WORKM8 CONSULTING · ADD-ON FOR SERVICEM8

Project Checklist guides

Four short reads on the ideas behind Project Checklist, the project board inside the ServiceM8 job card. Read them in order the first time; after that, dip in.

1. Is this a job, or is it a project?
2. Board or Timeline: which one for this project?
3. Setting up your first project
4. Working the plan when it changes

Online, with the interactive demo and the FAQ: workm8consulting.com/help/project-checklist

PROJECT CHECKLIST GUIDES · 1 OF 4

Is this a job, or is it a project?

Most work in ServiceM8 is a job and should stay that way. Here's how to spot the work that needs more than a job can hold.

ServiceM8 is built around jobs. You quote a job, schedule it, do it, invoice it. For most of the work most service businesses do, that's exactly right, and nothing here is asking you to change it.

But some work doesn't sit comfortably inside a single job. If you've ever finished a week and couldn't say off the top of your head where a particular job was up to, you've met it.

The quickest way to tell

Are you tracking it somewhere else?

A spreadsheet. A notebook in the ute. A whiteboard in the office. A Trello board someone set up once. A group chat where the real story lives.

If a job has quietly grown a second home outside ServiceM8, that job is asking to be a project. That second list exists because the job has more moving parts than the job card can hold, and someone needed somewhere to put them.

That's the tell, and it's usually enough on its own.

Three signals

If you want to be more deliberate about it, look for these.

- **It takes more than one visit.** You'll be back, maybe several times, and something has to happen between the visits.
- **More than one person touches it.** Different trades, different days, and someone needs to know who's doing what and whether the last bloke finished.
- **You can't hold it in your head.** There are enough steps that you're writing them down, or worse, meaning to.

One of these on its own might not be much. Two or three together, and you've got a project.

What that looks like in practice

These are usually projects:

- A bathroom or kitchen renovation — several trades, a fixed sequence, a date the customer is counting on
- A solar install — site assessment, approval, install, inspection, grid connection, spread over weeks and often several visits
- A switchboard upgrade and partial rewire — staged work, an outage window, follow-up
- Any fitout with a handover date and a defect list at the end

These are usually not:

- A blocked drain, a dead hot water unit, a fault callout — one visit, one person, done
- A routine service or scheduled maintenance visit
- Anything you'd finish and invoice the same day

The line isn't the dollar value. It's whether the work has parts that need tracking separately.

THE WORK	MORE THAN ONE VISIT	MORE THAN ONE PERSON	CAN'T HOLD IT IN YOUR HEAD	SO IT'S A...
Bathroom or kitchen renovation	✓	✓	✓	Project
Solar install	✓	✓	✓	Project
Switchboard upgrade and partial rewire	✓	✓	✓	Project
Fitout with a handover date and a defect list	✓	✓	✓	Project
Blocked drain, dead hot water unit, fault callout	–	–	–	Job
Routine service or scheduled maintenance visit	–	–	–	Job
Anything you'd finish and invoice the same day	–	–	–	Job

The three signals against the examples. Two or three together and it's a project; none, and it's a job.

What happens when you mark a job as a project

This is the part worth understanding, because it's simpler than people expect.

Open the job in Project Checklist and click the **board icon** next to the job name in the header. Grey means it's an ordinary job; blue Project means it's a project. That's it.

You can also do it from the **Project Summary** — the dashboard listing all your projects — using **Mark as project** on any job in the list.

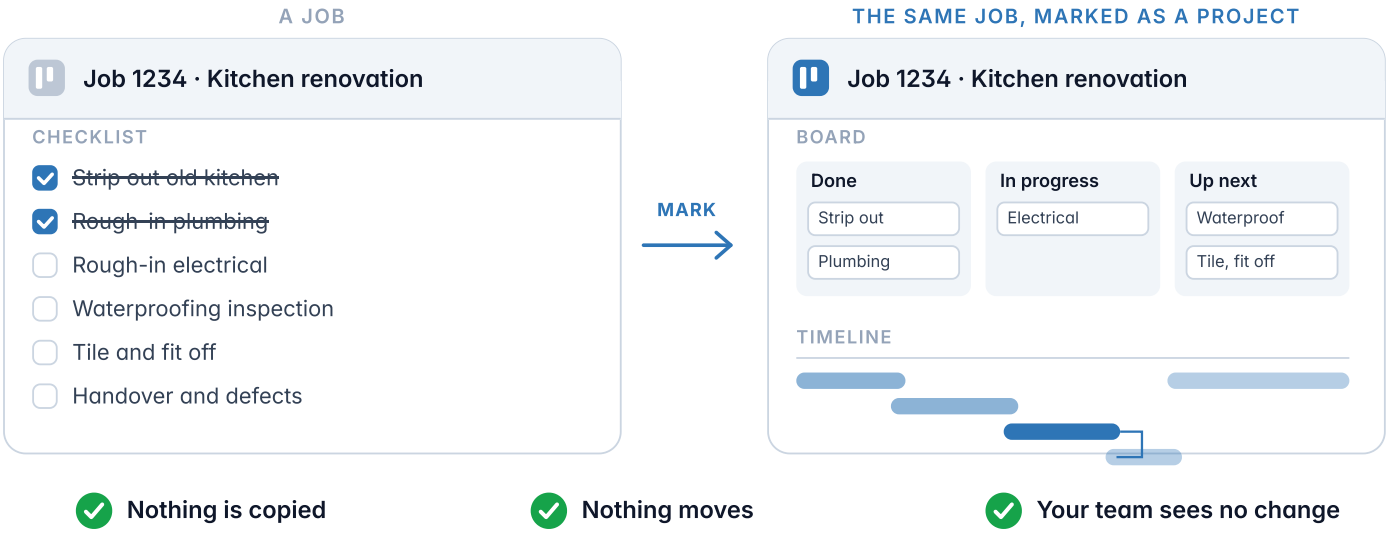
Here's what it does:

- The checklists already on that job become **lists**, and the items on them become **tasks**.
- You get new ways to look at them — the **Board** and the **Timeline** among them.
- The job appears on the **Project Summary** alongside your other projects.

And here's what it doesn't do:

- **Nothing is copied.** No second version of the work is created anywhere.
- **Nothing moves.** The job stays exactly where it is in ServiceM8.
- **Your team sees no change.** A tech opening that job in the ServiceM8 app sees the same checklist they always did. When they tick something off, it ticks off on your Board, because it's the same record. There's no syncing, because there's nothing to sync.

That last point is the whole idea. Project Checklist doesn't build a separate project system beside ServiceM8. It gives you better ways to look at the checklists you already have.



One click on the board icon. The checklist items are the same records; the Board and the Timeline are new ways to look at them.

When the work is split across several jobs

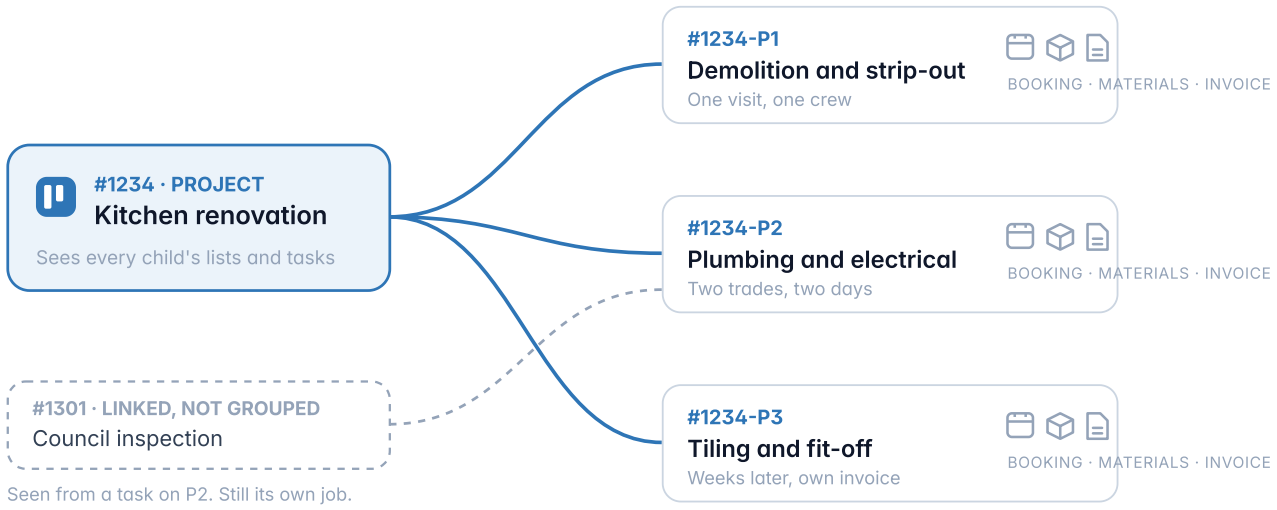
Plenty of project work is. A solar install gets quoted as one thing and then split, because the site assessment is one visit by one person, the install is two people for two days with a truck full of materials, and the grid connection happens weeks later. Each of those wants its own booking, its own materials, and often its own invoice.

That's not a problem to work around. That's just how the work is.

So a project can be more than one job. You pick the job you think of as the head of the work, mark it as a project, and then add the others underneath it as **child jobs**.

Each child stays a completely normal ServiceM8 job. Its own bookings. Its own staff. Its own materials and labour. Its own invoice. Nothing about it changes.

What you get is one place to see all of it. The parent project shows you the lists and tasks across every child job at once, so you can look at the whole install rather than four separate job cards.



One parent, as many children as it needs, one level deep. Each child is still an ordinary ServiceM8 job with its own booking, materials and invoice. A linked job is only looked at, never grouped.

Two ways to add one

From the project, open the three-dot menu ⋮ and choose **Child jobs...**

- **Create a new job** — same customer and address as the parent, and it's numbered off the parent so the relationship is obvious in ServiceM8. A parent numbered 1234 gets children numbered 1234-P1, 1234-P2 and so on.
- **Attach an existing job** — for work that's already been quoted or booked as its own job.

A job that's part of a project shows **Part of #1234**

in its header, with the number as a button that opens the parent. To take one out again, use **Remove from project**. The job goes back to being an ordinary job, and nothing is deleted.

One level, on purpose

A project is one parent with as many children as it needs. Children don't have children of their own.

That's a deliberate limit rather than something not built yet. Trade work has stages, not stages of stages, and a tree of jobs inside jobs would be harder to read than the four job cards you started with.

A lighter option: linking

If you only want to *see* another job from a task — not group it into the project — use **Link to job** on the task instead. That shows you the other job's current status and address without making it part of anything. Useful when a task depends on work someone else is handling.

Just try it

It's reversible. Click the board icon

again — or choose **Remove from projects** on the Project Summary — and you're back exactly where you started. Nothing is deleted, nothing is lost, because nothing was ever moved. If the project had child jobs, they go back to being ordinary jobs too.

So pick the job that's been living in a spreadsheet, mark it, and have a look. You'll know in a couple of minutes whether it belongs there.

One thing worth avoiding: don't mark everything. If every job is a project, the Project Summary stops telling you anything. Mark the handful of jobs that genuinely need watching, and leave the rest as jobs.

Board or Timeline: which one for this project?

Project Checklist gives you several ways to look at a project. Two of them are where you'll spend your time, and one question tells you which.

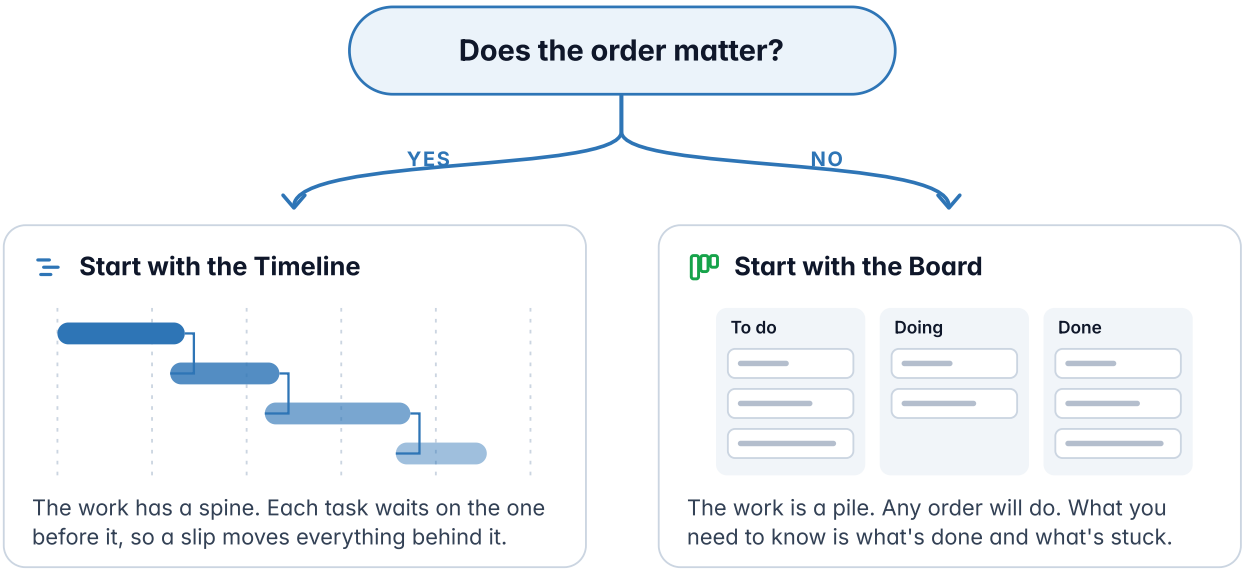
You've marked a job as a project. There are a few ways to look at it now, and two of them are where you'll actually spend your time — the Board and the Timeline. It isn't obvious which one you want.

There's one question that sorts it, and you can answer it in about five seconds.

Does the order matter?

That's it. That's the whole question.

Think about the work in this project. Does it have to happen in a particular sequence, or could you tackle any part of it on any day?



Five seconds. Sequence, so the Timeline. Pile, so the Board. Both views show the same tasks, so nothing is lost if you pick the other one.

If the order matters, start with the Timeline

Some work has a spine. You can't waterproof before the rough-in. You can't connect to the grid before the inspection passes. One thing has to finish before the next can start, and if something slips, everything behind it slips too.

That's what the Timeline is for. Each task becomes a bar on a calendar, so you can see:

- how long the whole project runs
- what's happening in which week
- which tasks are waiting on other tasks
- what a delay actually costs you

If there's a handover date, or subbies who need to be booked in the right order, you want the Timeline.

If the order doesn't matter, start with the Board

Other work is just a pile. Thirty defects to clear before handover. Forty sites to inspect this quarter. A maintenance contract's worth of callouts. Any of it could be done in any order — what you actually need to know is what's finished, what someone's on, and what's stuck.

That's the Board. Tasks sit in lists, and you move them along as the work progresses. It's the view that answers "where are we up to?" at a glance.

If you've used Trello or a whiteboard with sticky notes, you already know how this works.

Some examples

THE WORK	WHICH ONE	WHY
Bathroom renovation	Timeline	Demo, rough-in, waterproof, tile, fit-off. The sequence is fixed.
Handover defect list	Board	Thirty items, any order. You just need them cleared.
Solar install across three visits	Timeline	Assessment, then approval, then install, then inspection.
Quarterly compliance round, 40 sites	Board	Every site is independent. It's about what's done.
Switchboard upgrade with an outage window	Timeline	The date is fixed and everything works back from it.
Warranty callbacks on a finished build	Board	They arrive when they arrive.

Most projects are actually both

Here's the thing nobody tells you: real projects usually have both kinds of work in them.

A bathroom renovation has a strict sequence up front — you genuinely cannot reorder demo, rough-in, waterproofing and tiling. But once the work is done, you've got a defect list, and that's just a pile of small jobs in no particular order.

Same project. Two shapes of work. Both in Project Checklist at the same time.

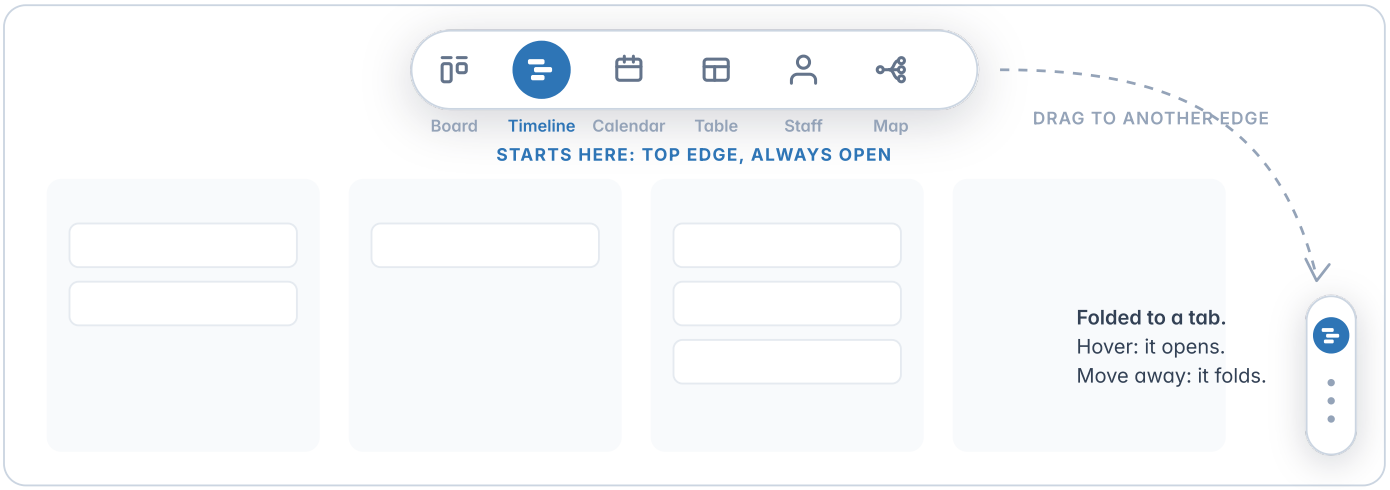
You don't have to choose. Every task lives in one place, and the Board and the Timeline are two ways of looking at the same tasks. Nothing is duplicated. Nothing needs syncing. Change something in one view and it's changed in the other, because it was never two things to begin with.

So the practical answer for most projects is: **plan the sequenced part in the Timeline, work the day-to-day on the Board, and switch whenever it suits you.**

You can't get this wrong

Because the views are just lenses on the same work, picking the "wrong" one costs you nothing. You switch views from the dock and everything's still there.

The dock is the small row of icons floating over the board: Hover it and it opens out; click an icon to switch. If it's ever in your way, drag it — it snaps to whichever edge you let go near, and stays there.



Left, right or bottom edge: the dock folds away and opens on hover. Top edge: it stays open.

So don't overthink it. If the project has a deadline and a sequence, open the Timeline. If it's a list of things that need doing, open the Board. You'll know within a minute whether the other one suits you better.

The other views

Two views is where you'll live. The rest are there for when you want something specific:

- **Calendar** — the plan laid out on a month, week or day. Handy for booking work in and for spotting the weeks that are too full.
- **Table** — every task as a row, sortable, with the columns you choose. This is the one to open when someone asks a question the Board can't answer at a glance.

- **Staff (Board)** — the same board, grouped by who the work belongs to.

And if your project has child jobs, you get one more:

- **Map** — the whole project drawn as a branching picture. The job at the root, branching out to its lists and its child jobs, each of those branching to its own lists. It answers "what is this project actually made of", which is a hard thing to see any other way once the work is spread across several jobs.

You don't need any of them to get started, and none of them change the answer to the question above.

Setting up your first project

From a marked job to a working board. Build it by hand, write the whole thing as plain text, or save a finished project as a template for next time.

You've marked a job as a project. This page gets you from there to something you can work with.

Five minutes by hand. Thirty seconds with a template. And once you've run one project properly, you can turn it into a template with two clicks, so every job like it after that starts already planned.

What you're looking at

Any checklists already on that job are sitting there as **lists**, and the items on them are **tasks**. If the job had no checklists, you'll see an empty board. That's normal.

Every list is a column. Every task sits in one list. That's the whole structure.

Adding a list

Choose **+ Add checklist**, give it a name, save.

Name lists for the **state** the work is in, or the **stage** it belongs to. That's the Board-or-Timeline decision showing up in your list names:

- **By state** — In Progress, Up Next, On Hold, Done. Work moves left to right as it progresses.
- **By stage** — Site Assessment, Approvals, Install, Commissioning. Work belongs to a stage and stays there.

Solar installs and renovations usually want stages. Defect lists and maintenance work usually want states. If you're not sure, start with states — it's easier to read at a glance, and you can rename later.

Adding tasks

Choose **Add task** on any list, type the name, save.

Write task names the way you'd say them out loud. *Book crane*. *Chase approval*. *Waterproofing inspection*. If you're writing a paragraph, that's two tasks.

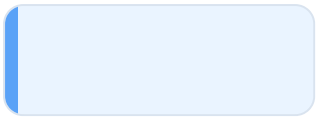
Don't use the same task name twice in one project. Two tasks both called *Site inspection* get confusing fast, for you and for anyone recording time against them. Make them specific — *Site inspection (pre-install)* and *Site inspection (final)*.

Colouring your lists

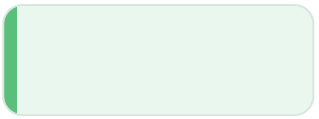
Every list can have a colour, and it's not decoration — colour is how you read a busy board without stopping to think.

Available colours: **sky, mint, sun, rose, lavender, sand, stone**.

Cool · work that is moving



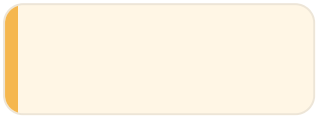
Sky
`[list colour: sky]`



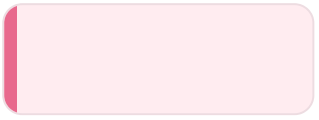
Mint
`[list colour: mint]`

Lavender
`[list colour: lavender]`

Warm · work that needs attention



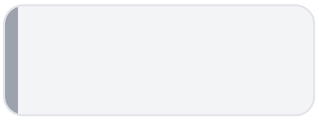
Sun
`[list colour: sun]`



Rose
`[list colour: rose]`

Sand
`[list colour: sand]`

Grey · work that is finished



Stone
`[list colour: stone]`

The seven list colours, exactly as the app draws them. One pattern that works: cool for moving, warm for attention, grey for done.

A pattern that works: cool colours for work that's moving, warm for work that needs attention, grey for work that's finished. After two or three projects you'll read the shape of a board before you read any of the words.

Tidying a long board

Once a project has a dozen lists, the board gets wide. Two things help.

Nest a list under another. Open a list's menu ⋮ and choose **Nest under…**, then pick the list it belongs beneath. It tucks in under its host and you can fold the whole group away. Good for grouping *Bathroom*, *Kitchen* and *Laundry* under *Second fix*.

You can un-nest at any time from the same menu. Nesting goes one level deep — a nested list can't itself host others, and the menu will tell you when that's why it isn't offering.

Turn a list into its own job. If a list has grown into work that deserves its own booking, materials or invoice, open its menu ⋮ and choose **Convert to child job…** — the list becomes a child job of this project, keeping its tasks.

That's the same child job idea from Is this a job or a project?, reached from the other direction: instead of adding a job and filling it, you build the work first and split it off when it's clearly its own thing.

The fast way: write the project as text

Go to the **three-dot menu** ⋮ → **Manage templates** and create a template. You write the project as plain text and it builds every list and every task in one go.

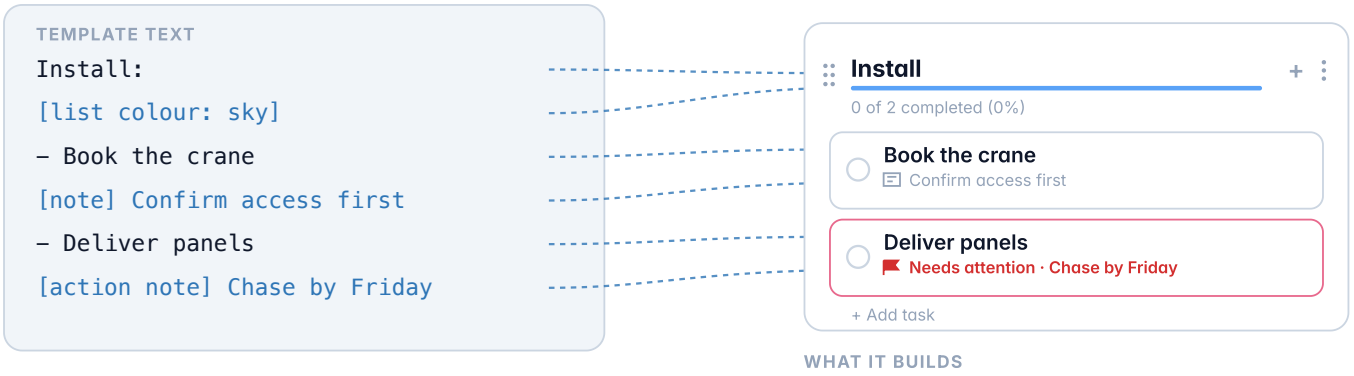
> **Let an AI assistant write it.** Download the template guide and give it to ChatGPT, Claude or Copilot with a description of the job. It knows every token below and will hand back a template ready to paste.
>
> Download the template guide
>
> or point your assistant at `https://workm8consulting.com/help/project-checklist/template-guide`

The basic format is two rules:

```
List name:
- A task
- Another task

Second list name:
- A task in the second list
```

Lists end with a colon. Tasks start with a dash.



Colon makes a list, dash makes a task. The bracketed tokens are extras: the list colour, shown as the line under the list name, and a note or an action note on the task.

That's the same format ServiceM8 uses for its own checklist templates, so **anything you've already built in ServiceM8 pastes straight in and works.** Everything below is extra — ServiceM8 ignores it, Project Checklist picks it up.

Colours and notes

```
Install:
[list colour: sky]
– Book the crane
  [note] Confirm access with the site supervisor first
– Deliver panels
  [action note] Chase the supplier if not confirmed by Friday
```

- `[list colour: sky]` goes on its own line straight after the list name. `[colour: sky]` works too.
- `[note]` adds a normal note under a task.
- `[action note]` adds a note that flags the task as needing attention, so it appears in **Needs attention** until someone deals with it.

Nesting a list

If you want a list tucked under another one, say so under its title:

```
Second Fix:

Bathroom:
[nest under: Second Fix]
- Waterproof
- Tile

Kitchen:
[nest under: Second Fix]
- Install cabinetry
```

It goes on its own line under the list title, in the same place as the colour line, and you can use both. The host name is just the list's name, spelled as you wrote it — capitals don't matter.

If the host doesn't work out, the list simply stays flat and you're told why. Nothing is refused, and the rest of the template still builds. You'll get a warning naming the line if the host isn't a list in the template, if you've nested a list under itself, or if the host is already nested under something else — lists nest one level only.

Planning the work in the template

If the order matters, you can build the schedule at the same time.

Give a task an id, so other tasks can wait for it

```
- Dig trench [id: dig]
```

The id goes on the task line itself, after the name. Keep it short. It's only used inside the template — you'll never see it again once the project is built.

Say how long it takes, and how much work it is

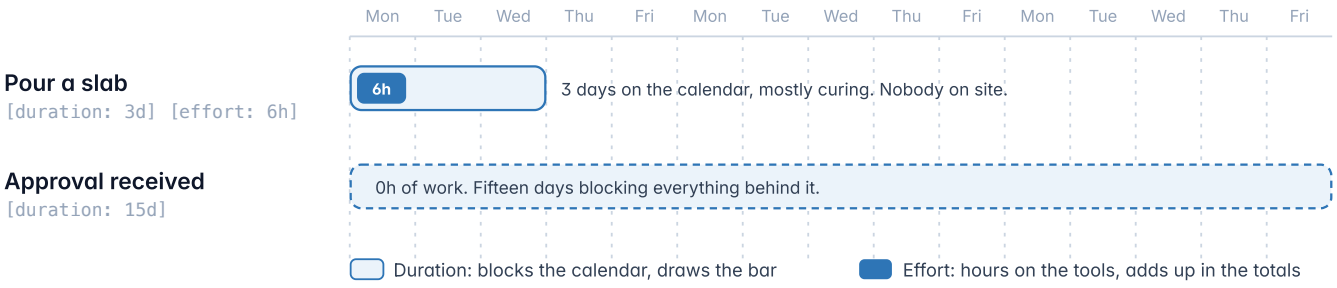
On the line underneath the task:

```
- Dig trench [id: dig]
  [duration: 3d] [effort: 6h]
```

- `[duration: 3d]` — how many **working days** the task occupies on the calendar. The `d` is optional; duration is always in days.
- `[effort: 6h]` — how many **hours of actual work** it involves. **Always give a unit**, `h` or `m`. A bare number is refused, because six could mean six hours or six minutes and guessing wrong would quietly wreck every total on your board.

These are two different things, and the difference matters.

Pour a slab and it might tie up three days but only involve six hours of labour, because most of that time is curing. A grid connection approval might sit there for three weeks with no work at all. Duration is how long it blocks the calendar. Effort is how long someone is actually on the tools.



Two different things. Duration is calendar time and draws the Timeline bar. Effort is work time and feeds the project's totals. A task can have one, both, or neither.

Duration draws the bar on your Timeline. Effort tells you how much work is really in the project. A task can have one, both, or neither.

Say what has to happen first

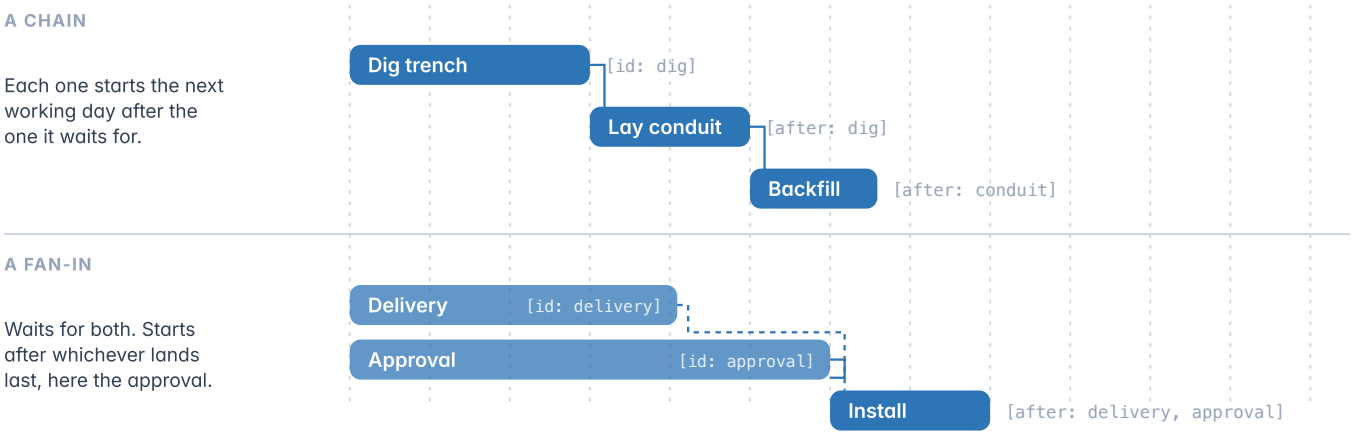
```
- Lay conduit [id: conduit]
  [after: dig] [duration: 2d]
- Backfill
  [after: conduit] [duration: 1d]
```

`[after: dig]` means this task starts once *Dig trench* is finished.

To wait on more than one thing, list them:

```
- Install and mount
  [after: delivery, approval] [duration: 2d]
```

That task starts once **both** are finished — whichever lands last. Real jobs fork and rejoin constantly, so this comes up quickly.



Give a task an id only when something waits for it. To wait on more than one thing, list them: the task starts once all of them are finished.

Choosing when the project starts

By default, dates are worked out from **the day you apply the template**. That's what you want for a reusable template — apply it in March and it plans from March.

For a project with a known start date, put it on its own line at the top:

```
[start: 2026-10-05]

Site Assessment:
- Site visit and measure [id: visit]
```

Your working week

Dates skip non-working days. By default that's Saturday and Sunday, but plenty of trades work Saturdays — solar crews especially.

Set yours in Settings → Working Week. Tick the days your team works. If you don't, a task marked `[duration: 3d]` starting Thursday will finish on Tuesday, when for you it should finish on Monday.

Worth doing once, before you build your first planned template.

A worked example

A solar install, start to finish:

```
Site Assessment:
[list colour: sky]
- Site visit and measure [id: visit]
  [duration: 1d] [effort: 3h]
- Prepare design and quote [id: design]
  [after: visit] [duration: 2d] [effort: 4h]

Approvals:
[list colour: sun]
- Submit grid connection application [id: gridapp]
  [after: design] [duration: 1d] [effort: 1h]
  [action note] Chase if nothing back within 10 working days
- Approval received [id: approval]
  [after: gridapp] [duration: 15d]

Install:
[list colour: mint]
- Deliver panels and inverter [id: delivery]
  [after: approval] [duration: 1d]
- Install and mount [id: install]
  [after: delivery] [duration: 2d] [effort: 14h]

Commissioning:
[list colour: lavender]
- Electrical inspection [id: inspection]
  [after: install] [duration: 1d] [effort: 2h]
- Grid connection and meter change
  [after: inspection] [duration: 5d]
```

Apply that and you have a planned project with dates on the Timeline.

Look at *Approval received* — fifteen days of duration and no effort at all. Nobody works on it. It just sits there blocking everything behind it. That's usually the task that decides when a solar job actually finishes, and it's exactly the sort of thing that never makes it onto a checklist.

If something's wrong with the template

You'll be told which line, and **nothing will be applied**. A half-built project is worse than none, so a template with a problem in it is refused whole rather than partly created. Fix the line, apply again.

The fastest way of all: save a finished project as a template

You don't have to write a template from scratch. **The best template is a project you've already run.**

When a project has worked — the lists are right, the order is right, the durations turned out about right — go to the **three-dot menu** ⋮ → **Board** → 💾 **Save board as template**.

It reads the board and writes the template text for you: lists, colours, tasks, notes, durations, effort and dependencies, all in the format above. The template editor opens with it filled in, so you can read it over, tidy anything up and give it a name before saving.

It writes the nesting too, as `[nest under:]` lines, so a board you've tidied comes back tidy.

It deliberately leaves out the real dates, who was assigned, materials, time entries, stars and archived tasks. A template is the *shape* of a job, not a copy of one.

One thing it can't carry: **a child job nested into the board doesn't come across**, because a template builds lists and tasks, not jobs. The save tells you when that's happened, so you know what you're getting.

This is the route most people should take. Run your next solar install or bathroom reno properly, fix things as you go, and when it's done save the whole thing as a template. Every job like it after that starts already built and already planned.

Two examples built in

Under **three-dot menu** ⋮ → **Manage templates** you'll find two ready-made templates:

- **Get Started** — a five-list board (In Progress, Up Next, Maybe, On Hold, Done) with tasks explaining what each list is for. A good shape for state-based work.
- **Solar Install (example)** — a staged project with dependencies, durations and effort already filled in. Worth opening even if you don't do solar, because it shows what a planned template looks like.

Deleted them? **Restore built-in templates** puts back whichever are missing.

What next

- **Move tasks between lists** by dragging, on the Board.
- **Switch to the Timeline** to see the shape of a planned project.
- **Assign people** so everyone knows what's theirs.

Working the plan when it changes

Materials didn't land. The plan you built on Monday is wrong by Wednesday. Here's what Project Checklist does about it, and what it deliberately doesn't.

Every guide to planning a project stops at the point where you've made the plan.

That's not where the work is. The work is Wednesday, when the materials didn't land, the plan you built on Monday is wrong, and you've got to work out what that actually costs you.

The plan was always going to change

Worth saying plainly, because a lot of software implies otherwise: a plan that never changes isn't a good plan, it's a plan nobody's using.

The point of planning a project isn't to predict the future. It's so that when something moves, you can see what else moves with it — before your customer finds out, and before three subbies turn up to a site that isn't ready.

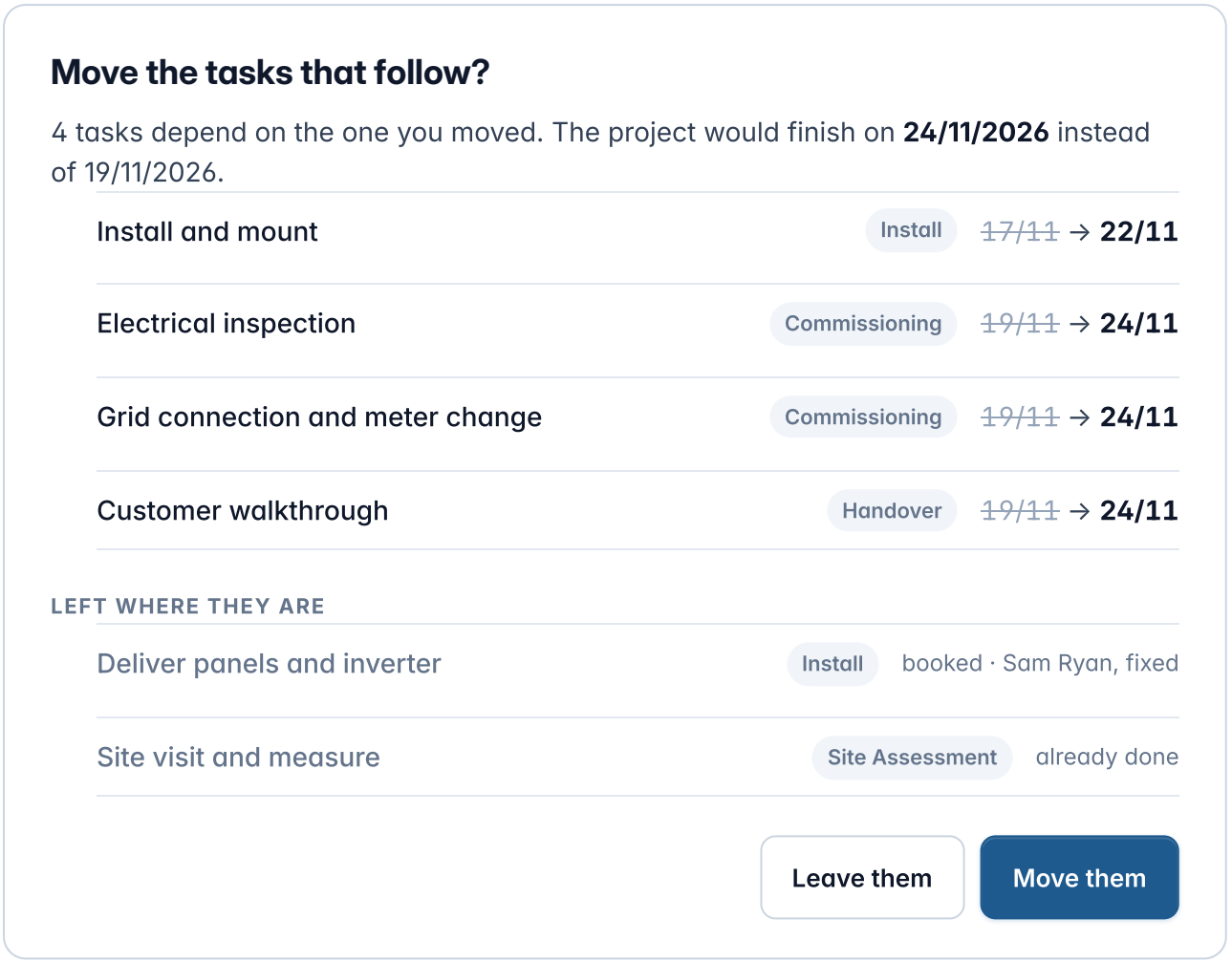
Move one thing, and see what follows

Drag a task to a new date on the Timeline, or change its dates in the task panel.

If other tasks were waiting on that one, you'll get asked:

> **Move the tasks that follow?**
>
> *4 tasks depend on the one you moved. The project would finish on 24/11/2026 instead of 19/11/2026.*

You'll see each task that would move, which list it's in, and its old and new dates. Then two buttons: **Move them**, or **Leave them**.



What you see after moving one task. Every task that would move, with its list and both dates. Under it, anything that stays put and why. Nothing changes until you press a button.

Nothing happens until you choose. The app works out the knock-on effects and shows you. It doesn't quietly rewrite your project while you're looking somewhere else.

That line about the finish date is usually the one that matters. Five days is the difference between a conversation with your customer today and an argument with them in three weeks.

Some things don't move

Under the moves you may see a short list headed **Left where they are**, with a reason next to each:

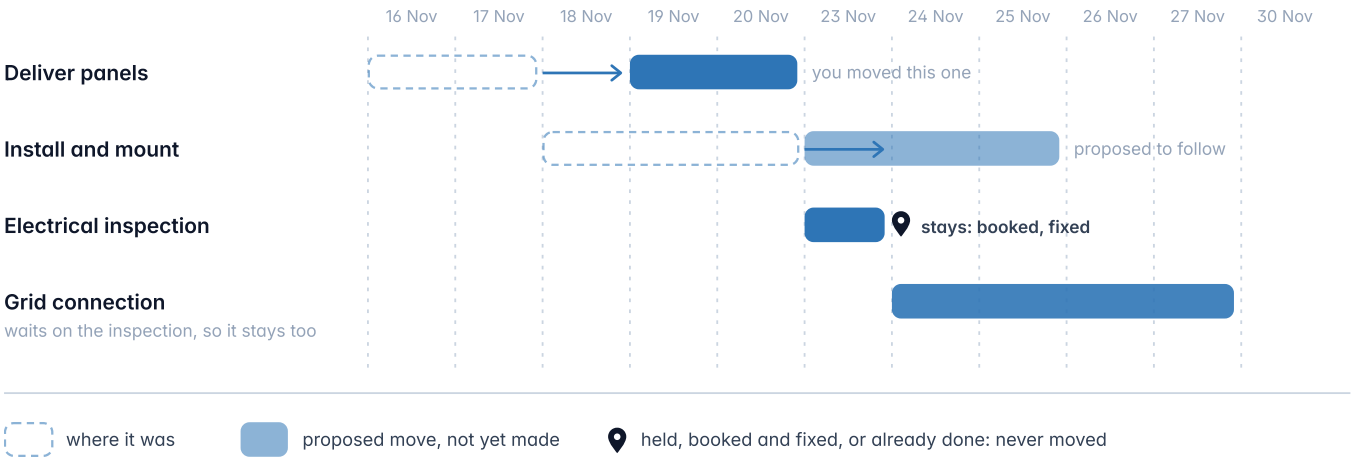
- **booked** — someone's booked to be there and you've marked that booking as fixed. More on this below, because bookings are the one thing here that isn't a simple yes or

no.

- **held** — you've told it not to move this one. More on that below.
- **already done** — the work's finished. Moving its dates would just be rewriting history.

When a task doesn't move, nothing after it moves either. That's deliberate — the app won't shuffle work past a fixed point and hand you a plan that can't happen.

Those tasks are the ones worth looking at. A booked task in that list means someone's booked for work that isn't ready yet, and you've just found out early enough to do something about it.



The ripple stops at the first task that can't move. Nothing behind a fixed point is shuffled past it, so the plan you're shown is one that can actually happen.

Holding a date

Some dates aren't yours to move. The crane's hired. The inspector's coming Thursday. The customer's been told.

Open the task and tick **Hold these dates — never move this task automatically**. It gets a small marker on the Timeline, and from then on it sits in **Left where they are** instead of being rescheduled.

Use it for real-world commitments, not for everything. Hold too much and nothing can move, which means the app can't help you see what a delay actually costs.

Bookings, and the one decision worth making early

A task with someone booked on it is the interesting case, because two things are true at once. The booking is a real commitment on a real person's calendar. And if it can never move, one booked task in the middle of a project freezes everything behind it.

So you decide, per booking. Open the task's booking and you'll see:

> **Fixed — never move this booking automatically**

Leave it unticked and the booking can be moved along with its task. Tick it and the task drops into **Left where they are** instead.

Movable is the default, including for bookings you made before this existed. That's safe because nothing happens on its own — every proposal is shown to you first, and any booking it intends to move is named:

> *Booked: Sam Ryan · Install crew · moves to 24/11/2026*

So you see who's affected and where they'd end up, before anything changes.

Tick Fixed ☑ when the date is somebody else's to set. The inspector's coming Thursday. The crane's hired. The customer's taken the day off work. Those shouldn't shuffle because something upstream slipped.

Leave it unticked for your own crew's work, where moving the booking is exactly what you'd do anyway.

It checks whether the person is actually free

When a proposal moves a booking, the app looks at where that person would land — their leave, their working days, anything already booked — and tells you if there's a problem.

A clash never stops you. It's information, not a rule. Often it's a half-hour job you're happy to book over, and you're the one who knows that. It just makes sure you're not finding out on the day.

What it deliberately doesn't do

Worth being straight about, because it's the difference between a tool you can trust and one you have to check.

It won't move a booking without telling you. Bookings can move — that's usually what you want, or one booked task freezes the rest of the project. But every one it intends to move is named in the panel, with the person and the new date, and it checks whether they're free. Nobody's Thursday changes without appearing on your screen first. And any booking you've marked **Fixed** is never touched at all.

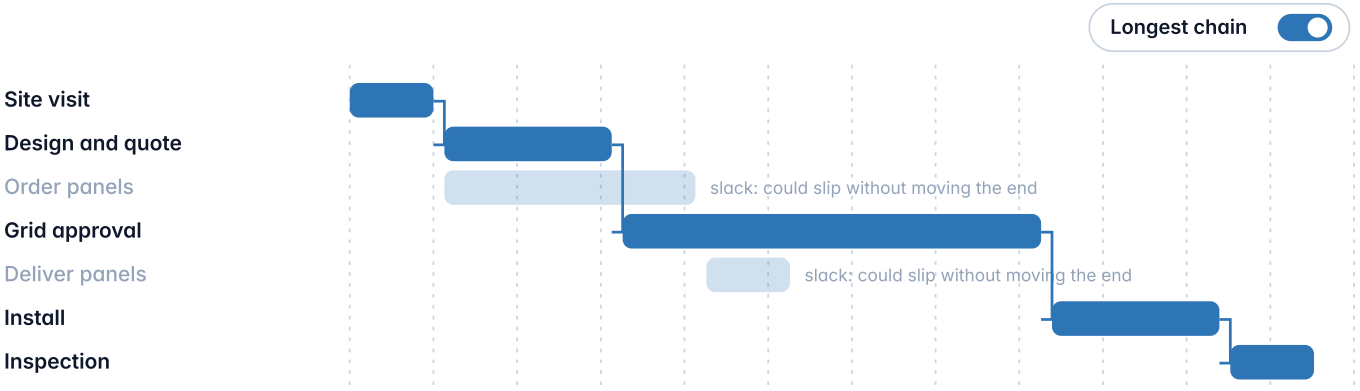
It only pulls work forward by what was gained. If something finishes early, the tasks waiting on it are offered the same number of working days earlier — not squeezed up to the earliest possible date. A gap you left on purpose, for a delivery lead time or curing, stays a gap. It's still a proposal you say yes or no to, it never moves a held task or a fixed booking, and it never moves anything into the past. Whether the people, materials and access are ready that early is still your call.

It won't decide anything. Every time, it shows you and asks. If you say **Leave them**, nothing moves and you'll be asked again next time something changes.

Seeing what's actually driving the date

On the Timeline there's a **Longest chain** Longest chain toggle.

Turn it on and it highlights the run of tasks that decides when the project finishes. If you want to pull the end date in, that's the chain you have to shorten — anything not on it has slack, and speeding it up won't finish the job any sooner.



Shorten a lit task and the end date comes in. Shorten a faded one and nothing changes.

The Longest chain toggle on the Timeline lights the tasks that decide when the project finishes. Usually there is one long wait on that chain that nobody had looked at.

It's also the fastest way to answer "why is this taking so long?" Usually there's one long wait on that chain that nobody had really looked at.

In practice

The routine is short:

1. **Plan the sequence** on the Timeline when the project starts.
2. **Work the day-to-day** on the Board.
3. **When something slips**, change its date and look at what's proposed.
4. **Check what didn't move** — that list is where your problems are.
5. **Tell the customer** while it's still early news.

That's it. The plan isn't a promise you made in week one. It's the thing that tells you what a delay costs, on the day it happens.