

Le kit de 7 Claude Skills pour faire tourner une agence

Sept Claude Skills à copier et un guide d'entrée pour traiter chaque client de la même façon : playbook, proposition avec exclusions, onboarding, vérification du périmètre, point hebdo.

BLYX-AGENCY.COM · 1 NOVEMBRE 2026 · PAR PIERRE CORDELIER

Dans une agence, ce qui se perd, c'est ce qui a été promis pendant l'appel de vente. Le contrat dit une chose, le client en a entendu une autre, et l'équipe de suivi l'apprend trois semaines après le lancement.

Ce kit traite chaque client de la même façon, de la première proposition au point de la semaine. Il tient en sept Claude Skills : des fiches d'instructions que Claude charge quand votre demande y correspond. Un guide d'entrée les lance dans l'ordre.

Le but n'est pas de remplacer votre équipe de suivi client. C'est de repérer les dérives de périmètre avant qu'elles commencent, de réclamer les accès manquants en un seul message, et de faire démarrer chaque nouveau client avec ce que le précédent vous a appris. Que vous dirigiez une agence de publicité payante, de SEO, de web ou d'automatisation, le principe est le même.

Ce que contient le kit



Le guide d'entrée : dit quelle skill lancer et dans quel ordre

config-agence

Écrit le playbook de votre agence à partir de six questions

proposition

Rédige une proposition avec le périmètre et une liste « non inclus »

espace-client

Crée la même organisation de dossiers pour chaque client

**onboarding-
client**

Brief, accès, audit de départ, kit de lancement

verif-perimetre

Répond à « est-ce que c'est dans le périmètre ? » contre le contrat

point-hebdo

Écrit le point client de la semaine, dans le ton de l'agence

client-test

Un client fictif avec 8 problèmes cachés pour tester le tout

Le guide d'entrée de ce kit est une skill ordinaire, que vous lancez depuis une conversation Claude. Ce n'est pas une fonction de Claude : c'est une consigne d'ordre de marche, donnée plus bas.

Installer le kit en dix minutes

1. Créez un dossier par skill, du nom exact de la skill (par exemple **verif-perimetre**). Dans chacun, un fichier **SKILL.md** : le nom, la description, puis les instructions.



2. Le nom du dossier et le nom de la skill sont identiques, en

Blyx n'ajoute pas de espaces et tirets. La description fait moins de 1 024 caractères  commence par le cas d'usage principal : c'est la seule partie que Claude lit pour décider de charger la skill.

3. Dans l'application Claude, compressez chaque dossier en ZIP (le dossier à la racine de l'archive), puis ouvrez Customize > Skills, « + », « Upload a skill ». Il faut avoir activé la création et la modification de fichiers dans les réglages. Les noms de menus évoluent vite : vérifiez-les le jour de l'installation.

4. Dans Claude Code, copiez les dossiers dans `~/ .claude/skills/` pour vous, ou dans `.claude/skills/` du projet pour l'équipe.

5. Testez avec une demande qui doit déclencher la skill. Si rien ne se passe, c'est presque toujours la description qu'il faut préciser.

Plus simple : donnez à Claude le texte de chaque skill ci-dessous et demandez-lui de créer le dossier et le fichier prêts à importer.

Deux règles pour tout le kit. Aucune clé, aucun mot de passe dans les fichiers. Aucune donnée d'un vrai client : les exemples restent fictifs.

Le guide d'entrée : agency-guide

name: agency-guide

description: Point d'entrée de l'agence. À utiliser quand je dis « nouveau client », « nouvelle proposition », « on est dans le périmètre ? » ou « point de la semaine » : indique la skill à lancer et ce qu'il faut me fournir avant.

Instructions :

1. Demander où j'en suis : agence pas encore configurée, prospect, client signé, client en cours.

2. Agence pas configurée : lancer config-agence. Prospect : proposition. Client signé : espace-client puis onboarding-client.



Client en cours : verif-perimetre ou point-hebdo selon ma demande.



Blyx



3. Avant de lancer une skill, lister en trois lignes ce qu'elle exige de ma part (contrat, notes d'appel, accès).
4. Ne rien inventer : si un document manque, le demander.

1. config-agence : votre playbook en six questions

Tout le reste s'appuie sur ce texte. Faites-le une fois.

```
name: config-agence
description: À utiliser pour écrire ou mettre à jour le playbook de
l'agence : services, prix, ton, règles de travail. Lance une
interview de six questions.
Instructions :
Poser ces six questions, une à la fois, en attendant la réponse :
1. Quels services vendez-vous, et lesquels refusez-vous ?
2. Comment fixez-vous vos prix (forfait, abonnement, régie) ?
3. Que contient toujours une proposition, et que contient jamais un
forfait ?
4. Comment écrivez-vous à vos clients (ton, formules, mots
interdits) ?
5. Quels accès demandez-vous toujours au démarrage ?
6. Quelle règle a déjà coûté cher à l'agence quand elle a été
oubliée ?
Rédiger le playbook en une page, à la première personne, dans les
mots de l'agence. Le montrer, attendre mon accord, puis
l'enregistrer dans playbook.md.
Règle : ne rien inventer, poser la question quand une réponse
manque.
```

2. proposition : le périmètre, et ce qui n'en fait pas partie



name: proposition
description: À utiliser pour rédiger une proposition commerciale à partir des notes d'un appel de vente, avec périmètre, livrables, exclusions et conditions de changement.
Instructions :
1. Lire playbook.md et mes notes d'appel.
2. Structure : contexte du client, objectif, livrables un par un, calendrier, prix.
3. Section « Non inclus » obligatoire. Y mettre tout ce qui a été évoqué à l'appel sans être vendu, plus les exclusions habituelles du playbook.
4. Fixer le nombre de tours de révision inclus, le délai de retour du client et le tarif d'un tour supplémentaire.
5. Ajouter la règle de changement : toute demande hors périmètre est chiffrée, puis démarrée seulement après validation écrite.
6. Terminer par la liste des points de l'appel qui ne figurent pas dans la proposition, pour que je tranche.
Règle : aucune date, aucun prix, aucun engagement absent de mes notes ou du playbook.

Faites relire les clauses d'exclusion et de changement par un juriste avant de les mettre dans un modèle de contrat.

3. espace-client : la même organisation à chaque fois

name: espace-client
description: À utiliser quand un client signe : crée l'arborescence de dossiers et les fichiers de départ de son espace de travail.
Instructions :
1. Demander le nom du client et la date de signature.
2. Créer le dossier du client avec : 01-contrat, 02-brief, 03-



accès, 04-audit, 05-livrables, 06-points-hebdo, 07-journal-des-changements.



Blyx

3. Pour chaque dossier, un fichier LISEZMOI de deux lignes : ce qui y va.

4. Créer le journal des changements, un tableau : date, demande, auteur, statut, impact, lien vers la validation.

5. Y déposer le contrat et la proposition quand je les fournis.



4. onboarding-client : brief, accès, audit, lancement

Quatre étapes, une seule skill. Les accès sont l'étape la plus souvent sous-estimée : un client qui ne sait pas quel rôle donner transmet la demande, et elle revient avec le mauvais niveau, Lecteur au lieu d'Éditeur. D'où un seul message, précis, envoyé le premier jour.

name: onboarding-client

description: À utiliser pour démarrer un client signé : brief de départ, demande d'accès, audit de compte, kit de lancement.

Instructions :

Étape 1, brief. Lire le contrat et la proposition. Poser les questions manquantes : objectifs, cibles, concurrents, contacts, personne qui valide.

Étape 2, accès. Écrire un seul message au client avec, pour chaque plateforme, l'accès demandé, le rôle exact à donner et la marche à suivre. Chaque ligne a un responsable côté client et une échéance. Lister aussi les éléments manquants (logos, mots de passe de comptes publicitaires par le canal prévu, textes).

Étape 3, audit. Analyser les données fournies. Chaque chiffre montre son calcul (source, période, opération). Si une donnée manque, l'écrire au lieu de l'estimer.

Étape 4, lancement. Préparer le kit : plan des 30 premiers jours, calendrier, message de lancement au client. Me le soumettre : rien ne part avant mon accord.

Règle : ne jamais inventer un chiffre, une date ou un engagement



5. **verif-perimetre** : « est-ce que c'est dans le périmètre ? »



Cette skill ne lit que ce que vous lui donnez. Le contrat doit donc être joint à chaque vérification, ou présent dans le dossier de l'espace client.

```
name: verif-perimetre
description: À utiliser quand un client demande quelque chose et que je me demande si c'est inclus. Compare la demande au contrat et à la proposition.
Instructions :
1. Lire le contrat et la proposition du client. S'ils ne sont pas fournis, les demander et s'arrêter.
2. Classer la demande : dedans, dehors, ou à clarifier.
3. Citer la clause exacte qui justifie le classement. Signaler aussi ce qui a été promis à l'appel (mes notes) sans figurer au contrat.
4. Si c'est dehors : proposer un arbitrage (budget, délai ou priorité) et rédiger la réponse au client.
5. Le format de la suite : un avenant si la demande modifie le périmètre, le prix, le délai ou les livrables principaux ; un nouveau devis si elle peut vivre seule ; une ligne de facture si le tarif est déjà prévu au contrat, avec un mail de validation.
6. Ajouter la ligne au journal des changements.
Règle : si le contrat ne tranche pas, répondre « à clarifier », jamais « dedans » par défaut.
```

6. **point-hebdo** : le point client, dans votre ton

```
name: point-hebdo
description: À utiliser chaque semaine pour écrire le point client : ce qui a été fait, les résultats, ce qui bloque, la suite.
Instructions :
1. Lire playbook.md pour le ton, le journal des changements et les
```



notes de la semaine que je fournis.



2. Écrire le point en quatre blocs : fait, résultats, blocages (avec ce qu'on attend du client), prochaine semaine.



3. Mentionner toute demande hors périmètre en cours, avec son statut.

4. Chaque chiffre vient de mes données, avec sa période.

5. Me le montrer : rien ne part sans mon accord.

Règle : si une donnée manque, l'écrire en tête du point.

7. client-test : un client d'entraînement avec 8 problèmes cachés

Avant de lancer un vrai client, vérifiez que vos skills trouvent ce qu'elles doivent trouver. Créez un client fictif, ici Northwind Labs, avec un contrat, des notes d'appel et un compte, dans lesquels vous cachez huit pièges. Voici une liste à reprendre.

Piège caché	La skill qui doit le trouver
L'appel a promis des révisions illimitées, le contrat en fixe deux	verif-perimetre
Une landing page évoquée à l'appel n'est dans aucun document	proposition
La rédaction des contenus n'est exclue nulle part	proposition
L'accès au compte publicitaire est en Lecteur, il faut Éditeur	onboarding-client
Le budget du brief diffère de celui du contrat	onboarding-client
Deux sources comptent la même conversion en double	onboarding-client



Personne n'est nommé pour valider les livrables

espace-client

Une demande de la semaine 3 est hors périmètre mais
n'a pas de ligne au journal

point-hebdo

name: client-test

description: À utiliser pour tester le kit sur un client fictif, Northwind Labs, avant un vrai client. Joue le rôle du client et vérifie que les skills trouvent les problèmes cachés.

Instructions :

1. Fournir le contrat, les notes d'appel et les données fictives de Northwind Labs, avec les huit problèmes de la liste.
2. Ne jamais révéler la liste pendant le test.
3. À la fin, comparer les problèmes trouvés par les skills à la liste et donner le score sur huit, avec les oublis.

Règle : aucune donnée réelle dans ces fichiers.

Faire apprendre le playbook de vos corrections

Chaque fois que vous corrigez Claude, une règle manque quelque part. Terminez la conversation par ce prompt.

Je viens de te corriger sur : [ce que j'ai changé]. Dis-moi dans quelle skill cette règle doit aller, écris la ligne à ajouter en une phrase, et attends mon accord avant de modifier le fichier.

Le kit s'améliore ainsi client après client. Le client suivant démarre avec ce que le précédent a appris.



Ce que ce kit ne fait pas



✓ Il ne voit pas vos outils (CRM, messagerie, Drive) sans connecteur.

Vous lui fournissez les documents.

- ✓ Il ne remplace pas la relecture. Une skill peut ajouter un détail absent de vos notes : la règle « ne rien inventer » est dans chaque fichier, vérifiez quand même.
- ✓ Les skills personnelles ne sont pas visibles par vos collègues. Sur les offres Team et Enterprise, elles se partagent depuis Customize > Skills ; sinon, remettez le ZIP à chacun avec la marche d'import.
- ✓ Une skill envoyée à quelqu'un lui transmet tous ses fichiers. N'installez que des skills dont vous avez lu le contenu.
- ✓ Avec un pouce haut ou bas donné dans Claude, la conversation entière peut être conservée jusqu'à cinq ans. Pour un client sensible, prévenez l'équipe.

Savoir quelles tâches de votre agence méritent une skill, et laquelle fait gagner le plus de temps, c'est le travail de fond de notre audit IA : cartographier, mesurer, chiffrer.

Source. Format, chargement et installation des skills : [la spécification Agent Skills](#), [Anthropic, Equipping agents with Agent Skills](#), [le Centre d'aide Claude](#), [la documentation Create custom skills](#) et [la documentation Claude Code](#). Données : [Anthropic, Is my data used for training](#). Périmètre et exclusions : [Lefreelance](#). Accès : [Chaser](#). Le contenu des sept skills et le client d'entraînement sont nos propres choix de conception.



