

GUIDE 10

Reconnaître un vrai cas d'usage

Niveau 3 — distinguer une idée intéressante, une démonstration impressionnante, et un cas qui mérite réellement un pilote.



Emplacement réservé à l'infographie de la collection (format portrait). Le schéma ci-dessus tient lieu de repère de mise en page.

Partir du travail réel, pas de l'IA

Un bon cas d'usage ne commence pas par «qu'est-ce qu'on pourrait faire avec Claude ?». Il commence par : **quel travail réel voulons-nous améliorer, comment fonctionne-t-il aujourd'hui, et comment saurons-nous que c'est mieux ?**

Une technologie n'est pas un cas d'usage. Une tâche améliorable en est un candidat — et la différence se voit à ce qu'on peut en dire avant même d'ouvrir l'outil.

Les signaux d'une tâche candidate

- 1 Fréquente et lente**
Elle revient, et elle prend du temps à chaque fois.
- 2 Riche en lecture ou en rédaction**
Synthétiser, reformuler, structurer, comparer.
- 3 Variable mais reconnaissable**
Chaque cas diffère, mais le résultat attendu se décrit.
- 4 Source d'erreurs ou de reprises**
On la refait souvent parce qu'elle a été mal faite.

Idee vague. « Utiliser Claude dans les réunions. »

Cas d'usage. « Après chaque réunion client, transformer les notes et la transcription en compte rendu au format interne, avec décisions, responsables, échéances et points à vérifier. »

Le second a une entrée, une sortie, une fréquence, une baseline possible — et un test limité possible.

À RETENIR

Un cas d'usage est un problème de travail avant d'être une solution IA.

VERSION WEB PERMANENTE

consulting.amplify-belgium.com/ressources/reconnaitre-un-vrai-cas-dusage

GUIDE 10 · COLLECTION KNOWLEDGE CLAUDE

Vérifié le 28 août 2026 · 6 pages

GUIDE 10 · 01

Valeur, faisabilité, risque : trois conditions, pas une

Un bon candidat réunit les trois. Un candidat qui n'en réunit que deux n'est pas forcément mauvais — mais il faut savoir laquelle manque, parce que ce n'est pas la même suite.

CONDITION	LA QUESTION À POSER	CE QU'ON REGARDE
1. Valeur	Si cela fonctionne, qu'est-ce qui s'améliore réellement ?	Temps, reprises, erreurs, délai, qualité, capacité, satisfaction des personnes qui font le travail. Une amélioration qu'on ne sait pas nommer ne se mesurera pas non plus.
2. Faisabilité	Claude dispose-t-il des bonnes entrées, et le résultat est-il définissable ?	Données et documents disponibles, tâche compréhensible, volume raisonnable, critères de réussite formulables, intégrations nécessaires.
3. Risque	Que se passe-t-il quand le résultat est mauvais ?	Impact humain, impact client, confidentialité, coût, réversibilité, niveau de validation requis — voir les Guides 05 et 08.

La formule de priorisation

Priorité $\approx$ valeur élevée + faisabilité suffisante + risque maîtrisable.

Ce n'est pas un score scientifique. C'est un moyen de ne pas sélectionner uniquement les idées impressionnantes — celles qui font une belle démonstration et un mauvais processus.

Le NIST structure la gestion du risque IA autour de quatre fonctions — *Govern, Map, Measure, Manage* — dont deux concernent directement cette page : cadrer le contexte d'usage, puis définir comment on mesurera.

[1][2]

Quelle condition manque ?

Valeur faible — même parfaitement exécuté, le gain reste marginal. À écarter, sans regret.

Faisabilité faible — souvent une question de matière : les documents n'existent pas, ou le résultat attendu n'est pas descriptible. À reporter, pas à abandonner.

Risque élevé — le cas peut rester valable, à condition de concevoir le contrôle humain en même temps que le processus, pas après.

À RETENIR

Savoir laquelle des trois conditions manque vaut mieux qu'un score global : c'est elle qui dit s'il faut écarter, reporter ou encadrer.

GUIDE 10 · 02

Concevoir un pilote qui apprend vraiment quelque chose

Tester petit ne veut pas dire faire une démo. Une démo montre que Claude *peut* produire quelque chose. Un pilote teste si la nouvelle façon de travailler produit un meilleur résultat, dans des conditions proches du réel.

- 1 Établir la baseline avant de commencer**
Temps moyen actuel, nombre d'erreurs à corriger, délai, satisfaction des personnes concernées. Sans point de départ mesuré, aucune amélioration ne sera démontrable — seulement ressentie.
- 2 Écrire l'hypothèse**
« Avec Claude et la méthode X, nous voulons réduire le temps de 30 % sans augmenter les erreurs critiques. » Une hypothèse se formule avant, avec un chiffre, et engage.
- 3 Réduire le périmètre**
Une équipe, un type de dossier, une vingtaine de cas, deux semaines. Assez petit pour apprendre vite, assez réel pour mesurer.
- 4 Définir le contrôle humain**
Que vérifie-t-on, à quel moment, et qui peut arrêter le test ? Cette question se règle avant le premier cas, pas au premier incident (Guide 08).
- 5 Comparer**
Avant et après, ou avec un groupe témoin lorsque cela a du sens. La comparaison est ce qui distingue un pilote d'une impression collective.

UNE DÉMO

On choisit un cas favorable, on montre le résultat, tout le monde est impressionné. On n'apprend rien sur ce qui se passera le mardi matin, sur le dossier incomplet, avec la personne qui n'a pas assisté à la démonstration.

UN PILOTE

On prend les cas tels qu'ils arrivent, on mesure ce qui était annoncé, et on accepte que le résultat puisse être décevant. C'est cette possibilité qui rend le résultat informatif.

Le critère qui rend un pilote honnête. Un pilote utile est construit pour pouvoir conclure « non ». Si le protocole ne peut produire qu'une conclusion positive, ce n'est pas un test : c'est une démonstration avec un vocabulaire de test.

À RETENIR

Tester sur un périmètre assez petit pour apprendre, assez réel pour mesurer — et décidé avant de commencer.

GUIDE 10 · 03

Mesurer autre chose que « c'était plus rapide »

Le temps gagné à la génération peut être entièrement annulé par le temps de vérification. C'est l'erreur de mesure la plus fréquente — et la plus coûteuse, parce qu'elle valide de mauvais déploiements.

MESURE	CE QU'ON REGARDE	POURQUOI ELLE COMPTE
1. Productivité	Temps brut, débit, coût.	La mesure évidente — et la seule que l'on prend souvent.
2. Qualité	Erreurs, corrections, respect du format, complétude.	Un gain de temps payé en qualité n'est pas un gain.
3. Reprise	Temps nécessaire pour réparer le résultat produit.	C'est la mesure qui manque presque toujours. Génération 2 minutes, correction 18 minutes, ancien processus 15 minutes : la génération était rapide, le processus ne l'est pas.
4. Adoption	Les personnes choisissent-elles la méthode ? l'utilisent-elles correctement ? l'abandonnent-elles ?	Une méthode que personne ne suit a un gain théorique de 100 % et un gain réel nul.
5. Risque	Incidents, erreurs critiques, données non autorisées, actions incorrectes.	Un pilote qui améliore le temps et dégrade le risque doit être vu comme un échec, pas comme un compromis.
6. Variabilité	Cas simple, cas moyen, cas incomplet, cas atypique.	Un bon résultat sur le cas idéal ne dit rien du déploiement. C'est sur les cas incomplets que se joue la différence.

Le calcul honnête

Le temps d'un processus assisté n'est pas le temps de génération. C'est **génération + relecture + correction + décision**. Comparer la première ligne à l'ancien processus complet produit mécaniquement un résultat flatteur et faux.

Mesurer dans la durée

Les premières semaines sont atypiques : effet de nouveauté, attention accrue, cas choisis. Une mesure prise uniquement au démarrage surestime le gain. Reprendre la mesure quelques semaines plus tard coûte peu et change souvent la conclusion.

À RETENIR

Six mesures, dont une seule est évidente. Le temps de reprise et l'adoption disent plus sur un déploiement que le temps de génération.

GUIDE 10 · 04

Savoir dire go, adapter... ou arrêter

La maturité d'une organisation en matière d'IA se voit autant dans les projets qu'elle décide de ne pas déployer que dans ceux qu'elle lance.

DÉCISION	CE QUE MONTRE LE PILOTE	CE QU'ON FAIT ENSUITE
Go	Gain tangible, qualité acceptable, risque maîtrisable, adoption réaliste, processus stable.	Industrialiser la méthode : propriétaire, version, tests, date de vérification (Guide 09).
Adapter	Le potentiel est réel, mais la méthode est instable, les sources insuffisantes, le contrôle humain trop lourd ou le périmètre trop large.	Réduire le cas d'usage — un type de dossier, une étape — et retester. C'est souvent la bonne décision.
Stop	Pas de gain réel, relecture plus coûteuse que le processus initial, trop de cas limites, risque disproportionné, données indisponibles, adoption faible.	Arrêter et documenter pourquoi. Un « non » motivé est un actif : il évite qu'on repopose le même cas dans six mois.

Le principe de sobriété

Anthropic recommande, pour construire avec des modèles de langage, de chercher **la solution la plus simple possible** et de n'augmenter la complexité que lorsque c'est nécessaire — en n'ajoutant de la complexité que lorsqu'elle améliore réellement le résultat. [3]

Traduction opérationnelle : un workflow simple peut être préférable à un système autonome. Et parfois, le processus sans IA reste le meilleur — ce qui est une conclusion valable, pas un échec de la démarche.

Application Satelliteam

Les entretiens de la mission produisent des **candidats**. Le rôle d'Amplify est ensuite de trier entre quatre destinations : quick win, pilote à encadrer, idée à reporter, idée à ne pas poursuivre.

Cette distinction est le principal livrable de la phase d'analyse : une liste d'idées sans tri ne vaut pas grand-chose ; une liste triée avec des motifs explicites oriente les mois suivants.

Ce que ce guide ne fait pas. Il ne fournit pas un catalogue d'idées IA, ni une méthode de sélection universelle. Il donne des critères pour qualifier, tester et conclure — y compris négativement.

À RETENIR

L'objectif n'est pas de trouver le plus d'usages IA possible. C'est de trouver ceux qui améliorent réellement le travail.

GUIDE 10 · 05

Sources, périmètre et mise à jour

Sources primaires consultées le 28 août 2026.

Trois idées à emporter

1. Commencer par la tâche actuelle et sa baseline.
2. Tester sur un périmètre assez petit pour apprendre, assez réel pour mesurer.
3. Prévoir dès le début les critères qui permettent d'arrêter.

Ce que ce guide ne couvre pas

Ni la manière d'industrialiser une méthode retenue (Guide 09), ni le niveau de contrôle humain à appliquer (Guide 08), ni les données mobilisables (Guide 05).

À revérifier : les priorités métier de Satelliteam, la disponibilité des données nécessaires aux cas retenus, et l'évolution des cadres de référence cités.

Sources primaires

[1] AI Risk Management Framework

NIST — cadre de gestion des risques liés à l'IA (AI RMF 1.0)
nist.gov/itl/ai-risk-management-framework

[2] AI RMF Core et Playbook

NIST AI Resource Center — fonctions Govern, Map, Measure, Manage
airc.nist.gov/airmf-resources/airmf/ · airc.nist.gov/airmf-resources/playbook/

[3] Building Effective AI Agents

Anthropic Engineering — chercher la solution la plus simple possible
anthropic.com/engineering/building-effective-agents

[4] AI Risk Management Framework: Generative AI Profile (NIST AI 600-1)

NIST, juillet 2024 — risques propres aux systèmes génératifs
nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf

Point de vigilance sur les sources. Le NIST indique que l'AI RMF 1.0 est en cours de révision, dans le cadre du plan d'action américain sur l'IA ; aucun calendrier de publication n'est annoncé à la date de vérification de ce guide. Le cadre publié reste une référence utilisable ; les versions suivantes devront être intégrées lors d'une prochaine mise à jour de la collection.