

Je veux que tu configures ce dossier de projet comme mon assistant exécutif personnel / second cerveau dans Claude Code. Tu vas faire ça en 3 phases. Termine complètement chaque phase avant de passer à la suivante.

Phase 1 : Créer la structure de dossiers

D'abord, initialise un repo git dans ce dossier s'il n'en existe pas déjà un.

Ensuite, crée la structure modèle suivante. Ne mets pas encore de contenu dans les fichiers, crée juste le squelette avec des fichiers vides là où c'est indiqué.

```
CLAUDE.md # Fichier cerveau principal (on le remplira en Phase 3)
CLAUDE.local.md # Mes overrides personnels (ignoré par git)
.gitignore # Ignorer .env, CLAUDE.local.md, settings.local.json
.claude/
  settings.json # Objet JSON vide pour l'instant : {}
  rules/ # On ajoutera les fichiers de règles en Phase 3
  skills/ # Vide, on construira des skills au fil du temps
context/
  me.md # À propos de moi (rempli en Phase 3)
  work.md # Détails de mon activité/travail (rempli en Phase 3)
  team.md # Mon équipe (rempli en Phase 3)
  current-priorities.md # Ce sur quoi je suis concentré actuellement (rempli en Phase 3)
  goals.md # Objectifs trimestriels et jalons (rempli en Phase 3)
templates/
  session-summary.md # Modèle pour la clôture de session
references/
  sops/ # Procédures opérationnelles standard (vide pour l'instant)
  examples/ # Exemples de livrables et guides de style (vide pour l'instant)
projects/ # Workstreams actifs (vide pour l'instant)
decisions/
  log.md # Journal de décisions (append-only, vide pour l'instant)
archives/ # Matériel terminé/obsolète (vide pour l'instant)
```

Pour les répertoires vides, crée un fichier `.gitkeep` à l'intérieur pour que git les suive.

Pour `templates/session-summary.md`, utilise ce modèle de départ :

```
# Récapitulatif de session

**Date :**
**Focus :**

## Ce qui a été fait
-

## Décisions prises
-

## Points ouverts / Prochaines étapes
-

## Mises à jour mémoire
```

- Préférences apprises :
- Décisions à journaliser :

Pour `decisions/log.md`, utilise ce démarrage :

```
# Journal de décisions

Append-only. Quand une décision significative est prise, journalise-la ici.

Format : [AAAA-MM-JJ] DÉCISION : ... | RAISONNEMENT : ... | CONTEXTE : ...

---
```

Pour `.gitignore` :

```
.env
CLAUDE.local.md
.claude/settings.local.json
node_modules/
```

Pour `CLAUDE.local.md` :

```
# Overrides locaux

Préférences personnelles et overrides qui ne sont pas partagés via git.
Ajoute ici tout ce qui est spécifique à ta configuration locale.
```

Phase 2 : Pose-moi les questions d'onboarding

Avant de remplir les fichiers de contexte, les règles et CLAUDE.md, interview-moi. Pose ces questions une section à la fois. Ne balance pas toutes les questions d'un coup, va section par section et attends mes réponses avant de passer à la suivante.

Section 1 : À propos de toi

- Quel est ton nom ?
- Quel est ton rôle/titre ? (ex. CEO, freelance, créateur de contenu, développeur)
- Quel est ton fuseau horaire ?
- En une phrase, qu'est-ce que tu fais ?
- Quelle est ta priorité n°1, la chose que tout le reste doit soutenir ?

Section 2 : Ton activité / travail

- Comment s'appelle ton entreprise ou ton activité ? (ou décris ton travail si tu n'es pas chef d'entreprise)
- Quels sont tes produits, services ou sources de revenus ? (Liste chacun avec une phrase)

- À peu près combien chacun génère-t-il de revenus ? (optionnel, mais ça m'aide à prioriser)
- Quels outils utilises-tu au quotidien ? (ClickUp, Notion, Slack, Google Workspace, etc.)
- As-tu déjà des serveurs MCP connectés à Claude Code ? (Si tu ne sais pas ce que ça veut dire, dis simplement non)

Section 3 : Ton équipe

- As-tu une équipe ? Si oui, combien de personnes ?
- Qui sont les 2-3 personnes clés que je devrais connaître ? (nom, rôle, quand les inclure)
- Où ton équipe communique-t-elle ? (ClickUp, Slack, etc.)
- Quel est ton plus gros point de douleur dans la gestion de ton équipe ?

Section 4 : Priorités, objectifs et projets

- Quelles sont les 3-5 choses sur lesquelles tu es le plus concentré actuellement ?
- Y a-t-il des deadlines ou des éléments urgents que je devrais connaître ?
- As-tu des objectifs trimestriels ou des jalons que tu suis ? (Si oui, liste-les. Si tu ne fais pas d'objectifs formels, c'est ok, tes priorités ci-dessus suffiront.)
- Quels projets ou workstreams actifs gères-tu en ce moment ? (Ce sont des initiatives spécifiques, pas des responsabilités continues. Ex. « lancer un nouveau cours », « recruter un VA », « refondre le site web »)

Section 5 : Préférences de communication

- Comment aimes-tu que l'information soit présentée ? (puces, paragraphes détaillés, etc.)
- Des manies d'écriture qui t'agacent ? (ex. « jamais d'emojis », « pas de tirets longs », « reste sous 3 phrases »)
- Quel ton veux-tu en interne (décontracté, professionnel, etc.) ?
- Quel ton veux-tu pour le contenu externe / public ?

Section 6 : Sur quoi veux-tu de l'aide ?

- Quelles sont les tâches récurrentes qui te bouffent du temps ? (Liste autant que tu veux)
- Qu'est-ce que tu déléguerais en premier à un assistant si tu pouvais ?
- Y a-t-il des workflows spécifiques que tu veux automatiser ou transformer en modèle ?

Phase 3 : Construire les fichiers

À partir de mes réponses, remplis tous les fichiers :

Fichiers de contexte

- **context/me.md** : Mon profil basé sur les réponses de la Section 1
- **context/work.md** : Détails de mon activité/travail basés sur les réponses de la Section 2

- **context/team.md** : Structure d'équipe basée sur les réponses de la Section 3 (si j'ai une équipe ; si je suis solo, note-le et saute ce fichier)
- **context/current-priorities.md** : Priorités de la Section 4, datées d'aujourd'hui
- **context/goals.md** : Objectifs trimestriels et jalons de la Section 4. Si pas d'objectifs formels, utilise leurs priorités principales comme objectifs informels. Date avec le trimestre courant (ex. « Q2 2026 »). Inclus une note en haut : « Mettre à jour ce fichier au début de chaque trimestre. »

Projets

Si l'utilisateur a listé des projets actifs en Section 4, crée un dossier pour chacun dans `projects/`. Chaque dossier de projet reçoit un bref `README.md` avec :

- Description du projet en une ligne
- Statut actuel (actif, en planification, en pause)
- Dates clés ou deadlines mentionnées

Si aucun projet spécifique n'a été mentionné, laisse `projects/` vide.

Fichiers de règles dans `.claude/rules/`

Crée des fichiers de règles basés sur mes préférences de communication (Section 5). Fichiers suggérés :

- **communication-style.md** : Ton d'écriture, préférences de mise en forme, manies à éviter
- Toute autre règle spécifique au domaine qui ressort de mes réponses (ex. si j'ai mentionné des conventions d'outils spécifiques, des consignes de contenu, etc.)

Garde chaque fichier de règles concentré sur UN seul sujet. Ne crée pas plus de 3-4 fichiers de règles pour démarrer.

CLAUDE.md (le fichier cerveau principal)

C'est le fichier le plus important. Garde-le SOUS 150 lignes. Il doit contenir :

1. **Identité en une ligne** : Qui tu es pour moi (ex. « Tu es l'assistant exécutif de [Nom]. »)
2. **Priorité principale** : La chose n°1 que tout soutient
3. **Imports de contexte** : Utilise `@context/me.md`, `@context/team.md`, `@context/goals.md`, etc. pour référencer les fichiers au lieu de répéter leur contenu
4. **Intégrations d'outils** : Quels outils sont connectés (ClickUp, etc.)
5. **Répertoire de skills** : Pointe vers `.claude/skills/` et explique comment les skills fonctionnent
6. **Journal de décisions** : Pointe vers `decisions/log.md` et explique la convention append-only
7. **Mémoire** : Ajoute une section expliquant comment la mémoire fonctionne :

- La mémoire persistante est stockée dans `.claude/memory/`. Le fichier index est `.claude/memory/MEMORY.md` — il liste tous les sous-fichiers avec une ligne de description chacun.

- Chaque mémoire est un fichier `.md` séparé dans ce dossier (ex : `feedback_language.md`, `reference_mcp_connections.md`).

Important : ces fichiers sont versionnés sur GitHub. Ne jamais y écrire de données sensibles : credentials, clés API, tokens, mots de passe, ou toute information confidentielle.

- « Si tu veux que ton assistant retienne quelque chose de spécifique, dis simplement "souviens-toi que je veux toujours X" et il le sauvegardera. »

- « Mémoire + fichiers de contexte + journal de décisions = ton assistant devient plus intelligent au fil du temps sans que tu aies à tout réexpliquer. »

8. **Garder le contexte à jour** : Ajoute une brève section maintenance :

- Mettre à jour `context/current-priorities.md` quand ton focus change

- Mettre à jour `context/goals.md` au début de chaque trimestre

- Journaliser les décisions importantes dans `decisions/log.md`

- Ajouter des fichiers de référence selon les besoins

- Construire des skills quand tu remarques que tu répètes la même demande

9. **Projets** : Pointe vers `projects/` et explique que les workstreams actifs vivent là

10. **Templates** : Pointe vers `templates/`

11. **Références** : Pointe vers `references/`

12. **Règle d'archivage** : Ne pas supprimer, archiver

Ne mets PAS les règles de style de communication dans CLAUDE.md, elles vont dans `.claude/rules/communication-style.md`.

Ne répète PAS le contenu des fichiers de contexte, importe-les juste avec @.

N'inclus PAS de « Protocole de démarrage de session » qui dit à Claude de lire un tas de fichiers, le système d'imports et les règles gèrent ça automatiquement maintenant.

Répertoire de skills

Laisse `.claude/skills/` vide. Ne crée aucun skill pour l'instant. Assure-toi juste que CLAUDE.md mentionne que les skills vivent là et explique le pattern :

- Chaque skill a son dossier : `.claude/skills/skill-name/SKILL.md`
- Les skills sont construits organiquement à mesure que des workflows récurrents émergent
- Inclus une note sur les réponses de la Section 6 (ce sur quoi ils veulent de l'aide) comme backlog « Skills à construire » dans CLAUDE.md ou un fichier séparé

Étape finale

Une fois tout créé :

1. Montre-moi une vue arborescente de chaque fichier et dossier que tu as créé
2. Montre-moi un bref résumé de ce qu'il y a dans chaque fichier (une ligne par fichier)
3. Liste le backlog « Skills à construire » basé sur mes réponses de la Section 6, ce sont les workflows qu'on transformera en skills au fil du temps
4. Montre-moi cette antisèche de maintenance :

Garder ton assistant affûté

- **Hebdomadaire** : Rien de requis. La mémoire automatique gère les apprentissages quotidiens pour toi.
 - **Mensuel** : Jette un œil à `context/current-priorities.md`. Si ton focus a changé, mets-le à jour.
 - **Trimestriel** : Mets à jour `context/goals.md` avec les nouveaux objectifs et jalons.
 - **Au besoin** : Journalise les décisions dans `decisions/log.md`. Ajoute des fichiers de référence. Construis de nouveaux skills.
 - **Astuce de pro** : Si tu veux que ton assistant retienne quelque chose en permanence, dis-le-lui simplement : « Souviens-toi que je préfère toujours X. » Il le sauvegardera dans toutes les conversations futures.
5. Crée le premier commit git avec tous les fichiers
 6. Demande-moi si je veux construire l'un de ces skills tout de suite

Règles importantes pour toi (Claude)

- Ne crée AUCUN skill pour l'instant. Le répertoire skills doit être vide après le setup.
- Garde CLAUDE.md SOUS 150 lignes. S'il devient long, c'est que tu y mets trop de choses.
- Utilise les imports @ (ex. @context/me.md) dans CLAUDE.md au lieu de répéter l'information.
- Un fichier de règle = un sujet. Max 3-4 fichiers de règles pour démarrer.
- Pose les questions d'onboarding UNE SECTION À LA FOIS. Attends ma réponse avant de poser la section suivante.
- Si je dis « passe » ou « je remplirai plus tard » pour une section, crée le fichier avec un placeholder et avance.