
A USEFUL PLACE TO START

Product scope & handover checklist.

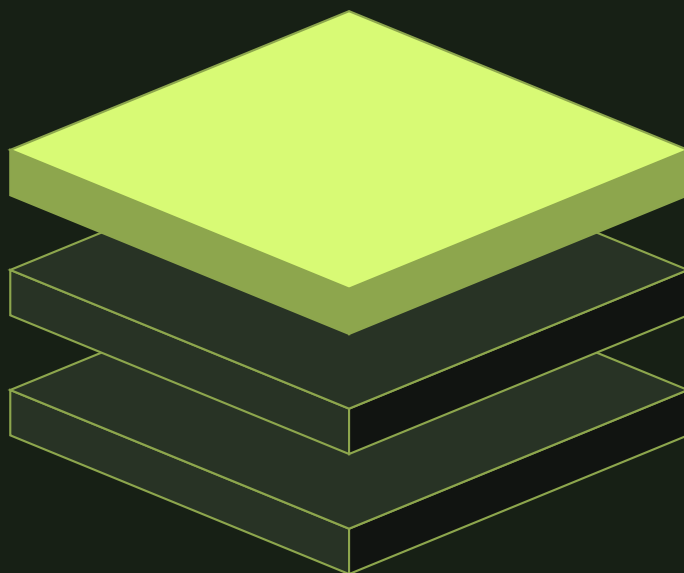
Six decisions for building, improving or connecting software.

DEFINE.

CHECK.

HAND OVER.

SIX DECISIONS TO MAKE
BEFORE YOU COMMIT.



Define a useful first scope.	02
Choose tools. Check behavior.	03
Plan ownership and support.	04
Worked example / client intake workflow	05

Define a useful first scope.

Use this guide for a new product, an existing application or a repeated workflow. Start with a useful change, then compare proposals against the same scope.

01 Start with a useful change

Describe who needs help and what they need to do. Choose a result you can observe before adding a feature list.

- Name the user and the task they need to complete.
- Show how the work happens today, including a bottleneck.
- Describe what a successful journey will look like.

IN PRACTICE

An operations coordinator approves a request once. It then appears in the team tracker without retyping the details.

02 Choose the first commitment

A clear brief may be ready for a build. An assessment resolves unknowns; a prototype tests a journey. Choose the step that answers your next decision.

- Agree what is included and what waits for later.
- Ask for concrete deliverables and review criteria.
- Distinguish a testable prototype from a production release.

IN PRACTICE

The first release lets customers request a quote and staff review it. Payments and historical migration are separate decisions.

DEFINE

CHECK

OWN

Choose tools. Check behavior.

The right technology fits the job and the systems you already use. Ask to see a working journey, including how it handles missing information and failed actions.

03 Fit the tools to the work

Start with your existing product, data and integrations. Compare options on fit, operating costs and maintainability rather than the length of a technology list.

- Keep working systems where they meet the requirements.
- Confirm required connections, access and permitted data.
- Check provider limits, ongoing costs and export options.

IN PRACTICE

Keep the existing CRM and connect one intake workflow. Confirm API access, usage limits and data export before selecting new tools.

04 Review behavior, including AI

Use examples that match daily work, including an important failure case. Add AI only when it helps the task and its output can be checked.

- Agree success and failure examples before building.
- For AI, check answer quality, allowed data and human review.
- Confirm how failed actions are reported and recovered.

IN PRACTICE

A document assistant is checked against known questions, cites approved sources and asks for human help when it cannot answer reliably.

DEFINE

CHECK

OWN

Plan ownership and support.

Compare the cost of building with the cost of running the result. Make ownership, access and support responsibilities clear before committing to the work.

05 Know what you receive

Ask for the handover items in the proposal. Confirm ownership and licensing terms, business account access and what another engineer needs to maintain the software.

- Agree code ownership, licenses and repository access.
- Confirm who controls hosting and connected service accounts.
- Include setup, deployment and operating instructions.

IN PRACTICE

The handover includes repository access, setup instructions and an account inventory. Ownership and third-party licenses are stated in the agreement.

06 Budget for life after launch

Separate the build fee from provider costs and ongoing support. A usable release needs someone responsible for operations and future changes.

- Identify hosting, API usage, licenses and support costs.
- Name who handles incidents, backups and maintenance.
- Agree support coverage and how later changes are priced.

IN PRACTICE

The proposal separates implementation, tracker subscription and support. A named owner handles operations after launch.

EXAMPLE / CLIENT INTAKE

One workflow. A clear boundary.

Illustrative project brief: a small team wants to stop copying requests between an inbox and a tracker. This example shows the detail needed to discuss a useful first scope.

01 / GOAL & FIRST SCOPE

An operations coordinator reviews a request and approves a tracker record. Include one request form and one tracker integration. Keep approval with the coordinator; billing and historical migration wait for later.

02 / TECHNOLOGY & DATA

Use the existing tracker if its API supports the workflow. Confirm access, permitted request data and provider limits. AI is not needed for this first scope; structured fields and a direct integration are enough.

03 / REVIEW & EXCEPTIONS

Try a valid request, missing details and a tracker outage. Keep failed requests visible for recovery and avoid duplicate records. The coordinator reviews these examples before accepting the release.

04 / COST & HANDOVER

Separate the build fee, tracker subscription and support. Agree code and account access, setup notes and the operating owner. Use the same brief to compare proposals. You can bring an incomplete brief to the first conversation.

Discuss your first scope