AI TOOLS WIKI · MÉMO

Taste Skill *mémo*

L'essentiel pour utiliser **Taste Skill**, la collection de skills qui évite à Claude Code, Codex ou Cursor de produire des interfaces génériques (« AI slop »). Chaque skill fait une seule chose ; on n'installe que celles dont on a besoin.

Famille Design en code · **Éditeur** Leonxlnx · **Licence** MIT
Note du wiki 8/15 · **Commandes vérifiées le** 25 septembre 2026
Fiche, notes et avis aitoolswiki.viellard.fr/outils/design-en-code/taste-skill

Installation

`npx skills add https://github.com/Leonxlnx/taste-skill`	Toutes les skills
`npx skills add https://github.com/Leonxlnx/taste-skill --skill "design-taste-frontend"`	Une seule, par son nom d'installation

Le nom d'installation est le champ `name:` du `SKILL.md`, pas le nom du dossier. On peut aussi copier un `SKILL.md` dans le projet ou le coller dans une conversation.

Les skills et quand les utiliser

Skills qui produisent du code

Nom d'installation	Quand l'utiliser
`design-taste-frontend`	**Le choix par défaut.** Lit le brief, en déduit la direction visuelle, puis règle trois curseurs (v2, expérimentale).
`design-taste-frontend-v1`	L'ancienne version, si la v2 casse quelque chose dans ton workflow.
`gpt-taste`	Variante plus stricte, pensée pour GPT/Codex.

Les skills et quand les utiliser (suite)

Nom d'installation	Quand l'utiliser
`redesign-existing-projects`	Améliorer un projet existant : audit de l'UI, puis corrections.
`image-to-code`	Générer des maquettes en image, les analyser, puis coder le site.
`high-end-visual-design`	Style haut de gamme et calme : contrastes doux, espace, typographie premium.
`minimalist-ui`	Style éditorial sobre, façon Notion ou Linear.
`industrial-brutalist-ui`	Style brutaliste : typographie suisse, forts contrastes.
`stitch-design-taste`	Règles compatibles Google Stitch, export `DESIGN.md` possible.
`full-output-enforcement`	Quand l'agent rend du travail inachevé ou des commentaires « à compléter ».

Skills qui produisent des images

Nom d'installation	Livrable
`imagegen-frontend-web`	Maquettes de sites (hero, landing page).
`imagegen-frontend-mobile`	Écrans et parcours mobiles.
`brandkit`	Planches d'identité : pistes de logo, palettes, typographies.

Ces trois skills s'utilisent avec un générateur d'images (ChatGPT Images…) ; on donne ensuite les images à l'agent de code.

Les trois curseurs (skill par défaut)

Réglés en haut du `SKILL.md`, de 1 à 10 :

Curseur	1	10	Valeur de base
`DESIGN_VARIANCE`	Symétrie parfaite, centré	Asymétrique, expérimental	8
`MOTION_INTENSITY`	Statique, simple survol	Animations cinématiques au scroll	6
`VISUAL_DENSITY`	Aéré, façon galerie	Dense, façon tableau de bord	4

Conseils d'utilisation

- **Commencer par** `design-taste-frontend`, puis ajouter une skill de style (`minimalist-ui`, `high-end-visual-design`…) une fois la direction choisie.
- **Baisser les curseurs pour un outil métier** : la skill par défaut vise surtout les landing pages, portfolios et refontes, pas les tableaux de bord ni les formulaires complexes.
- **Donner un vrai brief** (public, marque, ambiance) : la skill en déduit la direction visuelle avant d'écrire du code.
- **Pour** `image-to-code`, écrire explicitement : « suis la skill : génère les images, puis analyse, puis code ».
- **Sur un projet existant**, préférer `redesign-existing-projects` : elle audite avant de modifier.
- **Ne pas empiler les styles** : une seule skill de direction visuelle à la fois.

Sources (documentation officielle)

- README Taste Skill
- SKILL.md de la skill par défaut
- Documentation tasteskill.dev