Task management

Tom Rochette <tom.rochette@coreteks.org>

November 30, 2025 — e305e877

1 Steps to create a task

- Give it a title
- Decide if it's important or not
- Decide if it's urgent or not
- Estimate the effort required to accomplish the task (in hours)
- Estimate the value the task brings (in $)
- Give it at least one label
- Give it an assignee
- Give a description if necessary
- Set a deadline if possible
- Set status to Unprioritized (ideally automatically/default)
- Indicate who requested the task
- Indicate which users/how many users would be affected by the task

2 Statuses

- `Unprioritized`: New task are created as unprioritized, so as to avoid having to put them in a state immediately when creating them.
- `Won't do`: A task that ends up not being relevant or useful will not be acted further upon and set to "Won't do".
- `Backlog`: A task that is not of priority at the moment but something that would be worthwhile to do at some point in the future (although maybe never due to other priorities).
- `Queue`: A task with a priority that is not high enough to be done immediately but that should be done in the near future.
- `Scheduled`: A task that has been scheduled to be done by a specific date.
- `Today`: A task that is scheduled to be done today. Tasks picked from the queue are moved to today.
- `In progress`: Tasks that are were in progress yesterday but aren't blocked, waiting, or done are moved back to today, the queue, or the backlog.
- `Blocked`: A task that is blocked by something else. The task should be moved back to today when it is unblocked.
- `Waiting`: A task that is waiting for something to happen before it can be worked on. The task should be moved back to today when it is unblocked. The difference between a blocked task and a waiting task is that a blocked task is blocked by something that is not under your control, while a waiting task is blocked by something that is under your control.
- `Done`: A task that has been completed and does not require further action.

3 During a workday

- Go through the tasks in the Do section, then the Decide section, then Delegate

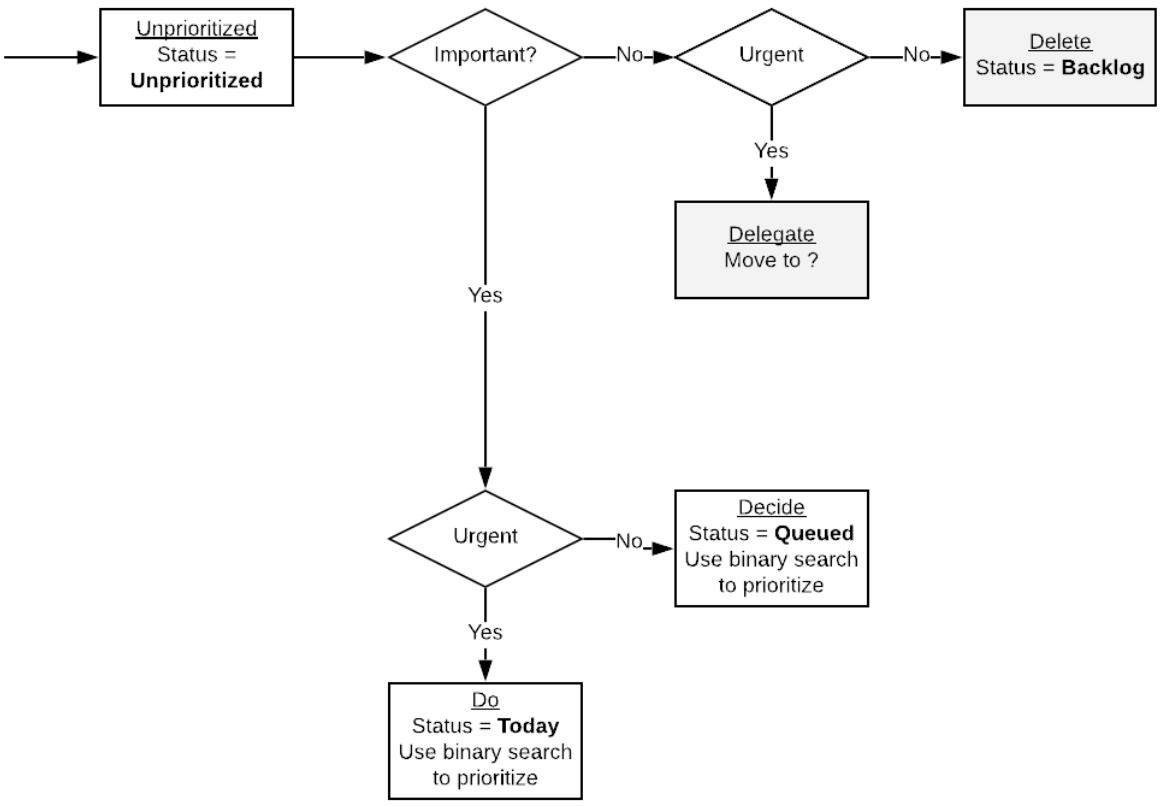

Figure 1: Task creation

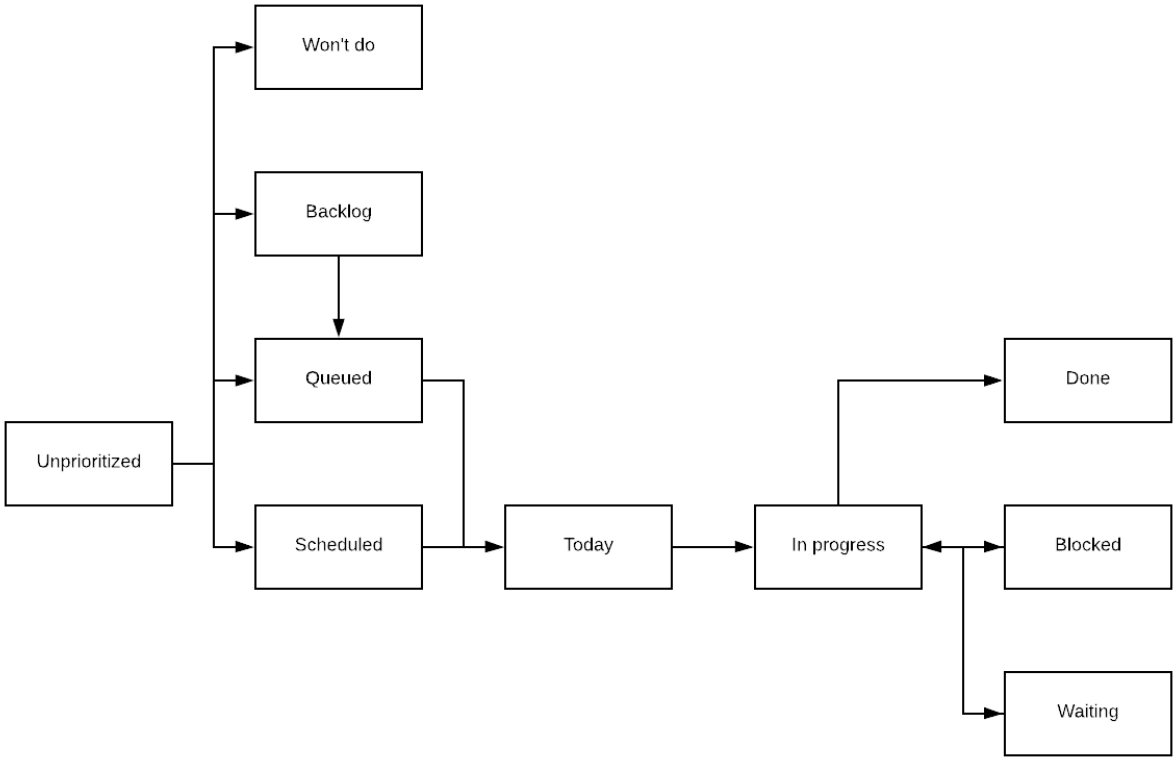

Figure 2: Task states

4 Important/Urgent

Use Eisenhower matrix to determine task importance/urgency

- Tasks that are not important/not urgent are moved to the backlog.
- Tasks that are not important/urgent are delegated to someone else.
- Tasks that are important/not urgent should be scheduled.
- Tasks that are important/urgent should be done ASAP.

5 Effort/Value/ROI

Using the estimated effort and estimated value of a task, you can compute the return on investment (ROI) of the task as estimated value divided by estimated effort.
For example, a task you estimate is worth 100 $ and takes 2 hours to complete will have a ROI of 50 $/h.

Use the ROI of your tasks to prioritize them. You will want to complete the tasks which are likely to have the best return on investment.

The ROI metric will also give you clues about the task you should probably not spend your time on. If you are paid 50 $/h and a task has a ROI less than 50 $/h, then it should probably be moved to the backlog and only reconsidered if its ROI changes.

6 Task processing

- `Unprioritized` tasks should be moved to `Queued`, given their priority compared to already queued tasks.
- You should attempt to keep the `Unprioritized` tasks count to 0.
- A task that is `Queued` should have a deadline date.
- Work on tasks `In progress` first, then take tasks from `Today`, then from `Queued`.
- When moving a task to `Blocked` or `Waiting`, add a comment indicating why it is blocked/waiting and what needs to happen for it to be unblocked.

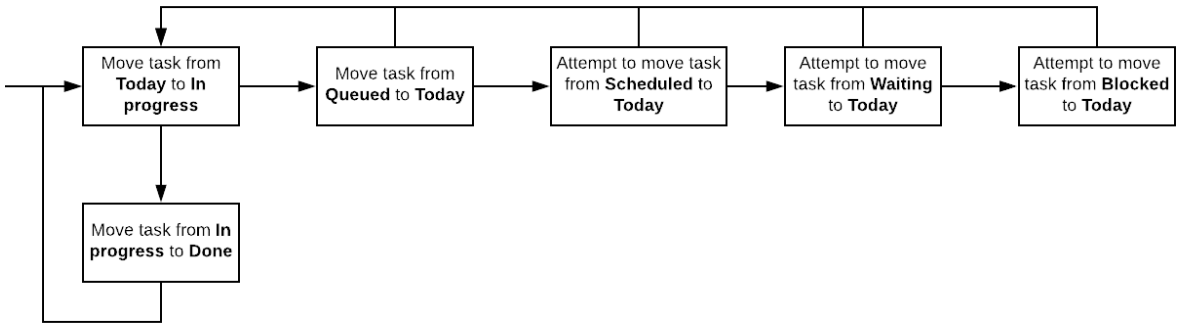

Figure 3: Task processing loop

7 Task management progression

If you've never done task management before, I suggest you do not start by applying all the above at once. First, start by simply recording the tasks you need to get done (title only).
Once you've recorded most of the tasks you have to deal with, start giving them an important/urgent assignation.

It will help you rapidly determine which tasks should be done and which ones are nice to have, but not critical, or even not really useful if you think about it.
Once you are able to have this information for most of your tasks, then you should start giving them effort and value estimates.
Once you've reached this point, you will have a much better grasp on the importance of your tasks, as well as their potential impact in terms of value, as well as to the amount of effort it will require from you.

8 Fields

- `Title` (string): a high level description of the task in one sentence.
- `Labels`: a set of short identifiers relevant to the task.
- `Status`: the status of the task (see Statuses above).
- `Assignee`: who is assigned to complete the task.
- `Important`: is it important or not?
- `Urgent`: is it urgent or not?
- `Estimated effort`: effort to complete the task (in hours).
- `Estimated value`: estimated value of completing the task (in $).
- `Description`: a long description of what the task is about, possibly containing more context and a TODO list.
- `Deadline`: a time by which the task should be done or may result in the task losing value or causing issues.
- `Who requested`: the person who requested the task to be completed.
- `Who is impacted`: the people that will be impacted by the completion of the task.

9 Notes

- Tasks should not remain in the `In progress` state for extended periods of time. After approximately a week in that state, I would suggest re-evaluating whether this task is still `In progress` or should be moved back into the `Backlog`, `Queued`, `Scheduled` or `Today` state. The same should apply to the `Waiting` and `Blocked` states.

10 References

- Given that you define a return on investment (ROI) on a task, when should you stop working on a task and abandon it given its cost?
- How do you prioritize things when there are so many of them competing against one another?