

L'IA et La RACHE

L'intelligence artificielle au service de l'approximation

📖 Le Manifeste de l'IA à La RACHE

1. ChatGPT est notre seule source de vérité

Stack Overflow, c'est has-been. La documentation officielle, c'est pour les faibles. ChatGPT/Copilot ont toujours raison, même quand ils inventent des APIs qui n'existent pas.

"Si ChatGPT le dit, ça doit être vrai. Si ça compile pas, c'est que ton environnement est mal configuré."

2. Ne jamais lire le code avant de commit

Lire le code généré par l'IA, c'est douter de son intelligence supérieure. Un vrai développeur La RACHE copie-colle en aveugle et merge directement en prod.

"Cmd+C / Cmd+V / Git push --force. La sainte trinité du développement moderne."

3. Plus le prompt est vague, mieux c'est

Oubliez les guides de "prompt engineering" qui recommandent la précision. À La RACHE, on privilégie le flou artistique :

- "fais moi un truc qui marche"
- "répare le bug"
- "jsp comment faire"
- "fait le code"

"Un bon prompt ne doit jamais dépasser 5 mots. Idéalement 2."

4. Accepter toutes les suggestions Copilot (Tab compulsif)

Copilot propose du code ? Appuyez sur Tab. Sans réfléchir. Sans lire. C'est la voie.

Statistiques d'un vrai développeur RACHE :

- 97% de code généré par Copilot
- 2% de commentaires "// TODO: comprendre ce que ça fait"
- 1% de code écrit à la main (par erreur)

5. Les hallucinations sont des fonctionnalités

ChatGPT invente une librairie ? Un framework imaginaire ? Des fonctions qui n'existent pas ? C'est pas un bug, c'est de la vision. L'IA est en avance sur son temps.

Exemples d'hallucinations légendaires :

- `react-ultra-next-turbo-14` - Framework du futur (pas encore sorti)
- `axios.superFetch()` - Méthode révolutionnaire (n'existe pas)
- `python antigavity --quantum` - Import philosophique

6. Tester c'est douter

Si l'IA a généré le code, il est forcément bon. Les tests unitaires sont pour ceux qui n'ont pas confiance en l'intelligence artificielle.

"Code coverage : 0%. Confiance en l'IA : 100%. C'est ça la vraie agilité."

7. La responsabilité est sur l'IA

Ça bug en prod ? C'est la faute de ChatGPT. Le code casse tout ? Copilot vous a trahi. Vous êtes juste l'intermédiaire innocent.

"Je suis pas responsable, j'ai juste fait Ctrl+V."

📖 Guide du copier-coller aveugle

Niveau 1 : Débutant

Vous commencez à peine votre voyage dans l'approximation assistée par IA.

- Copier-coller depuis ChatGPT sans comprendre
- Ignorer les warnings de compilation
- Commit avec message "wip"
- Espérer que ça marche

Niveau 2 : Intermédiaire

Vous maîtrisez les bases, maintenant place à l'optimisation.

- Copier-coller de 3 sources différentes simultanément (ChatGPT + Copilot + Stack Overflow)
- Mélanger du code dans 2-3 langages différents
- Push direct en prod (skip la CI/CD)
- Désactiver temporairement les tests qui échouent

Niveau 3 : Expert

Vous êtes un maître du non-code. Félicitations.

- Remplacer toute votre équipe par ChatGPT Plus (20\$/mois)
- Générer 10 000 lignes de code sans en lire une seule
- Accepter TOUTES les suggestions Copilot (même contradictoires)
- Certification RACHE-PE obtenue
- LinkedIn bio : "AI-Powered Developer 🤖"

Niveau 4 : Gourou

La légende vivante. Vous ne codez plus, vous channelisez l'IA.

- Vous ne savez plus quel langage vous utilisez
- Votre IDE est juste un terminal pour Copilot
- Vous avez oublié comment écrire du code manuellement
- Votre code est 100% hallucinations mais compile (parfois)
- Les juniors pensent que vous êtes un génie (vous faites Tab en boucle)

🌟 Bonnes pratiques : Ne JAMAIS lire le code avant de commit

Pourquoi lire est une perte de temps

- **Gain de temps** : Lire du code prend 10 minutes. Copier-coller prend 3 secondes. ROI évident.
- **Confiance en l'IA** : Si vous relisez, c'est que vous doutez. Doutez pas.
- **Productivité** : 10x plus de code produit si vous zappez la relecture.
- **Désresponsabilisation** : Si vous lisez pas, vous pouvez pas être coupable.

Comment résister à la tentation de lire

1. Copier en fermant les yeux (technique avancée)
2. Coller immédiatement sans regarder le résultat
3. `git add -A && git commit -m "feat" && git push` en une seule commande
4. Fermer l'IDE avant que la coloration syntaxique vous tente
5. Activer le mode sombre à 100% pour ne rien voir

Que faire quand ça ne compile pas

Stratégies éprouvées :

1. Redemander à ChatGPT de fix (et recopier sans lire)
2. Demander à Copilot de fix le code de ChatGPT
3. Copier l'erreur dans ChatGPT et appliquer la solution (toujours sans lire)
4. Essayer dans un autre langage au hasard
5. Déclarer que "ça marchait sur ma machine"
6. Fermer l'IDE et prendre un café

🏠 Stack Overflow, c'est has-been

Stack Overflow (2008-2022)	ChatGPT (2023-∞)
Réponses de vrais humains	✅ Réponses générées par magie
Code testé et vérifié	✅ Code potentiellement halluciné
Il faut lire et comprendre	✅ Copier-coller direct
"This question has been marked as duplicate"	✅ Répond toujours (même si c'est faux)
Communauté qui juge vos questions	✅ ChatGPT ne juge jamais (il hallucine juste)
Réponse en 2 heures	✅ Réponse en 2 secondes
Gratuit mais anxiogène	✅ 20\$/mois et zéro stress

Verdict : ChatGPT gagne 7-0. Stack Overflow, c'était bien pour nos grands-parents.

🔗 Ressources pour aller plus loin

📖 **Formations IA La RACHE** - Devenez Prompt Engineer certifié (RACHE-PE)

📰 **Actualités IA** - Les derniers retours d'expérience catastrophiques

📜 **Certificat de réussite** - Prouvez que vous savez copier-coller

💡 **Success Stories** - Quand l'approximation mène au désastre

Prêt à rejoindre la révolution de l'IA à La RACHE ?

Ne perdez plus de temps à comprendre votre code. Laissez l'IA penser à votre place.

S'inscrire à la formation RACHE-PE →

Garantie satisfait ou remboursé si vous arrivez à compiler le code généré.

"L'IA ne remplacera pas les développeurs. Elle les rendra juste inutiles."

