



Think Tank – Do Tank

sur l'IA Générative appliquée au domaine de l'Ingénierie Logicielle

3ème édition – Mercredi 26 janvier

Groupe de Travail : Vibe Coding / Serveurs MCP.

GT « Vibe Coding / Serveurs MCP »

Réunion Générale - 3ème édition

Groupe de Travail : Vibe coding & Server MCP .



Jean Pierre BRAJAL : Manager Dev Journey Team

AIRFRANCE **KLM**

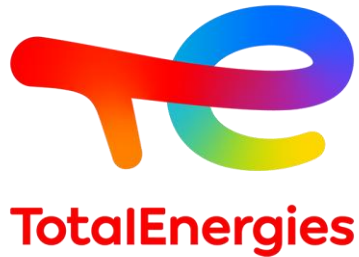


15 minutes

01

Introduction : les tendances émergentes du développeur augmenté

Les sociétés ayant participé au GT



AMADEUS

bpi**france**

in**Vi**vo

ENGIE

hagergroup



AIRFRANCE **KLM**

Introduction

L'année 2025 a été structurante sur l'impact de l'IA dans le milieu de l'ingénierie logicielle. Deux tendances ont été particulièrement marquantes :



L'apparition du « Vibe Coding »



Décrire en langage naturel (écrit ou oral) ce que l'on veut coder, et laisser l'IA générer le code



L'émergence de MCP (*Model Context Protocol*)



Protocole standard d'exposition d'outils (bases de données, APIs, fichiers), aux agents IA



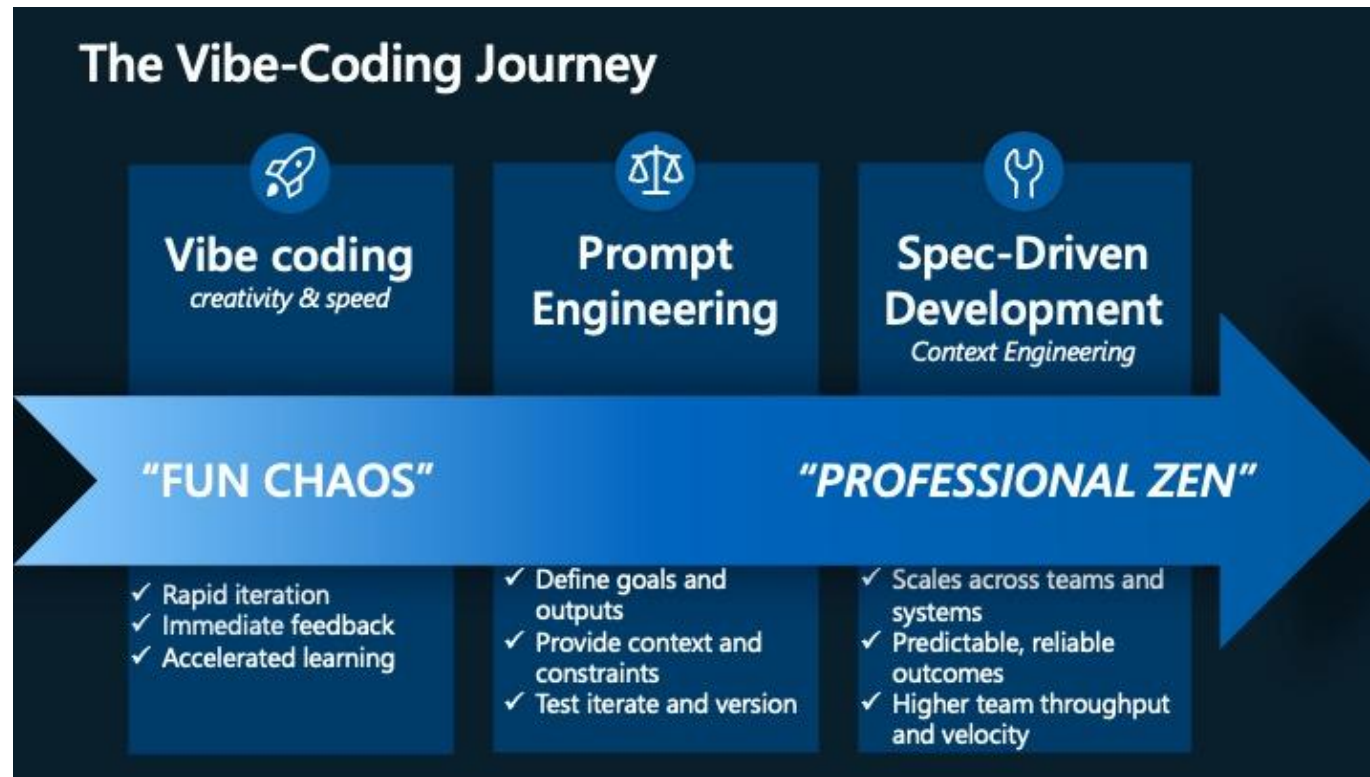
Toutes deux permettent de **présager le futur du développement logiciel**, et l'impact de l'IA sur l'ensemble des activités associées.

02

Focus sur le Vibe Coding

Vibe Coding - Introduction

Décrire en langage naturel ce que l'on veut coder, l'IA génère : c'est le cœur du Vibe Coding.
Au-delà du chat et de l'édition assistée, une approche agentique permet de concevoir automatiquement des parties d'applications



Source : AWS ([lien](#))

Des cas d'usage qui ont gagné en complexité et valeur ajoutée dans le courant de l'année 2025

Si le Vibe Coding a été associé sur le premier semestre à des **cas d'usage simples et rapides**

...



Le second semestre 2025 a marqué un tournant, avec l'apparition de **cas d'usage plus complexes**

Des prérequis légers mais essentiels, afin d'éviter que la vitesse ne se traduise par une perte de qualité



Au niveau du Vibe-Coder



Au niveau de l'entreprise

Le Vibe Coding nécessite des fondamentaux (installation d'outils, gestion de fichiers) et une culture du prompt. Les pratiques de sécurité et l'automatisation des tests/documentation sont clés. Un cadre d'architecture et des garde-fous organisationnels outillent la vitesse sans sacrifier la qualité.

Les risques

Le Vibe Coding apporte vitesse et créativité, mais comporte des dangers :



Niveau de qualité, de sécurité
et de maintenabilité



Perte de maîtrise



Contrôle des coûts



Risque géopolitique



Les impacts organisationnels associés

**Le goulot d'étranglement devient la
pratique**



**Un nouveau paradigme qui amène à redéfinir la
façon de faire du développement logiciel**

Un sujet qui évolue rapidement

Linus Torvalds is OK with vibe coding as long as it's not used for anything that matters

Linux inventor also discusses Rust in the kernel, Nvidia's proprietary code, and the problem of AI crawlers

 [Tim Anderson](#)

Tue 18 Nov 2025 // 13:38 UTC

Cursor CEO warns vibe coding builds 'shaky foundations' and eventually 'things start to crumble'



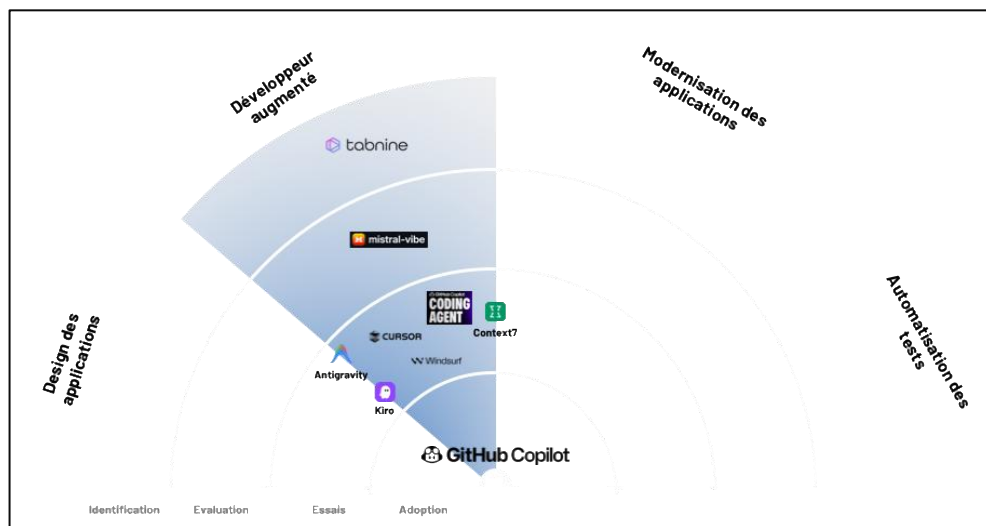
By [Marco Quiroz-Gutierrez](#)
Reporter

December 25, 2025, 4:04 AM ET

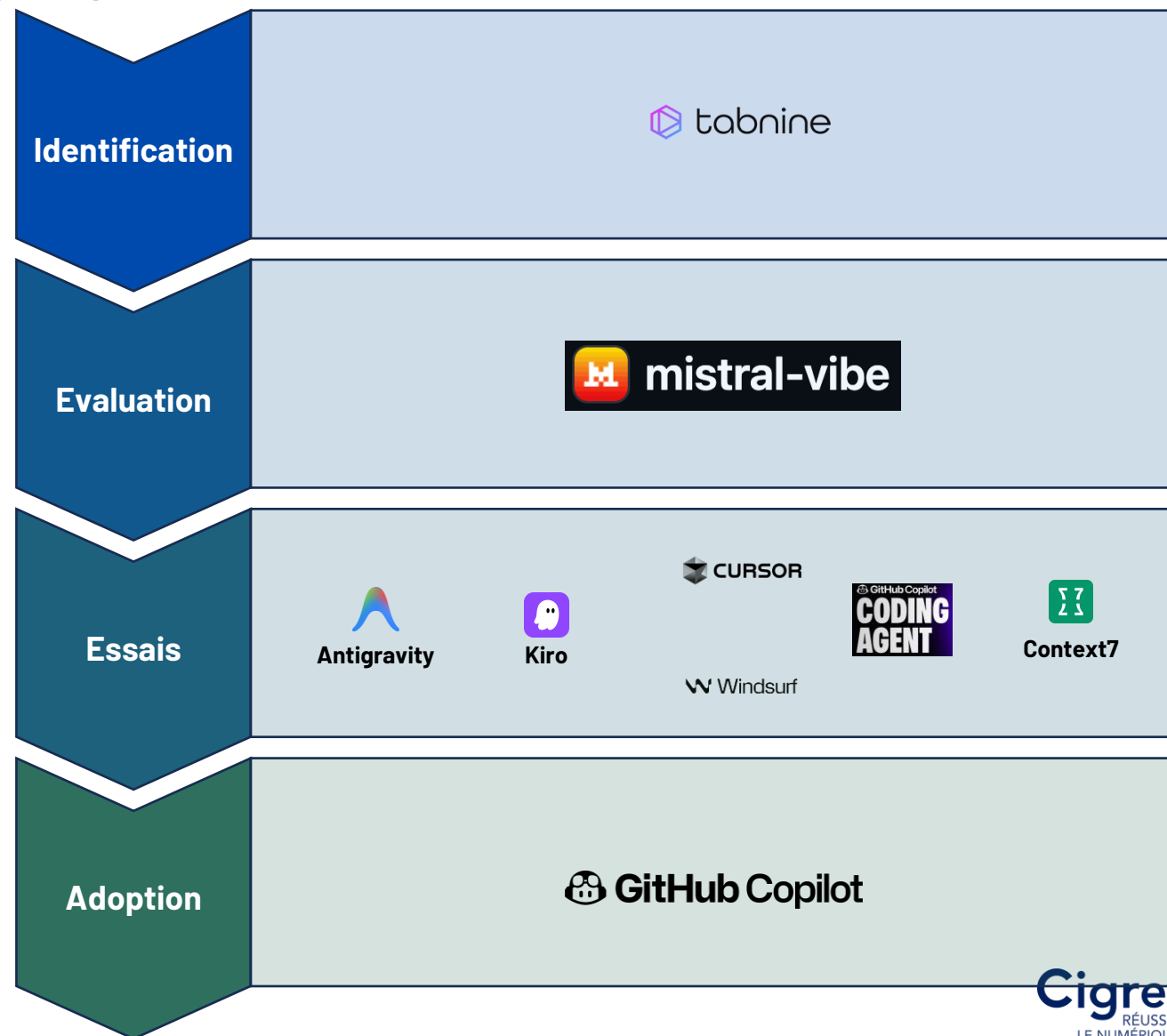
[Add us on](#) 



Vibe Coding – Les différents outils



Radar des outils des GT
(une version complète sera mise à disposition des participants)

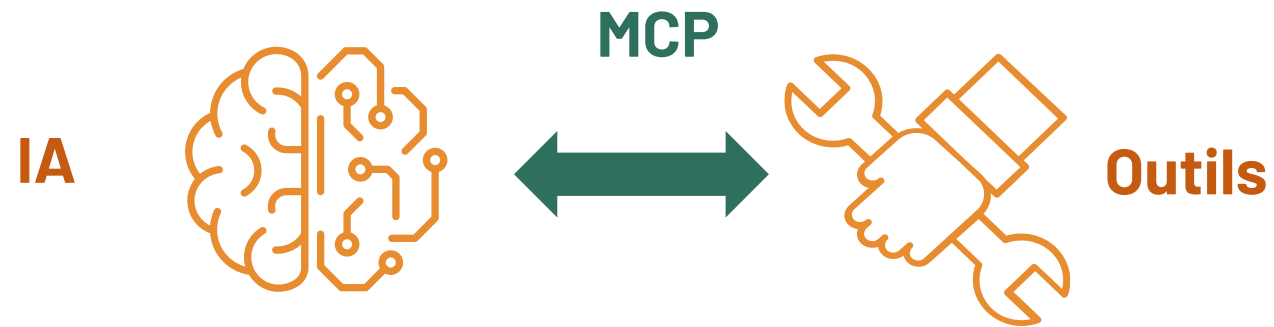


03

Focus sur les serveurs MCP

MCP – Introduction

Les serveurs MCP (*Model Context Protocol*) sont des composants d'un protocole ouvert standardisé qui **connecte les modèles d'IA** à des **outils dynamiques** (bases de données, APIs ou fichiers) via une architecture client-serveur basée sur JSON-RPC.



L'adoption a été accélérée en 2025, avec des milliers de serveurs et un intérêt des éditeurs/fournisseurs. MCP étend les capacités des modèles mais requiert un cadre de sécurité et de gouvernance.

MCP – Cas d'usage



Accès unifié aux outils de Delivery



Rétro-documentation / qualité logicielle



Développeur augmenté dans l'IDE



Intégration connaissance d'entreprise



Automatisation transverse (agents)

Les outils et écosystèmes



Clients MCP

(où brancher les serveurs)



(Desktop / Claude Code)



+ Autres IDE



Microsoft Copilot Studio

Google Workspace



Serveurs MCP

(intégrations prêtes)



Context7

Firebase

ATLASSIAN

Robot Framework



TIME

Figma

Jira

Filesystem

Une bibliothèque partagée de serveurs MCP a été réalisée par le GT et est mise à disposition des participants

MCP – Prérequis (organisationnels & techniques)



GOVERNANCE & PÉRIMÈTRE



SÉCURITÉ & CONFORMITÉ



ENGINEERING READINESS

MCP – Risques & contrôles : ce qu’il faut absolument cadrer



Prompt injection & tool poisoning



Chaine d’approvisionnement



Autorisation/consentement



Maturité du standard



Multiplication des MCP

04

Conclusion et perspectives

Conclusion : plusieurs remises en question à considérer pour 2026



Si les changements sont durables, ceux-ci **restent émergents**, et laissent un certain nombre de **questions ouvertes**

Rôles et compétences

Quels **rôles associés au développement logiciel** de demain ?

Le Vibe Coder devient-il le **développeur augmenté de demain** ?

Méthodologie / Processus

Comment modifier nos **processus** pour appréhender ces technologies ? C'est la condition essentielle pour en tirer des bénéfices !

Impacts applicatifs et technologiques

Quel avenir pour les **solutions de No Code / Low Code** par apport au Vibe Coding ?

Le Vibe Coding est-il une alternative au **modèle de licence des gros fournisseurs de SaaS** ? Doit-on recoder nos grosses licences en local ?

Modèles économiques

Quel sera le **coût du Vibe Coding** demain ? Les nouveaux modèles coûtant de plus en plus cher

Les **sociétés de services** (notamment **offshore** ou **centrées sur la seule production de code**) sont-elles viables désormais ?