

Logo and Mark Treatment Guide

A step-by-step playbook for preparing partner-provided logo and mark assets for use across the Movemental app and properties. The goal is **not** to redesign the marks but to give them proper treatment: clean exports, color variants, background removal, and a full set of production-ready assets that follow best practices.

Goals and scope

What we're accomplishing

- 1. **Audit and normalize** — Assess the source files (logo.webp, mark.webp) for resolution, format, and quality; establish a single source of truth.
- 2. **Background removal** — Produce transparent-background (alpha) versions for overlay on any background (light, dark, colored, imagery).
- 3. **Color variants** — Generate full-color, single-color (e.g. black, white), and optional brand-color versions so the logo works on light and dark surfaces and in monochrome contexts.
- 4. **Mark and icon treatment** — Export the mark (symbol only) as standalone assets at defined sizes for favicons, app icons, social avatars, and UI icons.
- 5. **Full logo treatment** — Lock up logo (wordmark + mark) with clear spacing, alignment, and minimum size rules; export horizontal and stacked versions if applicable.
- 6. **Format and optimization** — Deliver the right formats (SVG where possible, PNG/WebP for rasters) at appropriate resolutions and file sizes for web and print.
- 7. **Organization and documentation** — Place assets in a consistent structure and document specs (sizes, formats, usage) so the team and the app can reference them correctly.

Final outcome: asset manifest

By the end of this process you will have produced (and optionally not use every variant—only what the playbook and your needs dictate):

Asset type	Formats	Key sizes / notes
Full logo (horizontal)	SVG, PNG, WebP	Transparent + white + black; min height ~24–32px for legibility
Full logo (stacked)	SVG, PNG, WebP	If applicable; same color variants
Mark only (symbol)	SVG, PNG, WebP	Transparent + white + black; multiple sizes for different contexts
Favicon	ICO, PNG	16×16, 32×32, 48×48 (or single 32×32 SVG favicon)
App / PWA icons	PNG	192×192, 512×512 (and any intermediate sizes your framework expects)
Social / Open Graph	PNG or WebP	1200×630 or 1200×600 for OG; square 1200×1200 or 1024×1024 for Twitter/social

Asset type	Formats	Key sizes / notes
Print-ready	PDF or EPS (vector) or high-res PNG	300 DPI equivalent for one-color and full-color; crop marks optional

Details to document for each:

- **Naming:** e.g. `movemental-logo-full-dark.svg`, `movemental-mark-white-32.svg`
- **Dimensions:** width × height in px (or pt for print)
- **Color mode:** full color, black, white, or brand hex
- **Background:** transparent or specified (e.g. dark #1a1a1a)
- **Clear space / minimum size:** e.g. “clear space = height of ‘M’ in wordmark”; “minimum height 24px on screen”

Prerequisites

- **Source files:**
 - `public/media-library/images/logo/logo.webp` — full logo (wordmark + mark)
 - `public/media-library/images/logo/mark.webp` — mark/symbol only(PNG versions also exist; use the highest-quality source available.)
- **Tools (use as advised below):** Adobe Suite (Illustrator, Photoshop), Figma, and optionally AI tools (Midjourney, ChatGPT with images) only where noted—e.g. for quick mockups or idea exploration, not as the source of truth for final vectors.
- **Brand color(s):** If you have a primary brand hex (e.g. from `_docs/_guides/color-palettes-2026.md` or design system), note it for single-color variants.

Step-by-step process

Step 1 — Audit and establish source of truth

Goal: One definitive source per asset (logo, mark) so all downstream work is consistent.

1. Open both `logo.webp` and `mark.webp` (and `logo.png` / `mark.png` if present) and compare.
2. Record:
 - **Dimensions** (width × height in pixels).
 - **Resolution** (if embedded; for raster, note pixel dimensions).
 - **Background:** transparent or flat color?
 - **Art quality:** crisp edges, aliasing, or compression artifacts?
3. If the PNGs are higher resolution or cleaner, treat them as the raster source; if the WebPs are the only high-res versions, use those. Prefer **vector sources** (SVG, AI, EPS) if the partners can provide them—then use vector as master and export rasters from it.
4. If only raster is available, use the **largest, cleanest** file as the master (e.g. export from Figma/Illustrator at 2× or 3× your largest display size so you can scale down for exports).
5. **Document** the chosen “master” files and their specs in a short table (e.g. in this doc or in `_docs/_guides/logo-asset-manifest.md`).

Recommended tool: Figma or Adobe Illustrator (if you have vectorizable art or can redraw); otherwise Photoshop or Figma for raster audit.

Step 2 — Background removal (transparent alpha)

Goal: Logo and mark usable on any background.

1. If the masters already have transparency, skip to Step 3.
2. **In Photoshop:** Use Select Subject, Magic Wand, or Pen tool to isolate the logo/mark; delete or mask the background; save as PNG (or PSD with layers) to preserve alpha.
3. **In Figma:** Use Remove Background (if available) or trace with vector shapes and remove background fill; export as PNG with transparency.
4. **Alternative (quick check):** ChatGPT with image upload or an online “remove background” tool to get a draft; **re-import into Figma or Photoshop and clean edges** so the master in your project is pixel-perfect.
5. Export **logo** and **mark** as:
 - `logo-transparent.png` (and optionally `.webp`)
 - `mark-transparent.png` (and optionally `.webp`)Use a size that matches or exceeds your largest planned use (e.g. 800px height for logo, 512px for mark).

Recommended tool: Photoshop (raster) or Figma (vector or raster); AI/online tools only for a first pass, then refine in editor.

Step 3 — Color variants (full color, black, white)

Goal: One “primary” full-color version plus single-color versions for light and dark backgrounds.

1. Full color (primary)

- Keep the existing colors as the primary variant.
- Export: `logo-full-color.png` (and/or `.webp`, `.svg`), `mark-full-color.png` (and/or `.webp`, `.svg`), transparent background.

2. Black (for light backgrounds)

- Flatten the logo/mark to a single black (#000000 or near-black) with transparency.
- Export: `logo-black.png`, `mark-black.png` (and SVG if vector).

3. White (for dark backgrounds)

- Same as black but fill with white (FFFFFF).
- Export: `logo-white.png`, `mark-white.png` (and SVG if vector).

4. Optional — brand color

- If you have a primary brand hex, create one variant in that color (e.g. for CTAs or accent use).
- Export: `logo-brand.png`, `mark-brand.png` (and SVG if vector).

Recommended tool: Figma or Illustrator (recolor fills, then export); Photoshop (fill layers, then export). Use the same art and only change fill color so proportions and edges stay identical.

Step 4 — Mark-only exports (icons, favicons, app icons)

Goal: The mark alone, at defined sizes, for favicons, app icons, and UI.

1. Use the **transparent mark** (and vector mark if available).
2. Export at these sizes (adjust to your app's requirements):
 - **16×16** — legacy favicon
 - **32×32** — standard favicon / small UI
 - **48×48** — browser chrome / larger favicon
 - **192×192** — PWA / Android
 - **512×512** — PWA / splash / high-DPI
3. Produce each size in **black** and **white** (and full color if needed) so you can pick by context (light vs dark theme).
4. **Favicon package:**
 - ICO: 16×16 and 32×32 (use an ICO generator from PNGs if your stack doesn't use SVG favicon).
 - Or use a single **SVG favicon** (mark only) if all target browsers support it; then you only need one vector file referenced in the HTML.
5. Name consistently, e.g. `mark-black-16.png`, `mark-white-32.png`, `mark-full-color-192.png`, `favicon.ico`.

Recommended tool: Figma (export at 1×, 2× from frame) or Illustrator/Photoshop (Image Size / Export); use vector mark as source so scaling stays sharp.

Step 5 — Full logo lockup (spacing and minimum size)

Goal: Reproducible full logo (wordmark + mark) with clear space and minimum size.

1. In Figma or Illustrator, build the **horizontal lockup**:
 - Mark + wordmark with fixed spacing (e.g. spacing = 0.25× or 0.5× the height of the wordmark).
 - Align baseline or cap height so the mark and type look optically aligned.
2. Define **clear space**: e.g. "minimum padding on all sides = height of the letter 'M' (or the mark height)." Document this in your asset manifest.
3. Set a **minimum size**: e.g. "Minimum logo height 24px on screen (48px @2x)" and "Do not scale below 24px height."
4. Export:
 - **Horizontal:** `logo-horizontal-[full-color|black|white].svg` (and PNG/WebP at 1× and 2× a reference size, e.g. 200px height).
 - **Stacked (if applicable):** mark above wordmark, same spacing rules; export same variants.
5. If the lockup is only in raster, still document the spacing and minimum size so future vector work or redesign can match.

Recommended tool: Figma or Illustrator (vector); export SVG first, then PNG/WebP from the same art.

Step 6 — Social and Open Graph images

Goal: Correct aspect ratio and size for link previews and social.

1. Create a canvas at **1200×630** (Open Graph) and optionally **1200×1200** or **1024×1024** (square) for Twitter/social.
2. Place the **full logo** (or logo + tagline) centered (or per brand guidelines) on a background that matches your brand (solid or subtle gradient). Ensure the logo is readable at thumbnail size.
3. Export as PNG or WebP (e.g. `og-movemental-1200x630.png`, `social-square-1200.png`).
4. Reference these in your app's `<meta property="og:image">` and Twitter card tags.

Recommended tool: Figma or Photoshop; use the already-prepared logo assets so branding is consistent.

Step 7 — Optimization and format choice

Goal: Small file size and correct format per use.

1. **SVG:** Use for logo and mark wherever the stack supports it (nav, footer, inline); strip unnecessary metadata and collapse transforms to keep file size small.
2. **PNG:** Use when you need alpha and maximum compatibility (e.g. email, some social); compress with OptiPNG or similar (or export "Export for screens" in Figma).
3. **WebP:** Use for web where supported (Next.js, modern browsers); export with alpha and reasonable quality (e.g. 80–90) to reduce size vs PNG.
4. **ICO:** Only for legacy favicon; keep to 16 and 32 if possible.

Recommended tool: Figma (export settings), ImageOptim/Squoosh for post-export compression; build scripts (e.g. sharp in Node) if you want automated WebP/PNG from a single source.

Step 8 — File organization and manifest

Goal: All assets in one place with clear names and documented specs.

1. Directory structure (suggested):

```
public/media-library/images/logo/
├─ logo-horizontal-full-color.svg
├─ logo-horizontal-full-color.png
├─ logo-horizontal-full-color.webp
├─ logo-horizontal-black.svg
├─ logo-horizontal-black.png
├─ logo-horizontal-white.svg
├─ logo-horizontal-white.png
├─ mark-full-color.svg
├─ mark-full-color.png
├─ mark-black.svg
├─ mark-black-16.png
├─ mark-black-32.png
├─ mark-white-32.png
├─ mark-white-192.png
├─ mark-white-512.png
└─ favicon.ico
```

```
├─ og-1200x630.png
├─ (etc.)
```

Keep legacy `logo.webp` / `mark.webp` until the app is updated to use the new filenames, then remove or redirect.

- 2. **Manifest:** Maintain a short table (in this guide or in `logo-asset-manifest.md`) listing:
 - Filename
 - Asset type (full logo, mark, favicon, OG, etc.)
 - Dimensions
 - Color variant
 - Background
 - Suggested use (e.g. "Footer on dark", "Favicon", "PWA icon")
- 3. **App usage:** Update components (e.g. `PublicFooter.tsx`) to reference the appropriate variant by context (e.g. `mark-white` on dark footer, `logo-black` on light header) and use SVG where possible.

Tool summary

Step	Primary tool	Alternative
1. Audit	Figma, Illustrator	Photoshop
2. Background removal	Photoshop, Figma	Online/AI + cleanup in editor
3. Color variants	Figma, Illustrator	Photoshop
4. Mark/icon sizes	Figma, Illustrator	Photoshop (Image Size / Export)
5. Logo lockup	Figma, Illustrator	—
6. OG/social	Figma, Photoshop	—
7. Optimization	Figma export, Squoosh/ImageOptim	sharp (Node)
8. Organization	File system + manifest doc	—

Midjourney / ChatGPT with images: Use only for exploration or mockups (e.g. "how might this mark look on a dark card?"). Do **not** use them as the source of final logo/mark files; final assets should come from your mastered art in Figma, Illustrator, or Photoshop.

Checklist before calling it done

- ☐ Master source (logo + mark) chosen and documented.
- ☐ Transparent-background versions for logo and mark.
- ☐ Color variants: full color, black, white (and optional brand).
- ☐ Mark exported at 16, 32, 192, 512 (and any other required sizes).
- ☐ Favicon (ICO or SVG) in place and referenced in HTML.
- ☐ Full logo lockup with clear space and minimum size documented.

- ☐ OG/social image(s) at 1200×630 (and square if needed).
 - ☐ All exports optimized (SVG cleaned, PNG/WebP compressed).
 - ☐ Files named consistently and placed under `public/media-library/images/logo/` (or your chosen path).
 - ☐ Manifest or table updated with filename, type, dimensions, and usage.
 - ☐ App (e.g. footer, head) updated to use the correct asset per context.
-

References

- **Project design tokens:** `_docs/_guides/color-palettes-2026.md`, `_docs/_guides/typography-2026.md`
- **Current usage:** `components/shared/PublicFooter.tsx` uses `mark.webp` and `logo.webp`; update to new filenames/variants when the asset set is ready.