

IMPROVING DIGITAL BRANDS ONLINE

Extract any website's design system in one command.

Every other tool describes intent. Dembrandt observes **reality**, the one layer that sees what actually shipped.

Logo, colors, typography, borders, spacing and tokens, pulled out of any live site in one command. Consumable as design.md, DTCG tokens, or a full brand guide.

 zsh

```
$ > npx dembrandt example.com  
✓ extracted 142 tokens | wrote design.md, tokens.json
```

[Open the app](#)[View on GitHub](#)

What's inside.

01	One-page overview	Dembrandt in a single page
02	About Dembrandt	The problem, and the mission
03	Key features	What the CLI & app extract
04	Use cases & who it's for	Five audiences
05	How it works	Playwright in, tokens out
06	Visual assets & media	Logos, screenshots, downloads
07	Results & social proof	Traction and what people say
08	Pricing, licensing & contact	MIT, the app, and how to reach us

Dembrandt in one page.

Dembrandt is a developer tool that reverse-engineers any website's design system into tokens, and tracks token **drift** before it breaks a brand. One command. design.md, DTCG, or a brand guide out.

- Extract logo, colors, type, spacing & borders in ~10 seconds
- Emit W3C DTCG tokens, design.md, or a brand guide
- Compare snapshots and flag **drift** over time
- Feed **agentic pipelines** via MCP (Claude, Cursor)

● ● ● zsh

```
$ > dembrandt dembrandt.com
crawling | computing styles | normalizing...
✓ 10 colors | 46 text styles | 7 radii | 5 breakpoints
✓ wrote DESIGN.md tokens.dtcg.json
```

EXTRACTED TOKENS, BY CATEGORY

Design tokens	Colors Typography Spacing Radius Borders Shadows Gradients Motion
Components	Buttons Inputs Links Badges
Brand & system	Logo Favicons Breakpoints Icons Frameworks WCAG contrast

DTCG

W3C design tokens out

10 s

To a full token set

MIT

Open-source CLI

Design tokens shouldn't be copied by hand.

Every team that cares about a brand keeps a design system: colors, type, spacing, components. Keeping it in sync with what's actually shipped is tedious, manual, and quietly drifts out of date. We built Dembrandt to make that work mechanical: point it at a URL and read the real, computed design system back as structured tokens.

■ THE PROBLEM

Tokens get collected by hand: eyedropping hexes, measuring spacing, copying font stacks. It's slow, error-prone, and stale the moment a page changes. Drift goes unnoticed until the brand looks broken.

■ THE SOLUTION

One command crawls the live site, reads computed styles, and emits a normalized token set. Re-run it to compare snapshots and surface drift over time, in your terminal or in the app.

MISSION

Make the real, shipped design system of any site measurable, in one command.

VISION

Better digital brands online, where design intent and what actually ships never drift apart.

What Dembrandt extracts.

01

Token extraction

Colors, typography, spacing, borders, radius, shadows, motion, breakpoints, logo and components, read from computed styles.

02

DTCG / W3C tokens

Output the industry-standard Design Token Community Group format. Drop-in for any toolchain.

03

Output formats

Five primary deliverables: JSON (default), W3C DTCG tokens, DESIGN.md, a self-contained HTML report, and a printable PDF brand guide.

04

Dark mode, mobile & WCAG

Light and dark themes, a 390×844 mobile viewport, and WCAG 2.1 contrast grading on real DOM pairs.

05 **BETA**

App | drift tracking

Load extractions, browse a snapshot timeline, and diff versions to flag drift.

06

AI agent / MCP

Ships as an MCP server: agents like Claude Code and Cursor pull tokens directly via tools such as `get_design_tokens`.

`--dtcg``--design-md``--brand-guide``--crawl 5``--wcag`

● ● ● add the MCP server

```
$ > claude mcp add --transport stdio dembrandt -- npx -y --package dembrandt dembrandt-mcp
```

Built for everyone who ships a brand.

01 **Frontend developers**

Bootstrap a token file from any reference site instead of hand-building one.

02 **Design systems & DesignOps**

Audit the live system, document its state, and watch for drift over time.

03 **Agencies & freelancers**

Reverse-engineer a client's brand in seconds and deliver a clean handoff.

04 **AI agent developers**

Feed structured design context into agentic pipelines over MCP.

05 **Enterprises**

Enforce design consistency and prevent drift across many properties.

Playwright in. Tokens out.

A headless browser loads the real site and reads **computed styles**, not source CSS, so what you get is what users actually see. Values are normalized, de-duplicated, and grouped into roles.



01

Crawl

Playwright loads N pages

02

Compute

Read computed styles

03

Normalize

Group & de-dupe roles

04

Emit

DTCG | md | guide

DRIFT DETECTION IN CI

```
● ● ● .github/workflows/drift.yml
```

```
# run on every pull request
$ > npx dembrandt $DEPLOY_URL --dtcg --compare baseline.json
x 2 drifts | brand-500 #38BDF8 → #3AA0FF
```

Logos & screenshots.

 Dembrandt.

wordmark | white

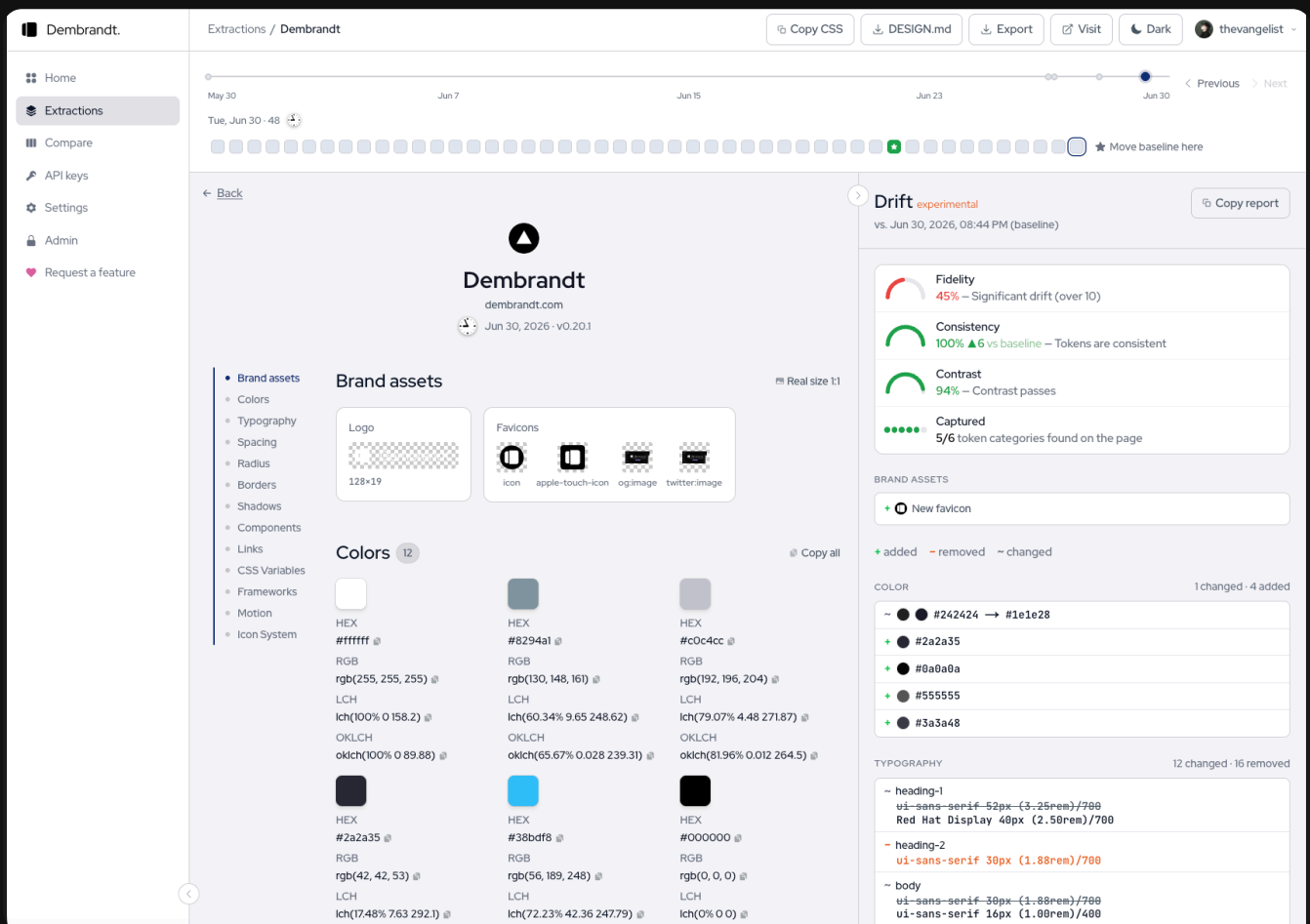
 Dembrandt.

wordmark | black



symbol

App | drift & extractions



The screenshot displays the Dembrandt app interface, which is used for managing brand assets and analyzing design drift. The interface is divided into several sections:

- Header:** Includes the Dembrandt logo, navigation links (Home, Extractions, Compare, API keys, Settings, Admin, Request a feature), and a timeline for extractions from May 30 to June 30.
- Brand assets:** A section for managing brand assets, including colors, typography, spacing, radius, borders, shadows, components, links, CSS variables, motion, frameworks, and icon system.
- Colors:** A section for managing colors, showing a grid of color swatches with their respective HEX, RGB, and LCH values. The colors include white (#ffffff), light gray (#f2f2f2), dark gray (#2a2a35), and black (#000000).
- Drift analysis:** A section for analyzing design drift, showing a timeline of extractions and a detailed report for a specific drift event. The report includes metrics for Fidelity (45%), Consistency (100%), Contrast (94%), and Captured (5/6 token categories).
- Brand assets table:** A table listing brand assets, including colors and typography. The table has columns for asset name, value, and status (added, removed, changed).

High-res asset pack

SVG and PNG wordmarks, symbol, favicon and OG image, light and dark.

[Brand assets](#)

Open, adopted, dogfooded.

★ 2 000+

GitHub stars

200 000+

npm downloads

5 000+

Skill installs

MIT-licensed CLI, 40+ agent design skills, and GitHub Sponsors on the repo. Dembrandt dogfoods itself: drift gating runs in CI on dembrandt.com.

"This team has some super interesting stuff cooking! Goes well beyond DESIGN.md. Brand governance at scale in an agentic world."

TC

Trevin Chow

@trevin | Jun 28

OPEN-SOURCE ECOSYSTEM

dembrandt

dembrandt-skills

dtcg-validator

awesome-agent-skills

dembrandt-next

Backed by internal R&D: an ONNX ML model and a growing brand-guide corpus.

Free CLI. Free app.

CLI

Free

MIT license

- ✓ Full token extraction
- ✓ DTCG | design.md | guide
- ✓ Unlimited local runs

APP BETA

Free

Beta, GitHub sign-in

- ✓ Hosted extractions
- ✓ Drift tracking & Compare
- ✓ Snapshot timeline

SPONSOR

Premium

GitHub Sponsors

- ✓ Support open source
- ✓ Roadmap influence
- ✓ Logo on the site

The CLI and the app are both free. The CLI is MIT-licensed open source; the app is in beta, sign in with GitHub. Press: dembrandt@tutamail.com.

Get in touch.

PRESS & PARTNERSHIPS

dembrandt@tutamail.com

Open the app

`npm i -g dembrandt`

PRODUCT

dembrandt.com

App

Validator

Explorer

Integrations

PACKAGE

npm

GitHub

Documentation

Releases

COMMUNITY

Discussions

Issues

Sponsor

Buy me a coffee

CONNECT

X | [@dembrandtcom](https://twitter.com/dembrandtcom)

LinkedIn

YouTube

Newsletter

At a glance.

Product	A CLI that extracts any public site's design system, logo, colors, type, borders and shadows, in one command
Latest version	v0.21.0, June 2026 (first released 2025), active release cadence
Core command	<code>npx dembrandt dembrandt.com</code> no install, no signup
Built with	Node.js 18+, TypeScript, Playwright (Chromium), ONNX ML model
ML layer	Trained model picks the right tokens, offline, no Python
AI / MCP	dembrandt-mcp serves tokens to Claude Code, Cursor and Windsurf
Output	Design tokens, including the W3C DTCG standard
License	MIT, CLI and ML layer, fully open source
Maintainer	GitHub org dembrandt, npm publisher thevangelist
Find it	npm: dembrandt github.com/dembrandt/dembrandt dembrandt.com

The tool I kept wanting to exist.



Designer & developer

Finland. 15 years in UX & front-end.

Design audits used to mean opening DevTools, copying hex codes, writing down font stacks, building a spreadsheet of spacing values. I did that for years across client projects. Every handoff, the same questions. Every project, the same manual groundwork.

Dembrandt is the tool I kept wanting to exist. One command. Any live website. A complete set of design tokens. No plugins, no access negotiations, no manual screenshots. If a browser can open it, Dembrandt can extract it.

THE NAME

Dembrandt nods to **Rembrandt**, a master of observing and rendering reality, light and shadow read exactly as they fall. Folded inside is **de-brand**: take a finished brand apart and read it back as its raw tokens. Observation in, tokens out.

My background is 15 years in UX and front-end development. The gap between design intent and what actually **ships** has been the consistent frustration across all of it. Dembrandt addresses that gap at the token level, where it is measurable and enforceable rather than subjective.

The CLI is open source and free. The work continues.