

Style Guide Design — Workbook

This workbook turns the course into a real, living style guide for one product, publication or site. You will scope the right type and tier, document a component with full anatomy, variants, states and tokens, write editorial image guidelines and a tone-of-voice document with we are and we are not pairs, assemble the guide in a living tool, and put a governance, versioning and maintenance process in place. Work one section per module and finish with a guide a team could use without you, plus a system that keeps it accurate.

What a Style Guide Is For

Decide which kind of style guide you need, at what tier, and name the single source of truth before documenting anything.

Exercise: Type and Tier Decision Drill

Take three different briefs (for example a one-writer newsletter, a two-designer SaaS product shipping weekly, and a multi-team retail website) and decide the right style-guide type and tier for each, justifying every call by who uses it and across how many surfaces.

- For each brief, name the dominant need: content and editorial, UI component library, image guide, voice document, or a combined guide weighted toward one or two.

- Count how many people will use the guide without help, across how many surfaces, and how often those surfaces change.

- Pick a tier for each: lean reference, working guide, or system-backed guide, and say why.

- Write a one-line scope statement for each brief that you would get the team to agree before building.

Worksheet: Project Scope and Source-of-Truth Sheet

Lock the scope for the one guide you will build in this workbook, and name its single canonical home, so every later entry traces back to a real type, tier and place.

Product / publication / site name

Dominant guide type (content / components / image / voice / combined)

Number of internal users (no help)

Number of external users (partners, freelancers, contractors)

Surfaces it must cover (product UI / marketing site / docs / social / print)

How often those surfaces change (rarely / monthly / weekly)

Tier chosen (lean / working / system-backed)

Single source of truth (tool + one short address)

What this guide does NOT cover (and where that lives instead)

One-line scope statement the team can agree

Worksheet: Section Plan and Navigation

Plan the top-level structure of your guide so it runs from foundational to specific and stays findable. Tick the sections your type and tier need.

Foundations (spacing scale, tokens) — include? (Y/N)

Components and patterns — include? (Y/N)

Content style and terminology — include? (Y/N)

Tone of voice — include? (Y/N)

Editorial imagery — include? (Y/N)

Layout / grid patterns — include? (Y/N)

Governance and changelog — include? (Y/N)

Top-level order (list the sections in reading order)

Checklist: Pre-Build Readiness

- I have named the dominant type of guide and not defaulted to a component library
- I have counted users and surfaces and picked a tier that matches them, not my taste
- I have chosen one canonical home and a single short address for it
- I have written down what the guide does NOT cover and linked where that lives
- I have a top-level structure ordered from foundational to specific
- I have a one-line scope statement the team can agree before I build

Documenting Components and Patterns

Turn one real component into a full reference entry with a repeatable anatomy, variants, states, tokens and do and do not rules.

Worksheet: Component Entry Spec

Document one real component end to end using the standard entry template, so it becomes a model for every other entry in your guide. Fill the values for your chosen component.

Component name

One-line description (what it is, plain words)

When to use (the decision the reader came for)

When NOT to use (the alternative to reach for instead)

Anatomy — list each labelled part (e.g. label, icon, container, helper text)

Variants (e.g. primary / secondary / tertiary / destructive) and each one's job

States covered (default / hover / focus / active / disabled / loading / error)

Spacing tokens (padding, internal gap, margin — named, not raw px)

Content guidance (how to write the label or copy inside it)

Exercise: Spacing Token Scale Drill

Build the spacing scale your components will reference, so every entry uses named tokens instead of loose pixels.

- Pick one base unit (4px or 8px) and write out the fixed scale (e.g. 4, 8, 12, 16, 24, 32, 40, 48).
 - Name each step (xs, sm, md, lg, xl, or a numeric token scale) and commit to using only those names.
 - Re-describe your component's padding and gaps using only token names, with no raw pixel values.
 - State the minimum touch-target size for interactive elements (commonly 44 by 44 px) and check your component meets it.
-

Exercise: Do and Do Not Page Build

Create the most-read part of the entry by applying real misuses to your actual component and marking each one, so the rules are concrete.

- List the real mistakes you have actually seen with this component, not hypothetical ones.
- For the top five or six, build a correct and an incorrect example at the same size, side by side.
- Mark each wrong example with a red cross and each right one with a green tick, ordered most-common first.

-
- Write a one-line, neutral caption for each that names the action and the rule, not the person.
-

Checklist: Component Entry Readiness

-] The entry follows the same template I will use for every component, with no missing sections
-] It says clearly when to use AND when not to use the component
-] An anatomy diagram labels every part of the component
-] Every legitimate variant is shown, with a clear job for each, and the set is capped
-] All interaction states are shown, including a visible focus state for keyboard users
-] Spacing is specified in named tokens from a fixed scale, not raw pixels
-] The do and do not section shows real misuses crossed out with neutral, specific captions

Editorial Images and Tone of Voice

Direct imagery concretely and write a tone-of-voice document that makes everyone sound like one author.

Exercise: Image Direction Drill

Direct your editorial imagery concretely, with approved and rejected examples, so photos and illustrations stay one coherent set.

- Describe the photography in concrete terms: subject, mood, lighting, colour treatment and composition.
-
- Gather three approved images and three rejected ones, each captioned with a one-line reason.
-
- List the exact aspect ratios you use (e.g. 16:9 hero, 4:5 in-article, 1:1 social) and tie each to a real placement.
-
- Write the alt-text rule (every meaningful image gets concise descriptive alt text; decorative images get empty alt) and the licensing rule (approved sources only, how to record credit).
-

Worksheet: Tone-of-Voice Sheet

Document how the brand sounds so anyone writing for it reads as one author. Make every trait concrete with a contrast pair and a rewritten example.

We are / we are not (trait pair 1)

We are / we are not (trait pair 2)

We are / we are not (trait pair 3)

Voice principles (3 to 5 short, actionable statements)

Tone in onboarding (one line describing the shift)

Tone in error messages (one line)

Tone in marketing (one line)

Tone in support (one line)

Before and after example (same message written wrong, then right)

Worksheet: Content Style and Terminology Sheet

Settle the small mechanical rules once, and build the agreed word list, so the product does not read as if many strangers wrote it.

Base style adopted (e.g. a recognised editorial style guide) and key deviations

Heading / button case (sentence case or title case)

Sign in vs log in, and standard verbs for standard actions

Numbers rule (when spelled out vs numerals; how ranges are written)

Date and time format (e.g. 7 June 2026; 12-hour or 24-hour)

Punctuation (Oxford comma Y/N; contractions allowed Y/N; & or and)

Terminology — preferred term vs avoid (list 5+ pairs)

Feature naming convention (how you name your own features)

Checklist: Editorial and Voice Readiness

Image direction is concrete (subject, lighting, treatment, composition) with approved and rejected examples

Aspect ratios are listed and tied to real placements

Alt-text and licensing rules are written and unambiguous

Voice is defined with we are / we are not pairs, not single vague adjectives

Tone is shown shifting across onboarding, errors, marketing and support

At least one before and after rewrite demonstrates the voice in practice

A terminology list names every key concept and the words to avoid

Building and Maintaining the Guide

Assemble the guide in a living tool, then put governance, versioning and a maintenance routine in place so it stays accurate.

Worksheet: Assembly and Live-Asset Brief

Capture how you will build the guide and connect it to real assets so it cannot silently disagree with the product. Fill the choices for your project.

Platform (Notion / Confluence / Zeroheight / Figma / Storybook / docs site)

Top-level navigation (sections in reading order, foundational to specific)

Live assets linked (Figma styles / Storybook stories / icon set / fonts)? (Y/N each)

Every component uses the same entry template? (Y/N)

Search or a clear index present? (Y/N)

Version number and last-updated date visible to readers? (Y/N)

One short, shared address announced to the team? (Y/N)

Worksheet: Governance and Versioning Plan

Decide who owns the guide and how it changes, so it stays trustworthy rather than drifting. Fill the choices for your team.

Owner (person or small group with final say)

Governance model (centralised or federated) and why

How a change is proposed (issue / form / request channel)

Review step (who checks, and against what: fit, consistency, specificity)

How versions are tracked (version number + changelog location)

How meaningful changes are announced (where the team will see them)

Whole-guide review cadence (e.g. quarterly)

Exercise: Maintenance Routine and Audit Drill

Design the small, regular habits that keep the guide current, and run a first audit against reality so you start clean.

- Write the rule that a change is not done until its guide entry is updated, and where that rule lives.
 - Open the guide and audit five entries against the real product or content; note every mismatch.
 - Scan recent team questions for drift signals: which answers are missing or unclear in the guide?
 - List any deprecated patterns to mark clearly (not silently delete) and any dead rules to prune.
-

Checklist: Launch and Upkeep Readiness

- The guide lives in a living tool, not a frozen PDF, at one shared address
- It links to real, live assets so nobody screenshots stale components
- Navigation is stable and every component uses the same entry template
- A version number and last-updated date are visible to every reader
- The guide has a named owner and one clear way to propose changes
- Versions are tracked with a changelog and meaningful changes are announced
- A maintenance routine and review cadence are set, and I have run a first audit

Your Action Plan

1. Name the dominant guide type and tier, pick a single source of truth, and write a one-line scope statement the team agrees
2. Build a stable top-level structure ordered foundational to specific, and decide what the guide will NOT cover
3. Define a spacing token scale from one base unit and commit to using only token names in entries
4. Document one component fully with the standard template: when to use and not, anatomy, variants, states, tokens, content guidance
5. Build the do and do not section by crossing out real misuses applied to your actual component, captioned neutrally
6. Write editorial image guidelines with approved and rejected examples, aspect ratios, alt-text and licensing rules
7. Write a tone-of-voice document with we are / we are not pairs, tone shifts, and before and after rewrites
8. Settle content style and build a terminology list of preferred terms and words to avoid
9. Assemble the guide in a living tool, link real assets, add search, and show a version number and date
10. Set up governance, versioning and a maintenance routine, then run a first audit against reality

