ChromaForge
COLOR COMPILER

PRODUCT PRE-ANNOUNCEMENT

Coming Soon from Mandelbrot Metal: ChromaForge

The color compiler for developers and designers, built on the palette technology behind Mandelbrot Metal.

Website: chromaforge.dev
Supported devices: iPhone, iPad, and Mac (iOS/iPadOS/macOS 26+)
Audience: Developers, web designers, Figma users, app builders, and design-system teams
Launch model: Free to explore; Pro for production color-system workflows

The Promise

ChromaForge is being designed as a true color compiler: it moves creators from color inspiration to usable color infrastructure, producing palettes that can be previewed, repaired, tokenized, mapped to framework and Figma roles, exported into code, and audited in CI workflows. The latest build adds in-app guidance, a sortable and searchable palette library, favorite/duplicate/restore actions, and undo for recent palette edits, making the workflow safer for experimentation and clearer for first-time users.

The core idea: Start with color inspiration, forge it into a perceptual palette, validate accessibility, then ship design-system tokens and exports.

Mandelbrot Metal began as a fast, beautiful fractal explorer, but one of its most powerful internal technologies has always been its palette engine.

Inside Mandelbrot Metal, color is not decoration. It shapes structure, depth, atmosphere, legibility, and emotional impact. The palette system must support smooth gradients, precise LUT behavior, high-impact contrast, wide-gamut color, and subtle transitions across complex visual fields.

That technology opened up a larger product opportunity: what if the same palette intelligence behind Mandelbrot Metal became a dedicated compiler for turning raw colors into production-ready design tokens, Figma variables, themes, accessibility checks, guided workflows, reusable palette libraries, and framework code?

That is the idea behind ChromaForge, a new product from Mandelbrot Metal coming soon. The latest build extends the concept with an in-app User Guide, a sortable palette library, favorites, palette duplication, factory restore, and undo for recent palette edits. The name is intentional: ChromaForge is built to forge color into code.

ChromaForge
COLOR COMPILER

Why Build ChromaForge?

Developers, web designers, and Figma users already have many ways to generate attractive swatches. The harder problem is compiling those colors into a working interface system. A palette must survive dark mode, high-contrast needs, text readability, component states, focus rings, design-tool variables, developer handoff, and CI checks.

ChromaForge is built for the practical middle ground between visual exploration and implementation. It keeps the creative feel of color experimentation while adding the compiler-like workflow production teams need: source, forge, validate, and ship.

Who It Is For

- Developers who need app-ready tokens, CSS variables, and framework themes instead of manually translating swatches into code.
- Web designers and Figma users who want to test color decisions against real interface states, accessibility requirements, variable modes, and developer handoff formats before implementation.
- App builders who need SwiftUI-ready colors, theme variants, native mobile exports, and accessible component previews.
- Design-system teams that need primitive, semantic, component-level, Figma-variable, and framework-specific tokens with reliable exports.
- Technical founders and indie developers who want a fast way to turn brand colors into production-ready themes, app surfaces, and launchable code.

What ChromaForge Will Include

Area	Capabilities
Palette authoring	Palette browser with favorites/search, built-in/imported/custom filters, Sort menu for Favorites, Newest, Oldest, Name, Most Stops, Fewest Stops, Type, Color Space, and LUT Size, row context menus and swipe actions, large gradient preview, exact-LUT preview, OKLCH/perceptual generation, locked stops, stop editing, Undo Last Palette Change, Restore Factory for edited built-ins, Display-P3/sRGB controls, LUT metadata, step markers, dithering preview, and editable notes.

Area	Capabilities
Imports	Manual stops, photo sampling, local file import, website import from HTML, inline styles, linked CSS, and modern CSS color functions. Supported inputs include ChromaForge JSON, Mandelbrot Metal palette JSON, design-token JSON, CSS/Sass/text color files, CSV, GIMP GPL, Adobe ASE, hex, RGB/RGBA, HSL/HSLA, HWB, CIELAB/LCH, OKLab/OKLCH, and color(display-p3 ...).
Color compiler engine	RGB, Display-P3, CIELAB, CIELCH, OKLab, and OKLCH serialization; perceptual interpolation; luminance and contrast calculations; APCA-style readability preview; perceptual distance for deduping, diffing, and accent selection.
Semantic compilation	Light, dark, and high-contrast theme matrix; primitive tokens; semantic role tokens; component tokens; Figma-variable handoff; one-click AA contrast repair; and palette diff inspection across stops and semantic roles.
Role presets	ChromaForge native roles plus practical mappings for shadcn/ui, Material 3-inspired systems, Apple UI-inspired systems, and data visualization palettes.
Preview and accessibility	Context preview, component preview lab, WCAG contrast inspection, non-text contrast checks, semantic theme checks, APCA-style Lc preview, system audit, modern swatch inspector, and color-vision simulation.
Developer exports	Raw CSS, Semantic CSS, modern CSS with OKLCH, CIELAB/CIELCH diagnostics, P3/fallbacks/color-mix/light-dark, Tailwind classic, Tailwind v4 @theme, shadcn/ui, Sass, Figma Variables JSON, SwiftUI, Android XML, Jetpack Compose, Flutter, React Native, DTCG 2025.10, Tokens Studio, Style Dictionary, CSV, and Mandelbrot Metal JSON.
Guidance, CLI, and CI path	In-app Info menu with About, User Guide, and App Store Review entry points; multi-page User Guide covering workflow overview, palette workflow, validation, handoff/export, troubleshooting, FAQ, glossary, and standards; plus a starter command-line audit script for extracting hex colors, summarizing WCAG pair coverage, reporting weakest/strongest pairs, and failing builds when contrast requirements are not met.

How ChromaForge Helps Developers

For developers, ChromaForge reduces the time between "these colors look good" and "these colors work in the app." It can generate semantic roles such as background, foreground, surface, primary, accent, success, warning, and danger, then preview those colors in real interface patterns before exporting production assets. The current build also adds safer authoring mechanics, including sortable palette libraries, favorites, duplication, factory restore, and Undo Last Palette Change for recent palette edits.

The Developer Handoff workflow is intended to produce practical assets: modern CSS, Semantic CSS, Tailwind and Tailwind v4 themes, shadcn/ui variables, Figma Variables JSON, SwiftUI Color extensions,

Swift asset-catalog planning, Android XML, Jetpack Compose, Flutter ColorScheme, React Native themes, DTCG tokens, Tokens Studio JSON, Style Dictionary JSON, CSV, and Mandelbrot Metal JSON.

How ChromaForge Helps Web Designers

For web designers and Figma users, ChromaForge makes it easier to validate color decisions before they become design debt. A palette may look beautiful as swatches but fail when applied to text, disabled states, alerts, buttons, charts, sidebars, forms, focus rings, or variable modes. ChromaForge lets designers inspect those cases before handoff.

Website import creates a useful bridge from an existing brand presence to a working color system. Designers can start from a live site, extract likely brand and theme colors from HTML, inline styles, linked CSS, and modern CSS color functions, then refine them into semantic tokens, component aliases, and Figma-ready variable modes. The new User Guide and library controls also make ChromaForge easier to evaluate during beta review, client exploration, and repeated design iteration.

Planned Launch Pricing

ChromaForge is planned as a freemium product: free to explore, Pro for production-ready color compilation, audits, role mapping, Figma handoff, and advanced exports.

Plan	Price	Cadence	What It Unlocks
Free	$0	Forever	5 custom/imported palettes, 2 website imports, photo import, file import, in-app User Guide, searchable/sortable palette library, core favorite, duplicate, restore, and undo editing controls, basic contrast checks, JSON, Mandelbrot Metal JSON, and raw CSS variable exports.
Pro Annual	$19.99	Per year launch price	Unlimited palettes and website imports, expanded file-import workflows, workflow-oriented validation/handoff, theme matrix, token layers, Figma Variables export, AA repair, component previews, system audit, role preset mapping, palette diffing, and all advanced exports. Planned to move to $29.99/year after launch validation.
Pro Monthly	$2.99	Per month	Same Pro feature access with flexible monthly billing for short client projects or occasional use.
Founder Lifetime	$49.99	One time	All Pro features for early supporters. Planned to move to $69.99-$79.99 after the founder period.

ChromaForge
COLOR COMPILER

The Direction

ChromaForge is not meant to be another casual palette generator. It is a color compiler and working palette library: a focused environment for sourcing, forging, validating, organizing, safely iterating, role-mapping, diffing, and exporting color systems.

The product is still in development, but the direction is clear: take the color technology and visual experimentation DNA of Mandelbrot Metal, then focus it into a practical tool for people building websites, apps, native interfaces, and design systems across iPhone, iPad, and Mac. The newest version strengthens that direction with guided help, launch-ready app information surfaces, sorted and favorited libraries, context actions, restore paths, and undo.

Color Compiler Pipeline

ChromaForge is organized around a workflow surface and compile pipeline rather than a one-screen palette picker. The user-facing workflow is Source, Forge, Validate, and Ship; the in-app User Guide explains that workflow, and underneath, each compiler stage makes color more useful to a production team.

- Source: Build, sample, import, extract, and organize colors through manual stops, photos, palette files, token files, websites, CSS, swatch formats, and the searchable, filterable, sortable palette library.
- Normalize: Convert colors into RGB, Display-P3, CIELAB, CIELCH, OKLab, and OKLCH representations so the system can reason about perceptual behavior, browser color syntax, wide-gamut output, and token handoff.
- Analyze: Check WCAG contrast, non-text contrast, color-vision resilience, APCA-style readability, and export readiness.
- Map: Generate primitive, semantic, component, Figma-variable, and framework-specific roles.
- Emit: Export CSS, framework code, Figma Variables JSON, design-token JSON, native platform code, CSV, Mandelbrot Metal JSON, and CI-friendly audit output.

Supported Inputs and Outputs

Category	Supported items
Inputs	Manual color stops; photo import; file import; website URL import; ChromaForge JSON; Mandelbrot Metal palette JSON; design-token JSON; CSS/Sass/text color import; CSV; GIMP GPL; Adobe ASE; HTML extraction; inline CSS; linked CSS; hex; RGB/RGBA; HSL/HSLA; HWB; CIELAB/LCH; OKLab/OKLCH; color(display-p3 ...).

Category	Supported items
Web outputs	Raw CSS variables; Semantic CSS; modern CSS with OKLCH, CIELAB/CIELCH diagnostics, Display-P3, hex fallbacks, @supports, light-dark(), and color-mix(); Sass variables.
Framework outputs	Tailwind classic config; Tailwind v4 @theme; shadcn/ui theme variables; SwiftUI Color extension; Swift asset-catalog planning; Android XML; Jetpack Compose Material 3 color schemes; Flutter ColorScheme; React Native theme object.
Design-token outputs	DTCG 2025.10 JSON, Tokens Studio JSON, Style Dictionary JSON, Figma Variables JSON, CSV, and Mandelbrot Metal palette JSON.
Audit outputs	System readiness score, WCAG/non-text checks, APCA Lc preview, color-vision simulation, palette diff summary, Figma handoff summary, User Guide troubleshooting context, and CLI audit results.

Glossary

The following glossary defines the acronyms, abbreviations, standards, formats, and technical terms ChromaForge uses or references.

Term	Meaning	Used for
AA	WCAG contrast target for normal text.	Accessibility checks
AAA	Stricter WCAG contrast target.	Accessibility checks
Adobe ASE	Adobe Swatch Exchange palette file.	File import
APCA	Perceptual contrast model for modern accessibility scoring.	Contrast preview
CI	Continuous integration checks run on every change.	Team workflows
CIELAB	Device-independent perceptual color space with L, a, and b axes.	Color diagnostics
CIELCH	Cylindrical form of CIELAB using lightness, chroma, and hue.	Hue analysis
CLI	Command-line interface for automated token and palette work.	Automation
Color gamut	Range of colors a device or space can represent.	P3 and sRGB

Term	Meaning	Used for
CSS	Stylesheet language used by web projects.	Web exports
CSS Color 4/5	Modern CSS syntax including OKLCH, Lab, LCH, and Display-P3.	Web exports
CSV	Comma-separated palette rows.	Import and export
Display-P3	Wide-gamut RGB color space common on Apple displays.	P3 workflows
DTCG	Design Tokens Community Group format.	Token exports
File import	Local palette ingestion from JSON, ASE, GPL, CSS, CSV, and text.	Palette setup
Flutter	Google UI toolkit for cross-platform apps.	Export targets
GIMP GPL	GIMP palette file format.	File import
HSL/HSLA	Hue, saturation, lightness, and optional alpha.	Color editing
HTML	Markup language used for web pages.	Website import
HWB	Hue, whiteness, and blackness color notation.	Color parsing
Jetpack Compose	Kotlin UI framework for Android.	Export targets
JSON	Structured data format used by palettes and tokens.	Import and export
Lab	CSS CIELAB notation.	CSS diagnostics
LCH	CSS cylindrical Lab notation.	CSS diagnostics
LUT	Lookup table used for color mapping.	Color pipelines
Mandelbrot Metal JSON	Palette JSON exported by Mandelbrot Metal.	File import
Material 3	Google design system with role-based color tokens.	Export targets

Term	Meaning	Used for
OKLab	Modern perceptual color space optimized for smooth gradients.	Color diagnostics
OKLCH	Cylindrical OKLab notation with lightness, chroma, and hue.	CSS exports
P3	Short name for Display-P3.	Wide-gamut colors
Perceptual color	Color model designed around how people see differences.	Better ramps
React Native	JavaScript framework for native apps.	Export targets
RGB/RGBA	Red, green, blue, and optional alpha.	Color values
Sass	CSS preprocessor with variables.	Web exports
Semantic token	Named role such as primary, surface, or error.	Design systems
shadcn/ui	React component system driven by CSS variables.	Export targets
sRGB	Standard RGB space for screens and web content.	Baseline colors
StoreKit	Apple framework for in-app purchases.	Subscriptions
Style Dictionary	Amazon token build system.	Token pipelines
SwiftUI	Apple declarative UI framework.	Export targets
Tailwind	Utility-first CSS framework.	Web exports
Token	Named design value such as color, spacing, or radius.	Design systems
Tokens Studio	Design-token plugin and JSON format.	Figma workflows
UI	User interface.	App design
WCAG	Accessibility guidelines for contrast and readability.	Compliance
XML	Markup format used by some platform resources.	Exports
Figma Variables	Figma color variable collection with Light, Dark, and High Contrast modes.	Design handoff

ChromaForge
COLOR COMPILER

Term	Meaning	Used for
Variable mode	Mode-specific variable values such as Light, Dark, and High Contrast.	Figma workflows
Workflow surface	Top-level Source, Forge, Validate, and Ship command area documented in the in-app User Guide.	Product workflow

Sample Screenshots