

Le mécanisme qui déclenche vraiment un Skill, le gabarit SKILL.md à copier, et les dix Skills qui font gagner le plus de temps dans une TPE/PME.

BLYX-AGENCY.COM · 13 AOÛT 2026

LE GUIDE DES SKILLS CLAUDE — par Pierre Cordelier.

Écrire une fois ce que Claude doit savoir de votre entreprise, et ne plus jamais le réexpliquer. Ce guide couvre le mécanisme réel qui déclenche un Skill — celui que la plupart des tutoriels passent sous silence —, le gabarit à copier, et les dix Skills qui font gagner le plus de temps dans une TPE/PME.

Comptez quinze minutes pour le premier. Le reste du guide sert à ne pas se tromper sur les suivants.

1. Le problème que ça règle

Vous utilisez Claude tous les jours. Et tous les jours, vous réexpliquez la même chose : votre métier, votre cible, votre style, la structure que vous voulez, ce qu'il ne faut surtout pas écrire.

Chaque nouvelle conversation repart de zéro. Vous payez le contexte à chaque fois.

C'est là que se joue la différence entre les deux façons d'utiliser Claude :



Déclenchement	Vous le collez à la main	Claude le charge tout seul
Coût	Retapé à chaque fois	Écrit une fois
Ce que ça devient	Une dépense quotidienne	Un actif de l'entreprise

Un prompt se paie tous les jours. Un Skill se paie une fois.

C'est la même différence qu'entre refaire son café à la main chaque matin et programmer la machine la veille au soir. Le geste est identique, le coût cumulé n'a rien à voir.

2. Le mécanisme, en une page

Un Skill, c'est un dossier avec un fichier `SKILL.md` dedans. Ce fichier porte deux choses : une **description** — quand s'en servir — et des **instructions** — quoi faire.

Voilà le point que presque tout le monde rate, et qui change la façon de les écrire :

Un Skill ne se déclenche pas par une commande. Il n'y a rien à taper, aucun mot magique. Claude a en permanence sous les yeux la description de tous vos Skills. Dès que votre demande correspond à l'une d'elles, il ouvre ce Skill-là et applique ce qu'il contient. Vous écrivez « prépare-moi le devis pour la société Martin », et votre Skill de propositions commerciales s'ouvre sans que vous l'ayez demandé.

Deux conséquences, et elles décident de tout le reste du guide.



La description fait le travail le plus important. Un Skill

B Blyx re...ment écrit avec une description floue ne se déclenchera jamais. Il existera dans votre compte sans jamais servir. C'est la partie sur laquelle il faut passer le plus de temps — et c'est celle que tout le monde bâcle en trente secondes.

Le corps, lui, ne coûte rien tant qu'il ne sert pas. Claude ne lit les instructions qu'au moment où il en a besoin. Vous pouvez donc être long, précis, maniaque dans le corps du fichier : ça n'encombre aucune de vos autres conversations. C'est exactement l'inverse d'un prompt géant collé partout par précaution.

3. L'anatomie d'un fichier SKILL.md

Un fichier texte, deux parties. L'en-tête entre deux traits, puis vos instructions.

```
---
name: proposition-commerciale
description: >
    Utiliser quand je prépare un devis ou une proposition pour un
    prospect.
    Se déclenche sur « fais-moi une propale pour », « prépare le
    devis de »,
    « chiffre cette demande ». Ne pas utiliser pour une facture ni
    pour une
    relance d'impayé.
---

# Proposition commerciale

## Ce que je fais
...
```



Trois choses à savoir sur l'en-tête, et elles suffisent :



✓ **name** s'écrit en minuscules avec des tirets, sans accent et sans espace.

✓ **description** dit **quand** intervenir, pas seulement quoi faire. C'est la seule partie que Claude lit en permanence.

✓ Tout ce qui suit le second trait est le corps : il ne s'ouvre que lorsque la description a fait mouche.

4. Le gabarit à copier

Copiez ce fichier, remplacez les crochets par votre contexte, enregistrez-le sous le nom **SKILL.md** . C'est tout ce qu'il y a à faire.

```
----  
name: nom-en-minuscules-avec-des-tirets  
description: >  
    Utiliser quand [la situation exacte, décrite comme vous la vivez].  
    Se déclenche sur « [une phrase que vous tapez vraiment] »,  
    « [une deuxième] », « [une troisième] ».  
    Ne pas utiliser pour [le cas voisin qui ressemble mais qui n'est pas celui-là].  
----
```

```
# [Nom lisible du Skill]
```

```
## Ce que je fais
```

```
[Une phrase. Le résultat qui doit exister à la fin et qui n'existait pas avant.]
```

```
## Ce que je dois savoir
```

```
- Entreprise : [nom, secteur, taille]
```



- Cible : [à qui on s'adresse]



Ce qu'on vend vraiment : [le résultat pour le client, pas le produit]



- Ton : [deux ou trois adjectifs, et un contre-exemple]

Comment je procède

1. [première étape]
2. [deuxième étape]
3. [troisième étape]

Les règles

- Ne jamais [ce que vous ne voulez voir sous aucun prétexte].
- Ne jamais employer [le mot ou la formule que vous détestez].
- En cas de doute sur [le cas ambigu qui revient], faire [le comportement attendu] plutôt que de deviner.
- Si une information manque, la demander. Ne jamais l'inventer.

Exemples

[Collez ici un ou deux de vos meilleurs résultats réels, en entier.]

La partie qui compte le plus est la dernière. Deux exemples réels apprennent votre style infiniment mieux que dix lignes de description abstraite. Montrez, ne décrivez pas. Si vous ne devez soigner que deux blocs dans tout le fichier, prenez la description et les exemples.

La règle « en cas de doute » n'est pas décorative non plus. Un Skill qui tourne sans qu'on relise tout a tendance à meubler quand il lui manque quelque chose. Lui dire quoi faire du vide évite de découvrir trois semaines plus tard un devis avec un tarif inventé.

5. Trois façons de créer votre premier Skill



Elles marchent toutes les trois. Elles ne coûtent pas le même temps.



Méthode 1 — le gabarit ci-dessus. Vous remplissez les crochets, vous enregistrez. Quinze minutes, et vous comprenez ce que vous avez écrit — ce qui compte pour le corriger ensuite.

Méthode 2 — faire écrire le Skill par Claude. Ouvrez une conversation et demandez : « Je veux créer un Skill qui [ce que vous voulez automatiser]. Pose-moi les questions nécessaires, puis génère le fichier SKILL.md complet. » Répondez à ses questions, récupérez le fichier. Cinq minutes.

Méthode 3 — recycler une conversation qui a marché. C'est la meilleure des trois, et personne ne la connaît. Prenez un résultat que Claude vous a déjà produit et que vous avez trouvé excellent. Dans la même conversation, écrivez : « Transforme cet échange en Skill réutilisable. Mets dans la description les phrases exactes qui devraient le déclencher, et garde ce résultat comme exemple. »

Vous capturez ce qui a marché au lieu de le décrire de mémoire. Deux minutes, et c'est le Skill le plus juste que vous écrirez.

Le bon réflexe, ensuite : chaque fois qu'un résultat vous plaît vraiment, vous le transformez en Skill. Votre bibliothèque se construit toute seule, à partir de vos propres réussites.

6. Installer et tester

Ajoutez le fichier à Claude depuis les réglages de votre application, section des Skills. Puis testez-le tout de suite : lancez une demande devrait le déclencher, formulée comme vous la formuleriez vraiment mardi matin.



Deux issues possibles.



Il se charge. Regardez ce qu'il produit et corrigez le corps du fichier.



Deux ou trois passes suffisent presque toujours.

Il ne se charge pas. Le problème est presque toujours dans la description. La section suivante est faite pour ça.

7. Votre Skill ne se déclenche pas — les quatre causes

1. La description dit quoi, pas quand. « Rédige des propositions commerciales percutantes » décrit un talent. Claude cherche une situation. Écrivez plutôt : « Utiliser quand je prépare un devis ou une proposition pour un prospect. »

2. Vos phrases déclencheurs ne sont pas les vraies. Vous avez écrit « générer une proposition commerciale » et vous tapez « fais-moi la propale pour Martin ». Mettez vos mots à vous, ceux que vous employez réellement, y compris le jargon de votre métier et les abréviations.

3. Le Skill essaie de tout faire. Un fichier qui couvre les devis, les factures et les relances ne se déclenche jamais au bon moment, parce qu'aucune situation ne lui correspond franchement. **Un Skill, un métier.** Découpez.

4. Deux Skills se marchent dessus. Quand deux descriptions se ressemblent, vous ne savez jamais lequel part. Ajoutez à chacune une ligne « Ne pas utiliser pour... » qui nomme l'autre. C'est cette ligne-là qui tranche.



8. Les dix Skills à créer dans une TPE/PME



Commencez par celui qui vous fait perdre le plus de temps chaque semaine. Pas par le plus impressionnant.



#	Le Skill	Ce qu'il fait tourner tout seul
1	Contexte entreprise	Qui vous êtes, ce que vous vendez, votre cible, vos chiffres — la base de tous les autres
2	Votre voix à l'écrit	Publications, messages, courriers : votre style exact, pas celui de Claude
3	Devis et propositions	Votre trame commerciale remplie en trente secondes
4	Réponse au client mécontent	Votre ton, vos limites, votre politique de geste commercial
5	Compte rendu de réunion	Votre format : décisions, actions, qui fait quoi, pour quand
6	Accueil d'un nouveau client	La séquence complète, du contrat signé au premier point
7	Fiche produit ou prestation	Des descriptions cohérentes avec votre positionnement
8	Analyse d'un concurrent	Votre grille de lecture, appliquée à n'importe quel nom
9	Cadrage d'une demande entrante	Le brief que vous remplissez à chaque nouveau projet
10	Relance commerciale	Votre rythme, votre ton, et ce que vous ne faites jamais



Celui par lequel commencer : le numéro 1. Un simple fichier qui dit

 Blyx, ce que vous vendez, à qui, et avec quels chiffres. Il ne  produit rien tout seul — et il rend meilleurs tous les autres, parce que Claude cesse enfin de découvrir votre entreprise à chaque conversation.

9. La règle apprise en cassant mon propre système

Le système qui produit mes publications LinkedIn tourne sur quinze Skills. Un qui choisit le sujet, un qui cadre l'angle, un qui vérifie les faits, un qui écrit, un qui relit sous six angles différents, un qui met en ligne. Cinq nuits par semaine, la chaîne tourne pendant que je dors et je lis le résultat au réveil.

Un jour, j'ai fait le compte des défauts accumulés : treize. **Onze d'entre eux étaient la même erreur** — une règle écrite à deux endroits, et les deux versions avaient fini par diverger. Le Skill qui lisait le premier fichier travaillait avec une règle, celui qui lisait le second avec une autre. Aucun des deux n'avait tort de son point de vue.

D'où la seule règle d'architecture que je donnerais à quelqu'un qui construit sa bibliothèque : **une règle, un seul propriétaire**. Avant d'écrire une consigne dans un Skill, cherchez si elle vit déjà dans un autre. Si oui, renvoyez-y au lieu de la recopier.

C'est contre-intuitif — recopier semble plus sûr, on a tout sous la main. C'est exactement l'inverse : deux copies finissent toujours par se contredire, et personne ne s'en aperçoit avant que le résultat soit faux.



10. Ce que ça donne au bout de quelques



Chaque Skill que vous écrivez rend Claude un peu plus proche de votre façon de travailler. Au bout de cinq ou six, quelque chose change de nature : vous n'avez plus un assistant générique à qui tout expliquer, vous avez un collaborateur qui connaît votre entreprise et exécute à votre niveau d'exigence.

C'est là que le temps gagné devient visible sur une semaine, pas sur une tâche.

Et c'est là que se pose la vraie question, celle qui n'est pas technique : **quoi automatiser en premier**. Dix Skills créés le même week-end, ce sont dix fichiers mal écrits qu'on abandonne le mois suivant. Un seul Skill, sur la tâche qui coûte le plus cher chaque semaine, se garde des années.

Pour aller plus loin

Vous avez le mécanisme, le gabarit, les trois méthodes et les dix idées. De quoi installer votre premier Skill cet après-midi.

Si vous voulez qu'on regarde votre cas précis — quelles tâches de votre entreprise méritent d'être écrites une fois pour toutes, et lesquelles ne valent pas le détour —, c'est exactement ce que fait notre audit IA : on cartographie le temps réellement passé, et on classe les automatisations par ce qu'elles rapportent.



[Réserver un appel](#) — 30 minutes, offert, sans engagement.

Une question sur votre propre cas ? Écrivez-moi, je regarde.



