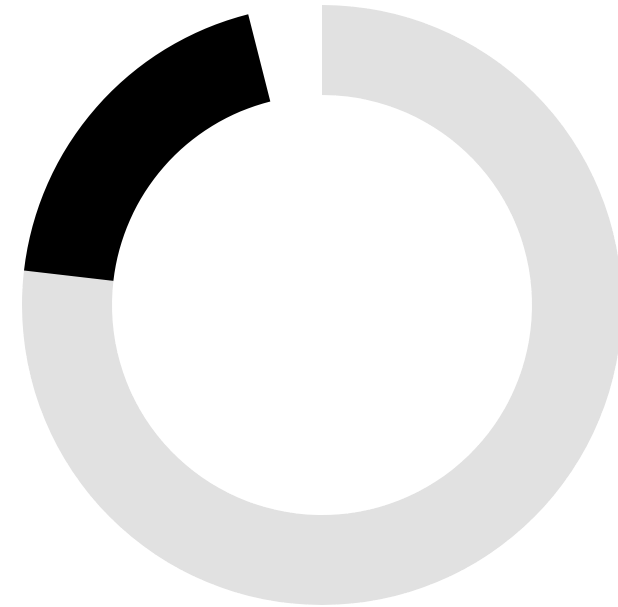


# Le Paradoxe du Développement IA

**< 20%**

Taux de succès des prompts uniques en production

La plupart des développeurs sont bloqués dans le "codage intuitif" - utilisant les outils IA sans structure ni vérification.



# Qui Nous Servons & Ce Que Nous Livrons

## Développeurs Freelance

Recherchant des gains de productivité et un avantage concurrentiel

---

## CTO de Startups

Cherchant à évoluer efficacement avec des ressources limitées

---

## Équipes de Développement

Nécessitant une intégration rapide et des sprints prévisibles

**"Transformez votre stack de développement en machine de livraison native IA"**

**3x**

Développement Plus Rapide

**90%**

Réduction des Bugs

# De ChatGPT aux Systèmes d'Agents Orchestrés

● 2022

## Assistants IA Simples

ChatGPT, GitHub Copilot - Aide ponctuelle au codage

2023

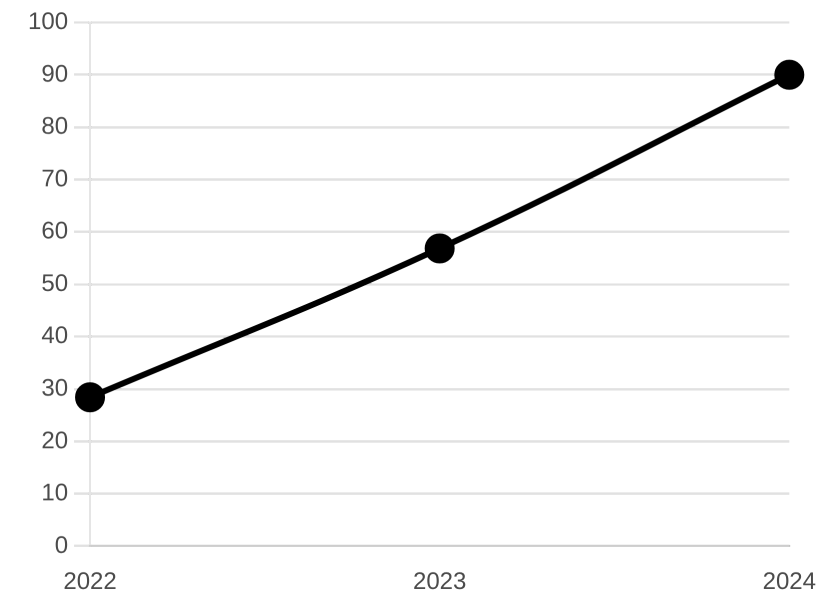
## Outils Spécialisés

Claude Code, Cursor - Intégration IDE et contexte étendu

2024

## Systèmes Multi-Agents

Pheromind, RooCode - Orchestration coordonnée d'agents spécialisés



# Pourquoi les Prompts Uniques Échouent en Production

Le "codage intuitif" manque de structure et de vérification systématique

## Prompts Ambigus

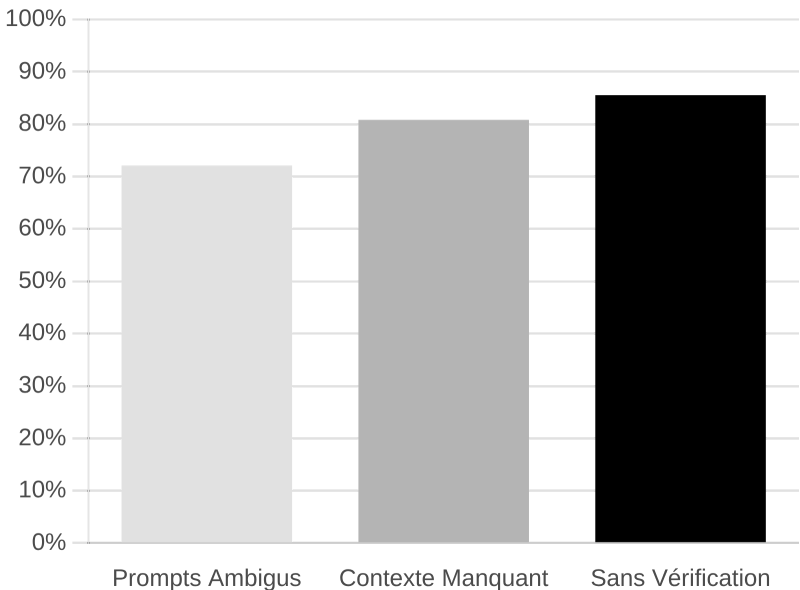
Instructions vagues qui mènent à des résultats imprévisibles

## Contexte Manquant

L'IA ne comprend pas les contraintes du projet réel

## Absence de Vérification

Aucun mécanisme pour valider la qualité du code généré



# La Solution d'Alignement : Développement IA Structuré

**Le codage agentique résout le problème d'alignement par la structure et la vérification**

## Spécifications Claires

Instructions précises et contexte complet pour l'IA

## Agents Spécialisés

Chaque agent a un rôle défini et des capacités optimisées

## Vérification Automatique

Oracles de test pour valider la qualité en continu

### Le Problème Oracle

Comment déterminer si un test a réussi ou échoué sans mécanisme de vérification définitif ?

Le codage agentique implémente des oracles automatisés pour résoudre ce défi fondamental.

# Choisir la Bonne IA pour le Bon Travail

## Claude-3 Opus

Architecture & Conception

Raisonnement complexe et planification système

## Claude-3 Sonnet

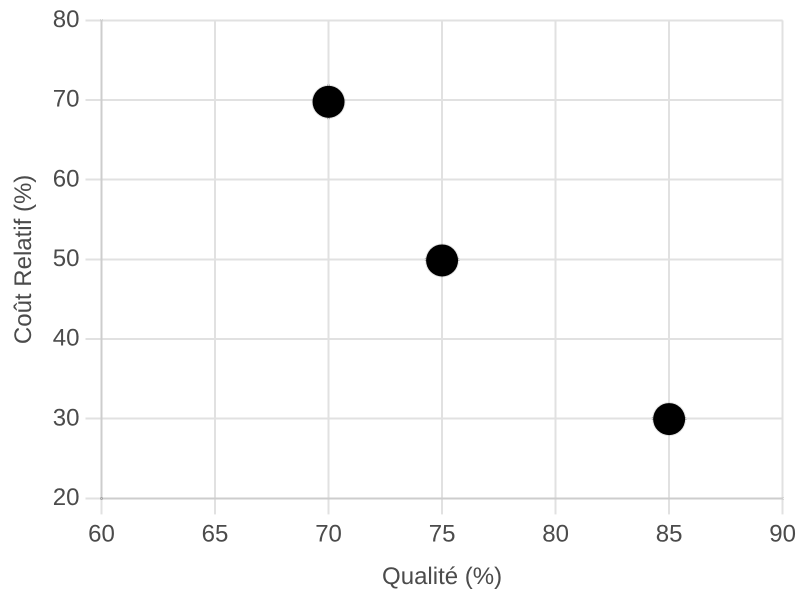
Développement & Implémentation

Équilibre optimal entre qualité et vitesse

## Gemini-1.5 Pro

Débogage & Analyse

Détection d'erreurs et optimisation de code



# Construire des Exigences Projet Lisibles par l'IA

**Des spécifications claires sont la fondation du succès en codage agentique**

## Objectifs Mesurables

Définir des critères de succès quantifiables et vérifiables

## Contraintes Techniques

Spécifier les limitations, dépendances et exigences système

## Critères d'Acceptation

Conditions précises pour valider chaque fonctionnalité

## Framework PRD pour l'IA

- Contexte métier et utilisateurs cibles
- Fonctionnalités avec exemples concrets
- Architecture technique recommandée
- Tests de validation automatisés
- Métriques de performance attendues
- Scénarios d'erreur et gestion

# Définir le Contexte Agent Parfait

**Un contexte bien défini multiplie  
l'efficacité des agents IA**

## Contexte Projet

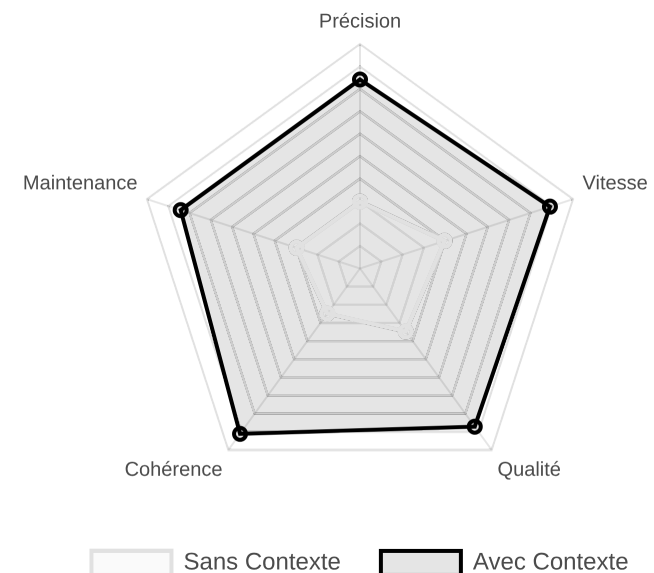
Vision globale, objectifs métier et contraintes organisationnelles

## Contexte Technique

Architecture existante, stack technologique et standards de code

## Contexte Opérationnel

Processus de déploiement, environnements et procédures de test





# Résoudre le Défi de Vérification

## Le Problème Oracle

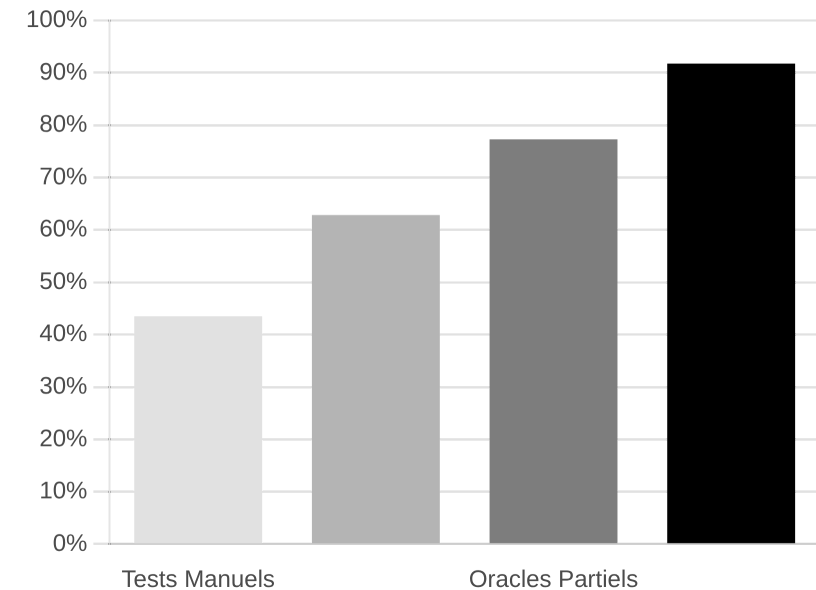
Comment déterminer si un test a réussi ou échoué sans mécanisme de vérification définitif ?

## Oracles Automatisés

Génération automatique de tests et validation continue

## Vérification Multi-Agents

Plusieurs agents valident indépendamment les résultats

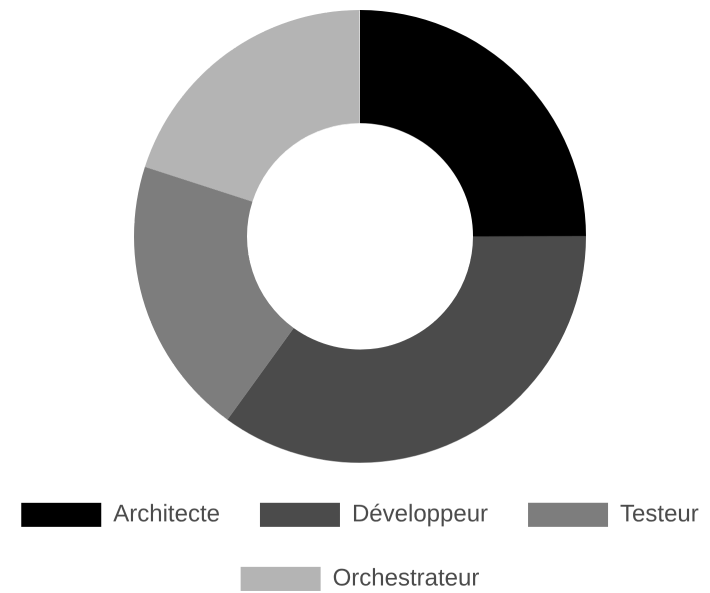


# Construire Votre Équipe de Développement IA

**L'orchestration multi-agents  
transforme le développement en  
processus coordonné**

## Méthodologie SPARC

- Specification - Définition claire des exigences
- Pseudocode - Architecture logique détaillée
- Architecture - Conception système globale
- Refinement - Amélioration itérative
- Completion - Finalisation et validation



# Étendre les Capacités IA avec MCP

## Model Context Protocol

Protocole standardisé pour connecter les IA aux outils externes, APIs et sources de données

## Perplexity

Capacités de recherche en temps réel

## GitHub

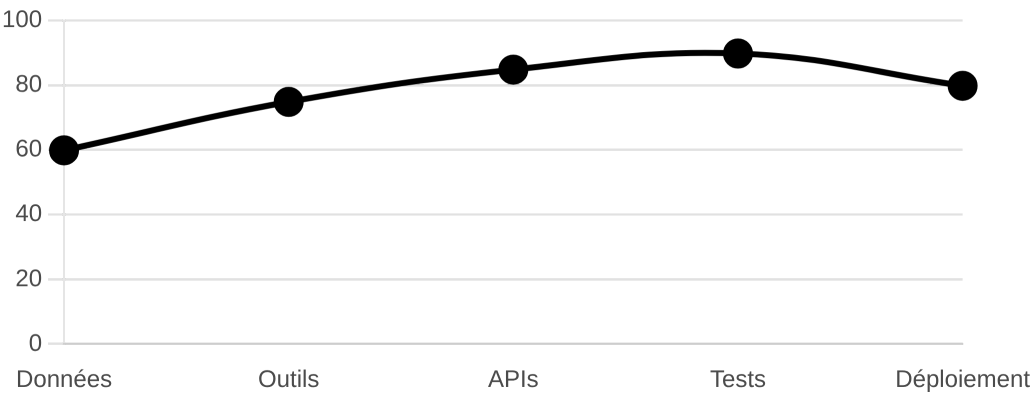
Gestion de dépôts et intégration continue

Bases de Données

APIs Métier

Outils de Test

Frameworks Dev



# **De la Théorie à la Pratique : RooCode en Action**

# Ce Que Nous Avons Construit avec le Codage Agentique

Le codage agentique livre constamment des gains de productivité 3-5x

## Plateforme E-Commerce

Reconstruction complète d'une plateforme legacy avec développement piloté par spécifications

73%

Réduction Temps

92%

Succès Tests

## Dashboard Analytique

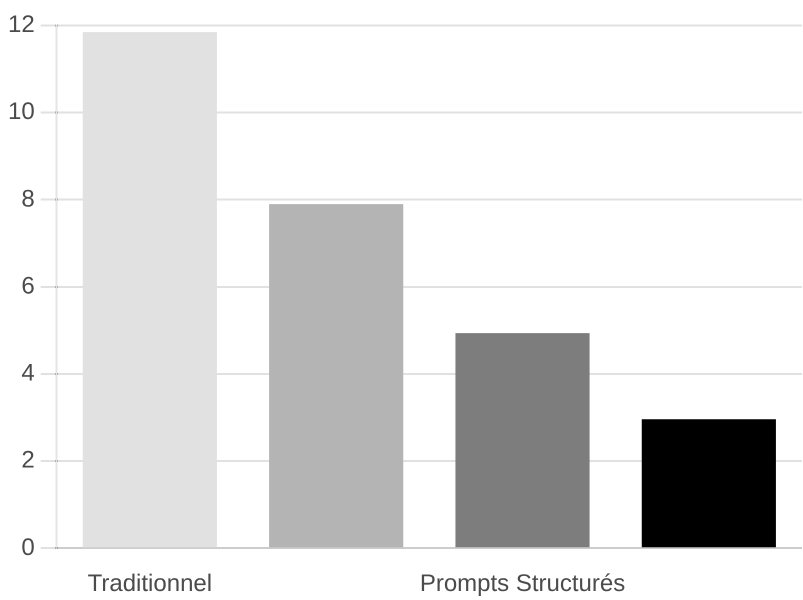
Plateforme de visualisation financière temps réel avec contraintes sécuritaires strictes

68%

Réduction Coûts

3.5x

Performance



# Transformez Votre Processus de Développement Aujourd'hui

## 1

### Évaluation

Analyser votre processus actuel et identifier les opportunités d'intégration IA

## 2

### Formation Équipe

Développer les compétences en codage agentique et développement IA structuré

## 3

### Mise à l'Échelle

Étendre l'implémentation et optimiser continuellement votre workflow IA

### Prêt à Transformer Votre Développement ?

Rejoignez notre communauté de praticiens du codage agentique et commencez à construire de meilleurs logiciels plus rapidement.

[Planifier une Consultation](#)