

# Getting started with Process

Last Modified on 16/07/2025 10:21 am AEST

When approaching automation, it is common to default to a **linear, task-based mindset**:

*"First we click this, then we show that, then we get approval..."*

While this step-by-step approach is useful later on, it is not where you start if you recall from our previous lesson on workflow automation. To build effective, scalable processes, you need to **zoom out** and think in terms of **systems**—interconnected flows that serve a broader business objective.

Start with the Outcome: What Are You Trying to Achieve?

Ask:

- What is the **business objective** of this process?
- What **triggers** it?
- What does "done" look like?
- Who are the **actors** (people/systems) involved?

This helps define the system boundary, what is in scope, what is not, and what a complete process means. Note that you may not be automating all the process so for our example a complete process is what we aim to automate within REDINow only.

## Breaking the Process down into logical steps

### 1. Identify Stages – Major Phases with a Clear Purpose

Ask:

- What are the *logical segments* of this journey?
- Where do **responsibility, intent, or outcome** shift?

Each stage should:

- Represent a **clear transition** (e.g., from creation to approval).
- Answer the question: *"What needs to happen before we move on to the next big step?"*

**Example (Employee Onboarding):**

- Stage 1: Submission
- Stage 2: Approvals
- Stage 3: Onboarding

### Decision framework to split a Stage

Decision Test	Key Question to Ask	If "YES" → Create a New Stage	If "NO" → Keep in the Same Stage
1. Purpose	Do the activities now serve a different business purpose?	The high-level goal changes (e.g., "standard approval" → "executive exception").	The activities still pursue the same overarching goal.
2. Milestone	Would stakeholders call this a distinct milestone worth tracking?	Crossing this point is a recognized hand-off or checkpoint.	It's just another task on the way to the same milestone.
3. Ownership & SLA	Do we need different owners, SLAs, or analytics beyond this point?	Different team, clock, or KPI applies (e.g., exec SLA vs. manager SLA).	Same team, same clock, same metrics continue to apply.
4. Frequency	Does this path occur frequently enough to merit its own spotlight?	Happens often (≥ ~10 % of cases) so visibility is valuable.	Rare edge-case; adding a stage would clutter the happy path.
5. Complexity	Does combining everything here make the stage hard to read or manage?	The stage is turning into a "junk drawer" of logic.	Diagram remains clear and maintainable with activities grouped.

In our example, we chose to keep the stages simple, all approvals can occur under the same stage, with the assumption that management are not bothered by emphasising the CEO approval as it is very rare.

## 2. Define Blocks – *Sub-Processes Inside Each Phase*

Ask:

- Within this phase, what **distinct tasks or decisions** need to occur?
- Are there **alternative paths** (e.g., rejection, escalation)?
- Does anything need to happen **sequentially** here or perhaps in parallel?

Each block should:

- Handle a **specific outcome** (e.g., “Manager Approval” or “Rejection Handling”).
- Group steps that are tightly coupled by purpose.

**Tip:** Think of blocks as **micro-flows** that can be reused, rerouted, or extended.

### 3. Detail the Steps – *Atomic Tasks or Logic Units*

Ask:

- What **actions** or **rules** need to be executed in order?
- Is this done by a human or the system?
- What **data** is being moved, changed, or validated?

Each step should:

- Be a **clear, actionable unit** (e.g., “Submit Form”, “Send Notification”, “Update Status”).
- Do just **one thing**, cleanly and predictably.

**Tip:** If you can't describe the step with a clear verb-object combo ("Approve request", "Update record"), it may need to be broken down further.

In summary:

1. What is the goal? → (Define the process)
2. What has to happen to get there? → (Identify stages)
3. What does each phase need to do? → (Define blocks)
4. What are the specific actions? → (Design steps)