

Process Pull Request Template

Bonus Material from *Team as Code*

Dmitry Matyukhin

2026

Process Pull Request Template

Chapter 8 of *Team as Code* introduced the meta-process — the system that builds systems. One of its most powerful tools is treating process changes like code changes: proposed, reviewed, approved, and merged with a full audit trail.

GitLab, the \$11 billion DevOps company, pioneered this at scale. Their entire operating model — over 2,000 printed pages — lives in a public handbook. Changes go through merge requests. If something isn't in the handbook, it doesn't exist as policy. This is "Team as Code" in production.

You don't need GitLab's infrastructure. This template works in GitHub, Notion, Google Docs, or a shared wiki. The principle is the same: **no process changes without a written proposal, a review, and a record.**

From the book: "When someone wants to change how we do keyword research, they don't send an email or raise it in a meeting. They open a pull request. The proposed change is reviewed, discussed, and either merged or rejected. There's a full audit trail. The process evolves the same way the software does."

The Template

Copy this for every proposed process change. Fill in each section. If a section feels unnecessary for a small change, write “N/A” — but think twice before skipping Impact Assessment or Rollback Plan.

Process Change Proposal

Title:

Proposed by:

Date:

Affects:

(team, process, or system name)

1. Current Process *Describe how things work today. Be specific — link to existing documentation if it exists. If it doesn't exist, that's part of the problem.*

2. Proposed Change *Describe the new process clearly enough that someone unfamiliar could follow it. Include the specific steps, responsibilities, and tools involved.*

3. Rationale *Why this change? What problem does it solve? Include data if available — cycle times, error rates, team feedback. Avoid “it feels better.”*

4. Impact Assessment *Who and what is affected?*

Area	Impact
People affected	
Tools/systems changed	
Training required	
Dependencies on other processes	
Expected improvement (measurable)	
Risks	

5. Rollback Plan *If the change doesn't work, how do we revert? Every process change should be reversible — at least for a trial period.*

6. Review Checklist Before merging this change:

- Current process accurately described (or documented for the first time)
- Proposed process is specific enough for someone new to follow
- Rationale includes evidence, not just opinion
- All affected people have been consulted or notified
- Impact on other processes assessed
- Rollback plan defined
- Trial period specified (recommended: 2–4 weeks)
- Success criteria defined — how will we know this worked?
- Documentation updated to reflect the new process

Reviewers:

Decision: Approved Rejected Needs revision

Date approved:

How to Use This in Different Tools

GitHub / GitLab: Create a /processes directory in your repo. Each process is a markdown file. Changes go through pull requests using this template as the PR description.

Notion: Create a “Process Changes” database. Each entry uses this template. Use Notion’s commenting for review. Status property tracks Proposed → Reviewing → Approved → Implemented.

Google Docs: Create a shared “Process Change Log” folder. Each proposal is a new doc using this template. Use comments for review. Add a final “Decision” section when approved.

Wiki (Confluence, etc.): Use page versioning as your audit trail. Propose changes in a draft, link the draft for review, then publish when approved.

Tip: The tool matters less than the habit. The first time someone proposes a process change using this template instead of mentioning it in a meeting, you’ve crossed a threshold. The second time, it’s a pattern. By the fifth time, it’s culture.

Common Objections

“This is too bureaucratic for small changes.” Scale the effort to the change. Fixing a typo in a process doc? Just do it. Changing who approves expenses over \$1,000? Use the full template. The judgment is: *does this change affect how other people work?* If yes, it deserves a proposal.

“We don’t have time for this.” You don’t have time for the confusion, rework, and misalignment that happens when processes change without communication. This template takes 15 minutes to fill out. A miscommunicated process change costs days.

“Nobody will read it.” They will if you make it the *only* way changes happen. From the book: “If something isn’t in the handbook, it doesn’t exist as policy.”

The meta-process from Chapter 8: Observe · Define · Implement · Measure · Refine · Repeat. This template is your implementation step. The review checklist ensures you’ve observed and defined. The trial period and success criteria ensure you’ll measure and refine.