

Tech lead role

Overview

Lead the technical decisions of the webapp team. Provide leadership bandwidth on decisions on a higher level and on adjacent teams. Provide development bandwidth as a Senior Engineer. Making sure we provide high quality value to our customers in a fast sustainable pace maintained by technical excellence, agility and simplicity.

Typical week

Time	Monday	Tuesday	Wednesday	Thursday	Friday	
9h	Standup	Standup	Standup	Standup	Standup	
10h	Pairing/Mobbing	1-1 CTO	Reviews	Tech meeting	Reviews	Individual Contribution
11h	Tech meeting prep	Reviews	Backlog/Guidelines	Reviews	PM/Retro	Product/(dev) team
12h						Executive+
14h	Code block	Shaping	Code block	1-1 PO	Code block	White: it depends on week
15h		Comprod		QA others		
16h		Overview for comex				
17h						

Goals

- Make it easy for all devs to contribute and new team members to join the team:
 - avoiding complex/hard solutions (e.g. Vault/GraphQL)
 - promoting quick feedback (e.g. Github Actions, quick PR reviews)
 - avoiding solutions that require high amounts of coordination (e.g. semantic CSS)
 - working on DevEx (aligned local setup with docker-compose)
 - promote small PR's
 - automate away toil
- Align high-level objectives with concrete requirements through guidelines

- ISO/OKR → Miro + RSpec coverage + e2e tests + agile product development
- Make majority supported technical decisions that provide stepwise long-term value increments with a clear but adaptable long term goal (roofshots to moonshot)
 - Clearly communicate decisions and trade-offs
 - Give team members a chance to come up with solutions (tech meetings)
 - Giving big decisions a chance to get feedback and be challenged (guideline PR's open for 1-2 weeks, tech meetings)
 - Promote spikes
 - Avoiding any “long shot” solutions (should provide some value within week/month)
- Level up the technical skill of the team
 - promoting pairing
 - promoting asking questions
 - promoting automated testing
 - writing some guidelines, tips and sharing resources
 - leaving “growth” issues for others
- Streamlining our way of working with other teams
 - shared understanding of design components (Lookbook)
 - shared understanding of domain/business rules (Miro)
- Being a role model in technical work, feedback and communication
 - writing exemplary PR's
 - writing exemplary PR comments
- Building up and reiterating the requirements slowly enough so everyone can follow (e.g. juniors)
 - retrospectives
 - stand-ups
 - knowledge transfer

- mob pairing
- Tackling hard high-value issues
- Clearing the way/drafting the solution, leaving the rest for others if possible given business needs (aka growth issues)
- Tackling small technical “rest” issues (e.g. logging), glue work
- Providing development bandwidth to the dev team

Shared leadership goals

- Promote a positive culture of constructive collaboration, teamwork, verifying assumptions, openness, transparency, humbleness, blamelessness, team ownership
 - best effort through tech meetings/retrospectives
- Building up and reiterating culture change slowly enough so everyone can follow (e.g. juniors)
 - retrospectives
 - stand-ups

Challenges/Obstacles/Pitfalls

- Not have any Engineering manager:
 - People not following/undermining (any) guidelines, culture change or advancing the decisions taken, small as they may be for whatever reasons
 - Low team morale, negativity, destructivity, blaming for whatever reasons, stress
- Successful roofshots that stray us away from an interesting moonshot

What success looks like

- an environment in which a motivated developer can contribute and grow to their full potential

- being able to keep up a sustainable (fast) pace in feature development
- keeping the codebase agile to future requirements
- a productive and welcoming team culture