Steps to Designing a Team Process

A process describes what people do to get from idea to deliverable. The following series of 11 steps, phrased as questions, will help you design a team process as part of step 3 (Determine tactics) in Strategy 1b (Design the way of working). You might need to modify some of these steps or add new ones based on your context. If you have more than one team, follow these steps for each team. Keep it simple; the process should implement the chosen mindset (values, beliefs, and principles), cover major cases reasonably well, and be easy to remember and follow.

Note: Your team probably already has a process, and many of the basic tasks will remain unchanged. It will be tempting, and seem reasonable, to retain most of your current process around those basic tasks. However, that might be counterproductive if the chosen mindset is different from the current one.

1. Who will manage the list of portfolio items, and how?

The team works on portfolio items or portions thereof. Determine which people will decide which items the team should work on, how they will make their decisions, and how often they should revisit or reprioritize the list.

2. Who will determine and sequence the deliverables, and how?

Ideally, the portfolio includes intended customer and business outcomes. Determine who will decide and sequence the deliverables that the team should produce to accomplish those outcomes. Also decide *how* they will make their calls.

3. Who will break down work items, and how?

Portfolio items usually move through the system in pieces, generically called *work items*. Some of these might be tasks (such as for specification, design, and coding) and some might be end-to-end portions of the portfolio item. Choose who will identify the work items, and how. The "how" question is particularly important if the values and principles align with Lean/Agile, which call for making those items quite small and outcome-focused.

4. What does "done" mean?

The team will strive to get each work item to "done." Define what that means – which attributes or qualities should a "done" item have that will allow the team to move on to the next thing, not concerned about loose ends?

5. What should the workflow of a typical work item be?

Identify a sequence of states, starting from "ready" (sometimes called "to do," "next," "planned," or "committed") and reaching all the way to the item being "done." Thinking about examples of current work might help you elucidate the workflow, by asking "What do we do first? Then what? And then what?"

The more granular the states in your workflow, the better you'll see where work stalls or piles up. That information will come in handy when you work on stabilizing the system (strategy 2b). With too many states, however, the administrative effort to accurately track each item's state becomes undesirable overhead. As well, a multi-state workflow might be unique to a particular class of work items, requiring you to have several distinct workflows.

6. How will you visualize the work?

Determine how to represent the content and status of current and upcoming work items. The typical artifact for this is a board (whether physical or electronic): columns represent workflow stages, and items move from left to right as they proceed through the workflow. It's often useful to subdivide boards horizontally into "swimlanes" to call out different classes of items, such as new value-adding work, defects, and improvements. For each work item, provide minimally necessary information such as its heading, any date expectations, and which team members are currently working on it. Expect this artifact to evolve, so start simple.

7. How will you constrain the team's work intake?

Determine enabling constraints on work intake – ones that will nudge the team to work strategically, efficiently, and sustainably. Based on your values, beliefs, and principles, you might choose sprints, work-in-process limits, or something else.

8. Which feedback loops should you have for work content?

People need feedback on their decisions and actions to make good product-impacting decisions. Use the following prompts to design practical, effective, and realistic feedback loops.

- What kinds of questions should you ask, and of whom?
- How frequently, or at what points, should you be asking those questions?
- What sort of feedback should you seek from colleagues before going outside for more?
- What needs to happen for feedback-receivers to feel safe both requesting it and hearing it?
- What needs to happen for feedback-givers to feel safe enough to say what they really think?

9. What sort of impediments do you foresee, and what will you do about them?

Every team faces impediments (obstacles) to their progress or productivity. Rather than being reactive – acting on impediments only after noticing them – think ahead to likely impediments and identify reasonable responses. Incorporate these ideas in your process design. The following ideas might help:

- Some impediments are due to workflow or team design. For instance, work can't be completed or moved to the next stage, work can't be started due to missing information, a deliverable is waiting for rework, or a team member is waiting to hear back from an expert or from someone outside the team.
- Some impediments arise simply from working within an organization. For example, a sign-off is needed, organizational procedures and business practices complicate or delay planning, or critical stakeholders are too busy to give the team feedback.
- Many impediments have to do with the team's and organization's being made up of humans. Examples include loss of focus due to excessive interruptions, incorrect progress caused by faulty assumptions or partial communication, and missed actions ("dropped balls").

10. How will you get finished deliverables into customers' hands?

Review the types of deliverables you produce and decide when you'll deliver each type: when it's ready, when you've amassed a few, or during specific windows of opportunity. How will the users know they're ready? How will you know that they're actually using those deliverables and benefiting from them?

11. Which touchpoints will the team have?

Depending on the chosen mindset, the team will need various touchpoints. Some will include only the team, and others will include fellow system people or stakeholders. Some might take place anytime there's new information or a decision to make, and others on a cadence. Common touchpoints include:

- Determining outcomes and/or deliverables (e.g., discovery)
- Choosing work for the next little while (e.g., sprint planning or queue replenishment)
- Assessing progress, risk, and impacts (e.g., sprint review)
- Micro-planning (e.g., daily huddle)
- Improving the team's way of working (e.g., retrospective)

Clearly define the purpose, deliverables, and process of each touchpoint, basing your choices on the values and principles.