

FIELD NOTES • 28 JUL 2026

Describe it. Claude *builds* it.

I've spent the last few months building with Claude Design at work – decks, prototypes, campaign pages. Here's what I've learned: where it earns its place, what I actually ship with it, the handful of habits that get me a usable draft on the first try, and the edges I hit before I trusted it with anything that mattered.

01 – WHEN TO REACH FOR IT

WHERE IT FITS

The setup is plain: a chat box on the left, a live canvas on the right. I describe a prototype, deck, or page and Claude builds it as **real, working code** – not a picture of a design. Then I refine by conversation, inline comment, or direct edit. The thing that took me longest to internalise: this isn't a mockup tool, it's a drafting tool that happens to output the real artifact. [Claude Design ↗](#)

REACH FOR IT

- I need an idea visible before anyone can react to it.
- I want an interactive prototype to user-test – no PRs, no code review.
- I need an on-brand deck, one-pager, or landing page, fast.
- I want a dozen directions in minutes, then to narrow down.

Best for · concept to a working first version.

STAY IN YOUR STACK

- Mature component workflows and design ops at scale.
- Real-time, multiplayer editing with live cursors.
- Production-grade polish and precise pixel control.
- A living design file the whole team co-edits and versions.

Best for · established tools still own this ground.

HOW I DECIDE

If the hard part is getting the idea visible, I start in Claude Design. If the hard part is production collaboration and pixel-level polish, I stay in my existing tools – and let Design hand off to them.

02 – WHAT TEAMS BUILD WITH IT

USE CASES

Anthropic points it at prototypes, decks, microsites, and engineering handoff. [Admin guide ↗](#) In practice I've found the range wider – it comes down to roughly four jobs, whichever team you sit on.

SALES & PRE-SALES

The pitch, before engineering builds anything

Branded decks and per-vertical variants, proposal and RFP packs on your template, case-study one-pagers to PDF, battlecards, and clickable demo mockups that show intent before a line of product code is written.

PRODUCT & UX

The idea, made testable

UI concepts and dashboards, interactive click-through prototypes for user testing, wireframe storyboards to explore layouts cheaply, a reusable UI kit, and a spec-plus-assets handoff to the dev team.

MARKETING

The campaign, one visual system

Landing pages and microsites, HTML emails and nurture sequences, sized social and ad sets, infographics and report visuals, lead-magnet assets (checklists, ROI worksheets), and event collateral.

DELIVERY & INTERNAL

The work, made repeatable

Branded status reports each sprint, architecture and process docs, workshop material, onboarding guides, and all-hands or board decks – the “ugly but necessary” artifacts that rarely justify a designer.

Almost none of my leverage came from clever prompting. It came from setup. These are the six habits I'd hand my past self.

NO. 01 **Set up your design system before anything else.** During onboarding Claude reads your codebase and design files and builds the system – then every later project uses your colours, type, and components on its own. Do it once and you skip the generic, Claude-shaped output I got every single time I skipped it. [Setup guide ↗](#)

NO. 02 **Start from something real, and start simple.** I get a cleaner result asking for the core layout and content first, then layering in interactions, edge cases, and polish – not dumping the whole spec into the opening prompt. [Get started ↗](#)

NO. 03 **Import; don't describe from scratch.** Feeding it a DOCX, PPTX, or XLSX, pointing it at a codebase, or web-capturing elements off my live site gets me far closer to the real thing than any prompt I could type. [Anthropic ↗](#)

NO. 04 **Chat for big moves, inline comments for small ones.** Broad shifts (“make it darker and more minimal”) go in chat; I click the exact element for a targeted tweak. When a comment doesn't land – and it happens – I paste it straight into chat, the documented workaround. [Get started ↗](#)

NO. 05 **Build templates you can reuse.** Regenerating a layout from zero every time is slower and eats into your usage – I keep on-brand starters and work from those.

NO. 06 **Hand off the moment it's ready.** Export to PPTX, PDF, or HTML; send it to Canva, Gamma, Adobe, Miro, Vercel; or hand to Claude Code, which picks up from your work instead of starting over from a screenshot. [Admin guide ↗](#)

It's good, and it's early – I've hit the seams. Three things I wish I'd known before I leaned on it for something that had to ship.

It's a beta, on paid plans only

claude.ai/design

Claude Design is in beta and included with Pro, Max, Team, and Enterprise – not Free. On Enterprise it's off by default; an admin turns it on under Organization settings. [Claude Design](#) ↗

Not a full design tool yet

[Handoff](#) > [source of truth](#)

Sharing is organization-scoped – private, view, or edit – but there's no real-time multiplayer with live cursors yet. [Anthropic](#) ↗ The export menu covers PPTX, PDF, HTML, and partners like Canva, Gamma, and Adobe; Figma isn't in that list, but the official [Figma connector](#) ↗ sends your screens into Figma as editable layers mapped to your design-system components.

How I use it: as the place ideas become visible and get handed off – not the file my team lives in and edits together all day.

Usage draws on your plan

Front-load · reuse · scope

Design work counts against your subscription's usage, the same pool as chat and coding. Setting up the design system once, reusing templates, and pointing Claude at a specific folder rather than a whole large repo all keep that spend in check.

What I've seen (not in the docs): large repos can slow the canvas or stall the chat – pointing Claude at one subdirectory instead of the whole monorepo fixes it for me. Worth checking against your own setup.

For the curious, the
bold, the *slightly*
ahead.