

MÉTHODOLOGIE · KÉVIN SILVE

PRODUCT PARTNER POUR SAAS B2B

Comment je fais avancer ton **produit.**

Ma méthodologie en 5 phases, pensée pour les founders qui veulent un produit livré, compris et utilisé.

Kévin Silve · nealabs.studio · kevin.silve@outlook.com

Pourquoi cette méthodologie.

La plupart des startups SaaS B2B ne souffrent pas d'un problème de tech. Elles souffrent d'un problème de pilotage produit.

Les founders ont la vision. Les devs ont les mains. Entre les deux, il manque souvent quelqu'un pour :

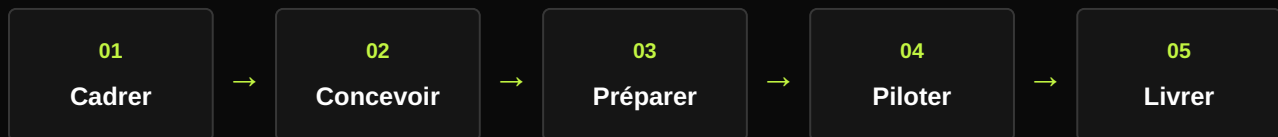
- Transformer une intention business en décisions produit claires
- Traduire ces décisions en specs exploitables
- Coordonner l'exécution sans devenir un goulot
- S'assurer que ce qui est livré résout vraiment le besoin

C'est ce que je porte en tant que **Product Partner embarqué**. Ce document décrit comment je travaille, étape par étape, pour que tu saches exactement ce que tu obtiens en travaillant avec moi.

Vue d'ensemble

Mon process s'articule autour de **5 phases**, alignées sur les 5 zones de mon offre. Ce n'est pas un process linéaire : c'est une **boucle continue** qui se nourrit en permanence des retours utilisateurs.

LE PROCESS EN UN COUP D'ŒIL



↳ RETOUR À 01 VIA LES RETOURS UTILISATEURS

01

PHASE 01

Cadrer

Comprendre la vision, écouter les utilisateurs, prioriser les vrais problèmes à résoudre. Avant de construire, on s'assure de construire la bonne chose.

CE QUE JE FAIS

Aligner sur la vision et la stratégie business

Comprendre la vision, le marché, le modèle économique, les objectifs à 6-12 mois. Sans cette compréhension, impossible de prioriser correctement.

Écouter les utilisateurs et les clients

La seule boussole fiable. Je m'appuie sur :

- Les retours du support
- Les retours sales et customer success
- Les interviews utilisateurs (existantes ou que je conduis)
- Les données d'usage si disponibles
- Les conversations directes avec les utilisateurs clés

Prioriser et définir les projets

On identifie les vrais problèmes à résoudre, on définit les projets dans Linear, et on écrit le **Pourquoi** : pour qui, quel problème, quel impact, comment on saura que c'est réussi.

POURQUOI C'EST IMPORTANT

Sans cadrage, on construit vite mais on construit mal. Les devs livrent des features qui ne résolvent pas le bon problème. La roadmap dérive.

Cette phase enlève **l'ambiguïté avant que le code ne soit écrit.**

LIVRABLES CONCRETS

- Roadmap produit structurée et priorisée
- Projets cadrés dans Linear avec le "Pourquoi" rédigé
- Synthèses des retours utilisateurs exploitables par toute l'équipe

02

PHASE 02

Concevoir

Traduire les besoins cadrés en parcours et en écrans clairs. Simplifier sans appauvrir. Co-construire avec les bonnes parties prenantes.

CE QUE JE FAIS

Designer les parcours et les écrans

Maquettes Figma et parcours utilisateurs, avec une attention particulière sur les écrans complexes :

- Tableaux denses
- Dashboards avec beaucoup de données
- Interfaces métier à forte charge cognitive
- États vides, états d'erreur, cas limites

Simplifier sans appauvrir

Réduire la friction sans perdre la richesse fonctionnelle. Un écran clair n'est pas un écran pauvre : c'est un écran où l'utilisateur comprend immédiatement ce qu'il regarde et ce qu'il peut faire.

Co-construire avec les stakeholders

Le design ne se fait jamais seul. J'itère avec le CEO pour valider l'alignement business, les développeurs pour anticiper la faisabilité, et les utilisateurs pour valider la compréhension réelle. Pas de design "joli" qui meurt au moment du dev.

POURQUOI C'EST IMPORTANT

Un produit B2B mal designé devient **invendable**. Les démos ratent, l'onboarding bloque, le support sature.

À l'inverse, un produit clair se vend plus vite et se fait adopter plus facilement.

LIVRABLES CONCRETS

- Maquettes Figma des écrans et parcours
- Spécifications visuelles (composants, états, comportements)
- Itérations documentées avec les décisions de design

03

PHASE 03

Préparer le delivery

Transformer le design validé en tickets prêts à coder. Specs claires, découpage propre, critères d'acceptation testables.

CE QUE JE FAIS

Rédiger les specs fonctionnelles

Une fois le design validé, je formalise les règles métier, les comportements attendus, les cas nominaux, les cas limites et les cas d'erreur.

Créer les tickets exploitables

Chaque ticket est structuré pour être pris en sprint sans questions de dernière minute :

- **Contexte** : pourquoi ce ticket existe
- **Objectif** : le résultat attendu
- **Éléments à implémenter** : décisions produit et UX
- **Tâches à réaliser** : liste opérationnelle
- **Critères d'acceptation** : précis, testables, non ambigus

Découper Front / Back / Général

Les sujets complexes sont découpés selon la couche concernée. Les dépendances sont explicites. Les risques de régression sont signalés.

POURQUOI C'EST IMPORTANT

Des specs floues = des devs qui posent des questions en boucle, des allers-retours, des reprises, du temps perdu.

Des specs claires = **des devs qui codent sans friction et livrent juste du premier coup.**

LIVRABLES CONCRETS

- Tickets Linear prêts à être pris en sprint
- Specs fonctionnelles documentées
- Critères d'acceptation testables

PAUSE · ON EN PARLE ?

Tu reconnais ces problèmes dans ton équipe ?

Specs floues, écrans denses, devs bloqués, arbitrages en boucle. Si tu lis ce document, c'est qu'au moins un de ces sujets te parle.

30 minutes pour voir ensemble si je peux t'aider à structurer tout ça.

[Réserver un appel découverte →](#)

Sans engagement · Sans présentation commerciale

04 PHASE 04 Piloter le delivery

Animer les sprints, débloquer l'équipe, sécuriser la qualité avant la mise en prod. Tenir le rythme sans devenir un goulot.

CE QUE JE FAIS

Animer les cérémonies agiles

Planning poker pour estimer collectivement, sprint planning pour cadrer, stand-ups pour suivre l'avancement, revues et rétrospectives pour ajuster en continu.

Tenir le rythme

Je m'assure que les sprints avancent, que les blocages remontent vite, que les arbitrages sont pris sans bloquer l'équipe. Disponible en continu pour les devs : réponse sous 24h en asynchrone.

Faire la QA fonctionnelle

Avant chaque mise en prod, je teste les features livrées : cas nominaux, cas limites, cas d'erreur, cohérence avec les autres modules, risques de régression. **Rien ne sort si ce n'est pas solide.**

POURQUOI C'EST IMPORTANT

Sans pilotage, les sprints dérivent, les tickets s'accumulent, les bugs passent en prod.

Avec un pilotage clair, l'équipe sait où elle va, livre régulièrement, et toi tu peux **te concentrer sur la stratégie** au lieu d'arbitrer en permanence.

LIVRABLES CONCRETS

- Sprints structurés et suivis
- Bilans de sprint (objectifs vs réalisés)
- Features validées en QA avant mise en prod

05

PHASE 05

Livrer et faire adopter

Une feature livrée mais pas comprise est une feature perdue. Cette phase garantit que les utilisateurs voient, comprennent et utilisent ce qui sort.

CE QUE JE FAIS

Rédiger les release notes

Pour chaque mise en prod : des release notes claires, orientées utilisateur, qui expliquent ce qui change, pourquoi, et comment l'utiliser.

Mettre à jour la documentation

Manuel utilisateur, FAQ, onboarding. La documentation suit les évolutions du produit, pas l'inverse.

Communiquer aux utilisateurs

Vidéos de présentation, emails de communication, notes internes pour le support et le sales.

Boucler avec la phase 1

La mise en prod n'est pas la fin. C'est le début d'une nouvelle boucle : mesurer l'adoption, récolter les retours, ajuster la roadmap.

POURQUOI C'EST IMPORTANT

Trop de features livrées ne sont jamais vraiment utilisées parce qu'elles ne sont pas comprises.

La communication produit est aussi importante que la feature elle-même. Et le support reçoit moins de tickets quand les utilisateurs comprennent ce qui change.

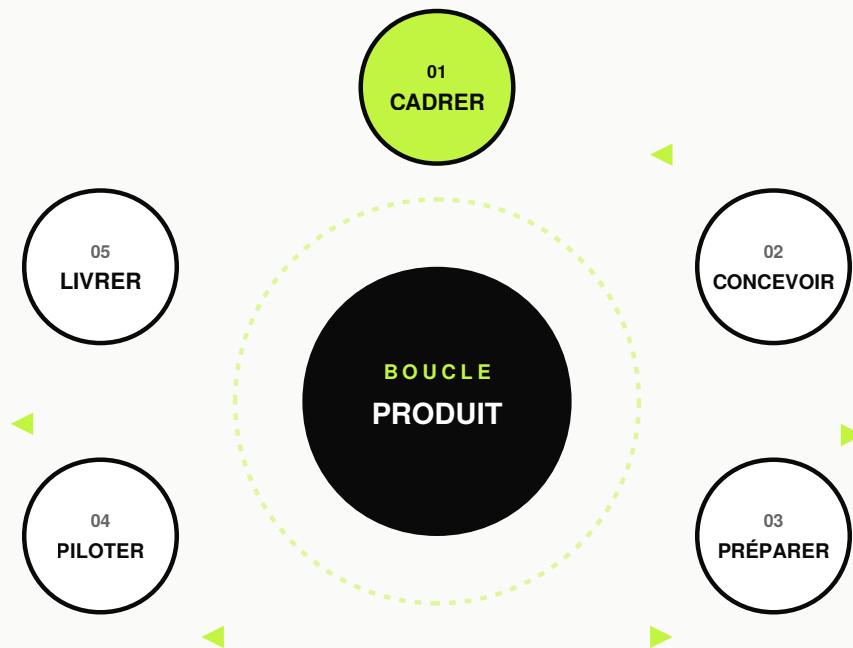
LIVRABLES CONCRETS

- Release notes prêtes à publier
- Documentation utilisateur à jour
- Vidéos et communications aux clients

LE CŒUR DU PROCESS

Une **boucle** qui se nourrit en permanence.

Mon process n'est pas linéaire. Chaque feature livrée alimente la prochaine phase de cadrage grâce aux retours utilisateurs.



↳ NOURRIE EN PERMANENCE PAR LES RETOURS UTILISATEURS

C'est ce qui garantit qu'on construit un produit qui résout de **vrais problèmes**, pas un produit qui empile des fonctionnalités.

Ce que ça change **concrètement.**

En travaillant avec moi sur ce process, voici ce que tu obtiens.



Clarté

Tu sais ce qui se fait, pourquoi, dans quel ordre, et pour qui.



Vitesse

Les devs livrent sans friction, les sprints tiennent leur cap.



Qualité

Ce qui sort en prod est solide, compris, et utilisé.



Temps

Tu te concentres sur la stratégie, pas sur les arbitrages quotidiens.



Sérénité

Tu as un partenaire opérationnel qui porte le produit avec toi.



Adoption

Tes features sont comprises et utilisées par les utilisateurs.

En résumé

PHASE	CE QUE JE PORTE	CE QUE TU OBTIENS
01 · Cadrer	Vision, écoute users, priorisation	Une roadmap claire et fondée sur le terrain
02 · Concevoir	Design, parcours, simplification	Un produit plus clair et plus vendable
03 · Préparer	Specs, tickets, critères d'acceptation	Des devs qui livrent juste du premier coup
04 · Piloter	Sprints, cérémonies, QA	Une équipe qui tient le rythme
05 · Livrer	Release notes, doc, communication	Des features adoptées, pas juste livrées

ON EN PARLE ?

30 minutes pour voir si on est alignés.

Sans engagement, sans présentation commerciale.
On regarde ton contexte, tes priorités, et si je peux t'aider.

[Réserver un appel découverte →](#)

Ou écris-moi directement : kevin.silve@outlook.com

KÉVIN SILVE · PRODUCT PARTNER POUR SAAS B2B · NEALABS.STUDIO