

ÉDITION 2026 • PRODUIT, CONTENU ET CODE

25 prompts IA vraiment exploitables

Des canevas précis pour réfléchir, écrire, concevoir et développer. Remplacez les éléments entre crochets et ajoutez vos données réelles.

DOCUMENT PRATIQUE

DEV WRITER

Bien les utiliser

Donnez toujours le contexte, le public, l'objectif, les contraintes et un exemple du résultat attendu. Ne transmettez jamais de données confidentielles. Vérifiez les faits, le code et les décisions importantes avant de les utiliser.

01 — Clarifier une idée de produit

Agis comme un stratège produit. À partir de cette idée : [IDÉE], identifie le public prioritaire, son problème le plus coûteux, la promesse centrale, trois risques et cinq questions à valider avant de construire. Ne suppose rien : marque chaque hypothèse.

02 — Construire une proposition de valeur

Transforme ces informations [CONTEXTE] en une proposition de valeur claire : cible, problème, résultat promis, mécanisme distinctif et preuve. Propose cinq formulations sans superlatifs ni jargon.

03 — Préparer des entretiens utilisateurs

Crée un guide de 12 questions ouvertes pour comprendre comment [PUBLIC] gère actuellement [PROBLÈME]. Évite les questions orientées et termine par une grille de synthèse des réponses.

04 — Prioriser une première version

Classe ces fonctionnalités [LISTE] avec la méthode impact / effort / risque. Définis un noyau indispensable livrable en [DÉLAI], ce qui attendra et les critères de décision.

05 — Écrire une page d'accueil

Rédige la structure et le texte d'une page d'accueil pour [OFFRE] destinée à [PUBLIC]. Objectif : [ACTION]. Inclus titre, sous-titre, bénéfices, preuves, objections, FAQ et appels à l'action. Ton : [TON].

06 — Améliorer un texte commercial

Réécrit ce texte [TEXTE] pour le rendre concret, naturel et crédible. Supprime le jargon, remplace les affirmations vagues par des bénéfices observables et conserve le sens. Donne ensuite les principaux changements.

07 — Trouver des angles de contenu

Propose 15 sujets de contenu pour [EXPERTISE] destinés à [PUBLIC]. Répartis-les entre pédagogie, opinion, étude de cas, coulisses et conversion. Pour chaque sujet, donne un angle précis et une accroche.

08 — Créer un plan d'article

Prépare un plan approfondi sur [SUJET] pour [PUBLIC]. Intention de recherche : [INTENTION]. Inclus questions à traiter, exemples nécessaires, objections et conclusion utile. Évite les titres génériques.

09 — Transformer une expertise en publication

À partir de ces notes [NOTES], rédige une publication LinkedIn de [LONGUEUR] mots. Commence par une observation forte, développe une seule idée avec un exemple, puis termine par une question sincère. Aucun cliché.

10 — Auditer une page

Analyse cette page ou ce texte [CONTENU] selon clarté, crédibilité, hiérarchie, friction et appel à l'action. Pour chaque problème, cite le passage, explique l'impact et propose une correction priorisée.

11 — Concevoir une navigation

Voici les contenus et objectifs d'un site : [CONTENU]. Propose une architecture simple pour [PUBLIC], avec menu principal, pages, ordre des sections et action attendue sur chaque page. Justifie chaque choix.

12 — Préparer une FAQ

À partir de [OFFRE] et de ces objections [OBJECTIONS], rédige 10 questions-réponses courtes. Réponds directement, sans esquivé, et indique les informations manquantes à confirmer.

13 — Rédiger des messages d'interface

Écris les textes pour les états vide, chargement, succès et erreur de [FONCTION]. Chaque message doit expliquer la situation et la prochaine action en moins de 18 mots.

14 — Examiner l'accessibilité

Passe en revue cette interface décrite ici [DESCRIPTION]. Dresse une liste de contrôles clavier, focus, contraste, libellés, ordre de lecture et messages d'erreur. Priorise les blocages.

15 — Préparer des cas de test

À partir de cette fonctionnalité [SPÉCIFICATION], crée des cas de test : parcours normal, données invalides, limites, permissions, réseau lent et reprise après erreur. Présente précondition, action et résultat attendu.

16 — Débuguer méthodiquement

Voici le comportement attendu [ATTENDU], le comportement observé [OBSERVÉ], les erreurs [ERREURS] et le code pertinent [CODE]. Formule trois hypothèses classées, puis propose le test minimal permettant de confirmer chacune.

17 — Relire du code

Examine ce code [CODE] pour repérer erreurs, sécurité, performance, accessibilité et maintenabilité. Ne réécris pas tout. Classe les remarques par gravité avec une correction précise.

18 — Documenter une fonctionnalité

Rédige une documentation concise pour [FONCTION] à partir de [INFORMATIONS]. Inclus objectif, prérequis, étapes, exemples, erreurs fréquentes et vérification finale.

19 — Préparer une API

Transforme ce besoin [BESOIN] en contrat d'API : opérations, données d'entrée, réponses, erreurs, authentification, permissions et limites. Signale toutes les décisions encore ambiguës.

20 — Modéliser des données

Propose un modèle de données pour [PRODUIT]. Liste entités, relations, contraintes, informations sensibles, règles d'accès et scénarios de suppression. Optimise d'abord la cohérence, pas la complexité.

21 — Analyser des retours clients

Regroupe ces retours [RETOURS] par problème, fréquence, gravité et segment utilisateur. Distingue symptôme et cause probable, puis propose les cinq actions les plus utiles.

22 — Préparer une offre commerciale

Structure une offre pour [SERVICE] destinée à [CLIENT]. Inclus contexte, objectifs, périmètre, livrables, méthode, calendrier, responsabilités, exclusions et prochaines étapes. Ne crée aucun prix non fourni.

23 — Estimer un projet

Décompose ce projet [BRIEF] en lots, tâches, dépendances et risques. Pour chaque lot, donne une fourchette de charge et les hypothèses. Termine par les questions qui peuvent modifier fortement l'estimation.

24 — Planifier un lancement

Crée un plan de lancement sur [DURÉE] pour [PRODUIT]. Répartis préparation, tests, communication, support et mesure. Ajoute responsable, échéance, preuve attendue et plan de retour arrière.

25 — Faire une rétrospective

À partir de ces faits [FAITS], mène une rétrospective sans blâme : réussites, difficultés, causes, leçons et actions. Chaque action doit avoir un responsable, une date et un indicateur observable.

Créé par Marie-Otniel Houédoté — DEV WRITER • marieotniel.lovable.app