools with strategic impact. Can objectively rationalise whether to build or buy, balancing tradeoffs with business strategy in mind.

Engagement

Ownership

- Fully capable of leading the most critical production incidents to successful resolution, coordinating company-wide responses and advising on public communications.

Communication

- Demonstrates long-term strategic thinking, working with leadership to communicate our engineering vision across teams and disciplines.

Planning

- Capable of running and directing larger teams for multiple cycles, and for large, long-running projects. Elevates team and company standards through developing new tooling, processes, procedures, and highly effective communication. Carries out research, testing, implementation and improvement for new systems. Pushes the whole organization forward regularly through implementing new systems and designs.

Trust

- Entire systems can be designed, developed, and shipped without the need for review. Provides developing oversight to everyone else in organization. Expected to review system architecture across multiple product suites to oversee work is produced to the highest technical standards. A leader in ensuring robust and effective best practices are understood and followed across the engineering organization.

Coachability

Feedback

- Is a formal mentor to programmers, providing specific, material, constructive feedback on work. Defines engineering-wide standards and guidance. Fully capable of leading teams of any size over an entire discipline. Consistently sought by colleagues across the organization for assistance with critical work or decisions. Is able to disagree and commit.