Se connecter
Proposer un contenu
Identifiant
Identifiant
Mot de passe
Mot de passe
Connexion automatique
Se connecter
Pas de compte ? S'inscrire...

Journal : Lire les effets de bord d'une compétence Markdown : deux fichiers à comparer

7 nov. 2026
-20

Posté par **Aileen Smith** le 07 novembre 2026 à 19:15. **Licence CC By-SA.**
Étiquettes : **mcp**, **spam**, **genai**

Un fichier `SKILL.md` peut décrire une tâche très simple tout en conduisant à plusieurs appels d'outils. Pour comprendre ce qui va se passer, il faut suivre les verbes jusqu'au bout : chercher, rédiger, enregistrer et envoyer ne désignent pas le même résultat.

Voici une lecture comparée de deux fichiers publics de la collection claude-code-skills. Le dépôt est sous licence MIT. Les exemples ci-dessous portent sur la révision `a96801a5d088243c181c9a79b9bbffe83e769aea`, afin que la discussion ne dépende pas d'un fichier modifié entre-temps.

Les instructions sont inspectables, mais leur utilisation demande un compte sur le service hébergé et une connexion autorisée à son MCP. Le dépôt ne contient pas de quoi auto-héberger ce service.

Première lecture : quel est le dernier effet attendu ?

Dans closedloop-ship-notify, le nom pourrait faire penser à un envoi de notification. Le déroulé décrit autre chose : trouver les clients concernés par une fonctionnalité livrée, puis préparer un message à partir des éléments retournés.

L'appel central est `find_launch_audience`. Le fichier demande d'utiliser un périmètre précis, de conserver les éléments justificatifs et d'afficher que rien n'a été envoyé. Il interdit aussi de transformer cette préparation en création de notification ou en journalisation d'un envoi.

Le résultat à relire est donc un brouillon accompagné de ses sources. Une adresse et un objet présents dans ce brouillon ne prouvent pas qu'un courrier a quitté le système.

Dans closedloop-watch, l'objectif comprend au contraire l'enregistrement d'une règle réutilisée après la conversation. C'est une différence de cycle de vie, pas seulement de formulation.

Deuxième lecture : où se trouve la transition ?

Le second fichier décrit une progression : consulter le catalogue disponible, construire la règle, la valider, présenter son sens à la personne, attendre sa confirmation, puis appeler `create_watch`.

Le point intéressant à relever dans une revue est la transition entre une proposition encore modifiable et un objet enregistré. Pour la rendre visible, on peut résumer chaque étape par une ligne :

```
Entrée : une demande de suivi et une destination explicite
Préparation : une règle construite avec les champs du catalogue
Validation : les erreurs doivent être résolues
Confirmation : la règle et la destination sont relues
Écriture : création de la règle de suivi
```

Ce résumé est une fiche de lecture du fichier, pas une démonstration du comportement du programme en production. Vérifier l'application réelle de ces consignes demanderait aussi d'examiner le client, les outils disponibles et leurs autorisations.

La référence des outils MCP de ClosedLoop AI classe explicitement `create_watch` parmi les opérations qui enregistrent un changement. Croiser cette documentation avec les instructions évite de déduire l'effet d'un outil uniquement de son nom.

Troisième lecture : que manque-t-il pour continuer ?

Une instruction complète doit aussi permettre d'identifier les informations absentes. Dans le fichier de suivi, le nombre, la fenêtre temporelle et la destination ne sont pas des détails décoratifs : ils changent la règle qui serait enregistrée.

Si la demande est seulement « préviens-moi quand cela augmente », plusieurs questions restent ouvertes. Combien d'événements ? Sur quelle durée ? Dans quel canal ? Choisir ces valeurs sans les faire préciser peut

produire une règle valide sur le plan de la structure mais différente de la demande.

Le premier fichier pose un autre problème de périmètre : retrouver la fonctionnalité réellement livrée. Le texte prévoit de privilégier des identifiants précis et de s'arrêter si le périmètre ne correspond pas à une livraison. Là encore, une sortie bien présentée ne remplace pas une entrée correctement définie.

Ce que cette comparaison apporte à une revue

Avant de lancer une compétence, noter son entrée, ses appels d'outils, son dernier effet attendu et son comportement quand une information manque permet de comparer la promesse au déroulé.

Dans ces deux exemples, le même environnement de données sert soit à préparer un texte, soit à enregistrer un mécanisme de suivi. L'intérêt des fichiers Markdown publics est de rendre cette différence discutable avant l'exécution.

Pour vos propres compétences, quelle trace utilisez-vous pour relier une instruction de confirmation à l'appel qui modifie effectivement quelque chose : commentaire dans le fichier, fiche de revue, ou scénario de vérification séparé ?

(1 commentaire). Markdown EPUB

mv slop.md ~/.Trash/

Posté par **steph1978** le 07 novembre 2026 à 20:49.
Évalué à 2 (+0/-0). Dernière modification le 07 novembre 2026 à 20:50.

> voire un petit `shred -n 2 -z -u slop.md`

Répondre

Envoyer un commentaire

Suivre le flux des commentaires

Note : les commentaires appartiennent à celles et ceux qui les ont postés. Nous n'en sommes pas responsables.

Revenir en haut de page

Derniers commentaires

- Re: Oui et... ?
- Re: Max est bien pire
- Re: Rigolo
- Re: Rapport avec Li…
- Re: alternativement...
- Re: Oui et... ?
- Re: Pareil avec mon …
- Re: Max est bien pire
- sources
- Re: Mauvais timing
- mv slop.md ~/.Trash/
- Re: déni de réalité

Étiquettes (tags) populaires

- intelligence_artificielle
- merdification
- slop
- startup_nation
- sécurité
- états-unis
- administration_fran…
- cybersécurité
- bigtech
- linux
- vibe_coding
- cyberattaque

Sites amis

- Agenda du Libre
- April
- Éditions D-BookeR
- Éditions Diamond
- Éditions Eyrolles
- Éditions ENI
- En Vente Libre
- Framasoft
- La Quadrature du Net
- Lea-Linux
- Open Source Initiative
- Imprimerie Grafik Plus

À propos de LinuxFr.org

- Mentions légales
- Faire un don
- L'équipe de LinuxFr.…
- Informations sur le s…
- Aide / Foire aux que…
- Suivi des suggestion…
- Wiki du site
- Règles de modération
- Statistiques
- API pour le dévelop…
- Code source du site
- Plan du site