

Figma Handoff Checklist

A handoff checklist for design teams - files, naming, tokens, exports, specs, links, and what clients/devs need to move fast.

Get the editable version

<https://corcava.com/templates/client-ops/figma-handoff-checklist>

This PDF is designed to be printed or used as a QA checklist. Publish the editable Doc version on the page above.



What's inside

- A complete handoff checklist (printable)
- A 'handoff package' table to organize files and links
- A quick QA pass before you send to devs/clients

How to use this checklist

- Before handoff, do a 10-15 minute cleanup pass: naming, structure, and removing dead pages.
- Run the checklist section-by-section and capture open questions in your handoff package notes.
- Send one source-of-truth link (the Figma file) and store supporting assets (exports, copy, specs) in a single handoff folder.

Tip: Share this checklist with your dev lead early. It prevents rework.

The checklist

Use this as a final pass before sending a handoff to engineering or a client. Check items that apply to your project.

1) File hygiene and structure

- Add a cover / overview page with: project name, owner, last updated date, and key links.
- Use a clear page structure (example): 00-Overview, 10-Components, 20-Screens, 30-Prototypes, 90-Archive.
- Rename pages, frames, and key layers so they are searchable and unambiguous (avoid 'Frame 123').
- Remove or move dead exploration work to an Archive page so the main file stays clean.
- Confirm any shared libraries (components, icons) are linked or included, not buried in a personal draft file.

2) Components and layout

- Use components for repeatable UI. Avoid detached instances unless there is a specific reason.
- Verify key component variants exist (states, sizes, error/disabled/loading, empty states).
- Use Auto Layout where appropriate so spacing rules are obvious to engineering.
- Set constraints / resizing behavior on key frames (what should stretch, what should stay fixed).
- Ensure components have consistent naming (e.g., Button / Primary / Default).

3) Tokens and styles

- Define and use text styles (type scale, weights, line heights).
- Define and use color styles / variables (prefer semantic names over raw hex).
- Document spacing rules (even a short note: 4/8 grid, spacing scale, etc.).
- Call out any brand assets or design tokens that live outside Figma (link to source-of-truth).
- Confirm icon style rules (stroke width, corner radius, filled vs outline).

4) Export specs and assets

- List what should be exported (icons, illustrations, logos) and in which formats (SVG/PNG).
- Confirm naming convention for exported assets (kebab-case or snake_case, no spaces).
- Confirm export sizes (@1x/@2x or specific pixel dimensions) and background transparency rules.
- Make sure assets are not duplicated across multiple frames with slight differences - pick one source.
- Note any platform-specific requirements (iOS PDF/SVG, Android vector, web SVG).

5) Specs, notes, and edge cases

- Add spec notes for anything non-obvious (spacing exceptions, component behavior, truncation rules).
- Include empty, error, loading, and permission states for core flows.
- Call out responsive breakpoints and what changes at each breakpoint.
- Document copy sources (final copy vs placeholder) and where to find the copy deck if separate.
- Link decisions: if something changed, note the reason and date in a short change log.

6) Prototype and flows

- Provide one prototype link as the primary demo (avoid multiple competing prototypes).
- Create clear starting points for major flows (login, onboarding, purchase, etc.).
- Add interaction notes for complex behavior (multi-step modals, conditional flows, animations).
- Ensure prototype hotspots match the latest UI (no stale screens).
- Include any analytics / tracking requirements as notes if they affect UI behavior.

7) Accessibility notes

- Check contrast for core text and buttons (and call out exceptions intentionally).
- Include focus states and keyboard navigation expectations for interactive elements.
- Note required aria labels / alt text for icons, images, and controls without visible labels.
- Include error message patterns (where the message appears, tone, and when it triggers).
- Confirm hit area size expectations for mobile / touch targets where relevant.

8) Where to store everything (client portal)

- Create one Handoff folder with a clear structure (see the handoff package table on the next page).
- Put the Figma link in the same place every time (top of folder or a README doc).
- Store exports and specs next to the Figma link - not scattered across Slack threads.
- Confirm permissions: client and devs should have access without requesting access from personal accounts.
- Add a short release note / handoff note: what shipped, what is pending, and who to contact.

Handoff package table

Use this table to keep every handoff artifact in one predictable place. Replace placeholders with your real links.

Item	Owner	Where it's stored	Link	Notes
Figma source-of-truth file	Design lead	Handoff/01-Figma	Figma URL	Include last updated date + version tag
Prototype (if separate)	Design lead	Handoff/01-Figma	Prototype URL	Single link, clear starting points
Exported assets	Designer	Handoff/02-Exports	Folder URL	SVG/PNG + naming convention
Specs / notes	Designer	Handoff/03-Specs	Doc URL	Edge cases, breakpoints, interactions
Copy deck	PM/Design	Handoff/04-Copy	Doc URL	Final vs placeholder callouts
Decision log / change log	PM/Design	Handoff/05-Decisions	Doc URL	Why changes happened + dates
Release note / handoff note	PM/Design	Handoff/00-README	Doc URL	What is done, what's pending, contacts

Handoff QA (10-minute final pass)

- Open the Figma file with viewer permissions (or incognito) and confirm everything is readable.
- Click every link in your handoff package and confirm it opens without extra permissions.
- Spot-check 3-5 key components: naming, variants, and constraints/responsiveness.
- Confirm exports open and match the latest UI (no stale icon set).
- Ask a developer to confirm: 'I know where to find breakpoints, edge cases, and final copy.'

Related templates: **Client Portal Requirements Checklist** • **Creative Brief Template** • **Scope of Work Template**