

Handling Client Feedback & Revisions — Workbook

This workbook turns the course into a working system you can apply to a live project. Pick one current or upcoming client engagement and use these exercises, worksheets, and checklists alongside the lessons. The included templates give you a reusable change log, a revision-rounds tracker, and a scope-and-rates sheet you can copy for every future project.

Why Feedback Goes Wrong (and What It Really Costs)

Make the hidden costs of unbounded revisions visible and lock your feedback rules into the proposal before work begins.

Exercise: Audit a Stuck Project

Choose one past or current project that ran long on revisions. Work through the prompts honestly with real numbers, not estimates from memory where you can avoid it.

- What was the flat fee or budget, and how many hours did you originally estimate? Calculate your intended hourly rate.

- How many hours did the project actually take once revisions were included? Recalculate your real effective hourly rate.

- Of the four hidden costs (time leakage, context-switching, opportunity cost, emotional cost), which hurt you most on this project and why?

- What single missing agreement at the start would have prevented the most rework?

Worksheet: Feedback Rules for Your Proposal

Draft the feedback and revision terms you will put into your next proposal and read aloud at kickoff. Fill every field; vague fields are where scope creep enters.

Number of revision rounds included in the package

Definition of one round (what counts, where feedback goes, the time window)

The single channel all feedback must come through

Your turnaround promise (how fast you respond)

Client turnaround expectation (how fast they consolidate feedback)

Rate for extra rounds or out-of-scope work (per hour or per round)

Checklist: Expectation-Setting Checklist

- Revision round count is stated in writing in the proposal
- A round is explicitly defined, including the single feedback channel
- Both my turnaround and the client's consolidation window are stated
- The rate for extra rounds and out-of-scope work is written down
- I read the revision terms aloud during the kickoff call
- The client verbally acknowledged the revision terms

Collecting and Decoding Feedback

Set up one structured intake channel and practise turning vague reactions into confirmed, prioritised change requests.

Worksheet: Design Your Feedback Intake

Define exactly how feedback will arrive on your next project so notes are specific, in-context, and consolidated by default.

Deliverable type (design, video, copy, document, other)

Chosen in-context review tool (e.g. Figma, Frame.io, Google Docs, Markup.io)

The single review link or location you will send

Feedback deadline you will request (date and time)

Your three milestone-review prompts (what is working, what is not and what was expected, what is missing)

Your standard reply for redirecting stray email or text notes back to the channel

Exercise: Decode Three Vague Notes

Take three real or realistic vague notes (for example make it pop, it feels cheap, I just do not love it) and run each through the four-step decode framework, writing out what you would actually say.

- Write your acknowledge line that does not defend the work.

- Write your locate question to find which specific part is the problem.

- Write your translate question that converts the adjective into an outcome (more premium, more modern, more trustworthy).

- Write your confirm sentence that reflects the concrete change back and asks for a yes.

Exercise: Prioritise a Flood of Notes

Imagine a client returns 15 to 20 notes at once. Sort a real or sample set using MoSCoW and turn it into a round you can actually close.

- List each note and tag it Must, Should, Could, or Will not.

- Identify any two notes that contradict each other and write the question you would ask to make the client choose.

- Group related notes so you make each kind of change only once.

- Draft the message that sends the prioritised list back and gets sign-off on the round's scope before you start.

Running Bounded Revision Rounds

Run revisions as finite, batched rounds with explicit approval gates and a light tool stack that keeps everything traceable.

Worksheet: Plan a Single Revision Round

Before starting your next round, complete this plan so the round has a clear start, end, and deliverable, and is closed to new additions once confirmed.

Round number and version label you will deliver (e.g. v2)

List of confirmed changes in this round

Items explicitly parked for the next round (the parking lot)

Your confirm-the-scope message before you begin work

Your change-summary message that will accompany the returned version

Date you will return the round to the client

Exercise: Write Your Approval-Gate Scripts

Draft the exact wording you will use to request approval and to handle a reopened decision, so you never have to improvise these high-stakes moments.

- Write your concept-gate approval request (approve the direction before detailed production).
- Write your round-gate request that asks for the word approved or triggers the next round.
- Write your final sign-off request before releasing assets.
- Write your calm reference script for when a client reopens something already signed off.

Checklist: Revision Tracking Setup Checklist

- [] One in-context review tool chosen and the review link ready to send
- [] A change log exists with one row per change request
- [] Each row records the client note, my decoded version, MoSCoW priority, status, and round
- [] A clear file-naming and version convention is in place (e.g. project-v1, project-v2 with date)
- [] Written approvals are stored and linked from the change log
- [] Mid-round ideas have a parking lot so the current round stays focused

Scope, Conflict, and Closing Well

Protect scope, defuse hot feedback, and close projects with sign-off plus a retrospective that improves your next brief.

Worksheet: Run the Scope Test

Apply the scope test to a real incoming request to decide whether it is an in-scope revision or new work, then plan how you will price or fold it in.

The exact change the client requested

Was it in the original brief or deliverables list? (yes / no)

Does it change quantity, medium, or format, or reverse an approved decision? (describe)

Verdict: in-scope revision or new scope

If new scope, your estimate (hours and rate) and your friendly framing message

Decision: fold into this round, quote separately, or absorb as a stated gift

Exercise: Build Your De-escalation Toolkit

Prepare reusable scripts for hard feedback moments so you can stay calm and reference them under pressure rather than reacting in the heat of the moment.

- Write your acknowledge-the-emotion line for harsh or disappointed feedback.
 - Write your script for handling two contradictory notes by asking the client to choose.
 - Write your script for locking a direction when the client has become a moving target.
 - Note your personal pause rule (how long you will wait before replying to hot feedback).
-

Worksheet: Project Closing Retrospective

After delivery, complete this short self-retrospective so each project sharpens your next brief and shrinks future revision counts.

Rounds quoted versus rounds actually used

The single missing agreement that caused the most rework

The one sentence you will add to your brief, proposal, or kickoff to prevent it next time

One thing that went well that you will deliberately repeat

Confirmation that final sign-off, organised files, and the final change log were delivered

Checklist: Clean Handover Checklist

- [] Final sign-off confirmed in writing before assets were released
- [] Final files delivered, organised, with a short summary of what was produced
- [] Final change log shared so the client sees how feedback was honoured
- [] Post-delivery terms stated (what minor fixes are included and what counts as new work)
- [] Five-minute retrospective completed and one brief or kickoff improvement captured

Your Action Plan

1. Pick one live or upcoming client project to apply this system to this week.
2. Add written revision rules (round count, round definition, single channel, rates) to your proposal and kickoff.
3. Choose one in-context review tool and prepare the single review link you will send for feedback.
4. Stand up the change-log template with one row per change request, including the decode and MoSCoW columns.
5. Use the four-step decode framework on every vague note before doing any work, and confirm each change in writing.
6. Consolidate and prioritise all feedback with MoSCoW, then send the list back for sign-off before starting a round.
7. Run each batch of changes as a bounded round with a version label, and park mid-round ideas for the next round.
8. Request explicit approval at each gate and store the written yes, linked from the change log.
9. Apply the scope test to every request and estimate or defer anything that is new scope, warmly and early.
10. At delivery, confirm sign-off, hand over cleanly, and run the five-minute retrospective to improve your next brief.

