AI TOOLS WIKI · MÉMO

Glif *mémo*

L'essentiel pour utiliser **Glif MCP**, le serveur MCP hébergé de Glif qui permet à Claude Code ou Codex de générer images, vidéos et audio, transcrire, rendre du HTML ou enchaîner plusieurs opérations média, à partir d'une simple description.

Famille Image, vidéo et médias · **Éditeur** Glif · **Licence** MIT (dépôt de métadonnées ; serveur hébergé propriétaire)
Note du wiki 10/15 · **Commandes vérifiées le** 25 septembre 2026
Fiche, notes et avis aitoolswiki.viellard.fr/outils/image-video-medias/glif-mcp

Installation

`claude mcp add --scope user --transport http glif "https://glif.app/api/mcp"`	Claude Code
`codex mcp add glif --url "https://glif.app/api/mcp"` `codex mcp login glif`	Codex

Au premier appel, le client ouvre une connexion OAuth dans le navigateur pour lier ton compte Glif : pas de clé API à gérer. Page d'installation pour d'autres clients : glif.app/mcp.

Outils MCP

Outil	Ce qu'il fait
`compose_project`	**L'outil principal.** Décrire ce que tu veux en langage naturel (y compris une série de variantes) : Glif choisit les modèles et enchaîne les étapes. Renvoie tout de suite un identifiant de tâche. Niveaux : `lite`, `smart`, `genius`.
`get_job_status`	Suit l'avancement d'une tâche ; renvoie les médias (liens CDN) une fois terminée.
`view_media`	Affiche un média terminé.
`get_project` / `list_projects`	État d'un projet, liste des projets récents.
`upload_file`	Envoie un média (image de référence, audio…) à utiliser dans un projet.
`list_user_skills` / `get_user_skill`	Accède à tes skills Glif personnelles.
`whoami`	Compte, forfait, **solde de crédits** et dépenses.

Le déroulé type

1. `compose_project` avec la description : « Un portrait noir et blanc façon film noir d'un paresseux détective, en 4 variantes »
2. `get_job_status` jusqu'à la fin (souvent moins d'une minute, parfois plus)
3. `view_media` pour voir le résultat
4. Pour itérer, rappeler `compose_project` **avec le même** `projectId`

Conseils d'utilisation

- **Ne jamais relancer** `compose_project` **pendant qu'une tâche tourne** : ça démarre une seconde tâche, facturée elle aussi. Attendre avec `get_job_status`.
- **Continuer dans le même projet** (`projectId`) pour les retouches : Glif garde le contexte et les médias précédents.
- **Vérifier les crédits avec** `whoami` avant une série de générations ; seules les générations consomment des crédits, les lectures sont gratuites.
- **Demander les variantes en une seule description** plutôt qu'en plusieurs appels.
- **Choisir le niveau** : `lite` pour un essai rapide, `genius` pour un résultat soigné qui coûte plus.
- **L'ancien serveur local npm (**`@glifxyz/glif-mcp-server`**) est obsolète** : utiliser l'URL hébergée.

Sources (documentation officielle)

- README glif-mcp-server
- Documentation pour agents (llms.txt)
- Page d'installation MCP