

skillify

skillify

“ Catalogue généré le 2026-05-11

En une phrase

Transforme le dernier scraping réussi (via `/scrape`) en un skill permanent et automatisé, pour que la même extraction reparte en 200 millisecondes au lieu de tout réapprendre.

Quand l'utiliser

- Tu viens d'utiliser `/scrape` pour extraire des données d'un site, ça marche, et tu veux le réutiliser plus tard.
- Tu veux automatiser une extraction récurrente (front page d'un site, listing, stats).
- Tu veux passer d'un scraping piloté pas-à-pas par un agent à un script déterministe rapide.
- Tu veux partager ton extraction avec d'autres projets (mode global) ou la garder pour un projet précis.

Comment l'invoquer

- **Slash command** : `/skillify`
- **Phrases déclencheurs (texte)** : "skillify", "codify this scrape", "save this scrape", "make this permanent", "codify"
- **Auto-invocation** : Sur demande explicite (typiquement après un `/scrape` réussi).

Description détaillée

Le skill `skillify` est le multiplicateur de productivité du système de scraping de gstack. La logique : `/scrape` est lent parce que Claude pilote le navigateur en direct, page par page, sélecteur par sélecteur, en essayant des options jusqu'à trouver. Une fois que ça marche, c'est dommage de tout refaire à chaque fois. `/skillify` prend le dernier scraping réussi de la conversation et le transforme en un script Playwright propre, testé, et réutilisable.

Le workflow est strict pour ne jamais livrer un skill cassé. Il commence par retrouver le dernier `/scrape` valide dans la conversation (max 10 tours en arrière). Il propose un nom de skill court, des phrases déclencheurs (3 à 5), et te demande où le ranger : niveau global (`~/.gstack/browser-skills/`, accessible à tous tes projets) ou niveau projet (uniquement ce dépôt). Il synthétise ensuite un `script.ts` (la logique d'extraction sous forme de fonction pure), un `script.test.ts` (un test qui rejoue contre une capture HTML enregistrée comme fixture), et capture cette fixture (HTML figé du site cible). Tout est écrit dans un dossier temporaire d'abord.

Le test tourne dans ce dossier temporaire. S'il passe ET que tu approuves explicitement, le skill est promu dans son emplacement définitif. Sinon, le dossier temporaire est supprimé entièrement : pas d'état "à moitié livré". Une fois en place, les prochaines fois que tu lances `/scrape` avec une intention similaire, ce script déterministe tourne en environ 200 millisecondes — sans agent, sans navigation manuelle. Limitations à savoir : un seul URL par skill, les fixtures peuvent devenir obsolètes si le site change, et ce skill ne gère pas les flows qui modifient le site (formulaires, clics destructifs) — c'est le job de `/automate`.

Source

- **Plugin** : `gstack`
- **Nom interne** : `skillify`
- **Fichier** : `/home/thymon/.claude/skills/gstack/skillify/SKILL.md`

Revision #2

Created 2026-05-11 21:19:18 UTC by thymon

Updated 2026-05-11 21:36:57 UTC by thymon