

ask-questions-if-underspecified

ask-questions-if-underspecified

“ Catalogue généré le 2026-05-11

En une phrase

Force Claude à te poser des questions de clarification avant de se lancer dans le code, quand ta demande peut être interprétée de plusieurs façons — pour éviter de coder pendant deux heures dans la mauvaise direction.

Quand l'utiliser

- Quand tu demandes quelque chose de vague (« améliore l'affichage », « ajoute une page ») et que tu veux que Claude vérifie qu'il a bien compris.
- Avant un gros chantier où une mauvaise compréhension coûterait cher.
- Quand plusieurs interprétations sont possibles et que tu préfères choisir explicitement.
- Quand tu veux établir ensemble les critères de « c'est fini » avant de coder.

Comment l'invoquer

- **Slash command** : `/ask-questions-if-underspecified`.
- **Phrases déclencheurs (texte)** : "clarify requirements", "ask before implementing", "are you sure you understood".
- **Auto-invocation** : Activé automatiquement dès que la demande est ambiguë, ou sur demande explicite.

Description détaillée

C'est un skill méta : il ne fait pas de code, il discipline la manière dont Claude réagit à une demande. L'idée part d'un constat simple : quand un dev (humain ou IA) se lance sans avoir compris le besoin, il finit souvent par livrer quelque chose qui ne correspond pas, qu'il faut tout refaire. Plutôt que de deviner, autant prendre cinq minutes pour s'aligner.

Concrètement, dès que Claude reçoit une demande, il vérifie six choses : l'objectif (qu'est-ce qui doit changer ?), la définition de « terminé » (à quoi je reconnais que c'est fait ?), le périmètre (quels fichiers, quels composants ?), les contraintes (perf, style, dépendances), l'environnement (quelle version, quel OS ?), et la sécurité (faut-il une procédure de rollback ?). Si une de ces dimensions est floue, il s'arrête et te pose 1 à 5 questions courtes, sous forme numérotée avec des options à choisir.

Mieux : il propose toujours une option par défaut (marquée « recommandée ») et accepte une réponse compacte type `defaults` ou `1a 2b 3a`. Tu n'as donc pas besoin de rédiger un cahier des charges, juste de cocher des cases. Une fois les réponses obtenues, Claude reformule ta demande en deux phrases pour confirmer, puis commence. Ça évite la moitié des « ah non, ce n'était pas ce que je voulais ».

Pour aller plus loin

Pour les détails techniques, exemples et patterns spécifiques, voir le SKILL.md original.

Source

- **Plugin** : `trailofbits/ask-questions-if-underspecified`
- **Nom interne** : `ask-questions-if-underspecified`
- **Fichier** : `/home/thymon/.claude/plugins/cache/trailofbits/ask-questions-if-underspecified/1.0.1/skills/ask-questions-if-underspecified/SKILL.md`

Revision #2

Created 2026-05-11 21:19:26 UTC by thymon

Updated 2026-05-11 21:37:03 UTC by thymon