**ChatGPT m’aide à écrire un shell Linux et finaliser un développement**

Mon métier est plus orienté sur le développement de bases de données Oracle. Il arrive que les demandes de mes clients dépassent le cadre de mon travail au quotidien ou de mes connaissances.

J’ai par exemple récemment eu à traiter une problématique de compression et de chiffrement de fichier, bien loin de mon domaine habituel plus axés sur de l’informatique de gestion, avec des problématiques orientés produit et plus particulièrement de gestion financière, et des problématiques d’aspect technique qui restent à 99% dans le cadre de développements autour de la base de données.

Jamais de compression ou de chiffrement de données.

Pourtant la problématique de mon client, est au-delà de mes activités quotidienne qui dans ce cas m’invitent à produire un fichier de données respectant la règlementation, d’automatiser son chiffrement.

Oracle ne propose pas de manière native ce genre de fonctionnalités. Aussi j’ai décidé de réaliser un shell Linux qui va gérer ces parties de compression et chiffrement.

Mais je ne fais jamais de développement sur du shell, même si je pourrai trouver en regardant les documentations disponibles et réussir à écrire ce petit script.

Je pourrai mais depuis la mise en place de d’IA je n’ai plus besoin d’effectuer des recherches de documentations pour apprendre à utiliser du shell et utiliser correctement les bonnes instructions avec les bons paramètres.

Il me suffit de bien comprendre le besoin et surtout de bien le restituer et d’expliquer très clairement le contexte à l’IA (ChatGPT dans ce cas).

Rien de confidentiel au niveau du script, il regarde s’il existe des fichiers en attente de traitement, s’il en trouve il va les compresser, chercher la clé de chiffrement et chiffrer le fichier.



Une fois le script mis à disposition par ChatGPT, je n’ai plus qu’à regarder s’il répond à mon besoin. Comprendre ce qui a été écrit, l’adapter si besoin et le tester jusqu’à ce qu’il fonctionne exactement comme je l’attends.

L’IA est beaucoup plus rapide que moi pour écrire le script, d’un langage que je ne maîtrise pas, cependant je peux l’adapter si besoin ou demander à l’IA de le faire pour moi en lui donnant plus d’éléments de contexte dans le prompt.

Elle est aussi beaucoup plus performante que moi pour écrire des instruction de type recherche ou gestion d’expressions régulières dans ce cas.

Elle pourrait me remplacer, mais elle a encore besoin de moi pour lui donner le bon contexte.

Je pense avoir gagné des heures et peut être des jours, car quand on ne maîtrise pas un langage c’est toujours compliqué d’apprendre, et puis on peut facilement bloquer sur des options de commandes que l’on n’a pas l’habitude d’utiliser.

J’aurai peut-être même pu abandonner l’idée de réaliser ce genre de demande, car c’est une demande hors de mon périmètre de compétence mais le fait de comprendre bien le contexte, d’avoir des idées de solutions et de pouvoir être accompagné par cet outil d’intelligence artificielle m’a permis de surmonter ces difficultés techniques.

**ChatGPT m’aide à traiter un ticket client**

Un autre cas d’utilisation où ChatGPT m’a aidé dans mon quotidien de développeur concerne un ticket client avec une demande peu habituelle.

Le contexte est le suivant :

* L’automatisme qui permet de mettre à disposition les factures fournisseurs de mon client (flux XML) dans son Workflow a dysfonctionné sur une période et certaines factures n’ont pas été mises à dispositions.
* Le client dispose de toutes les factures, a une liste de plusieurs centaines de références absentes dans son Workflow.
* L’idée est donc de remettre les factures manquante à disposition du Workflow, sachant que les quelques centaines de factures absentes sont disséminés dans des milliers d’autres

Comment retrouver les fichiers à partir des références factures.

J’ai commencé à récupérer les noms de fichiers manquants à l’aide de requête SQL, ce qui m’a permis de créer un fichier contenant la liste des fichiers manquants.

Mais je n’avais pas envie d’aller pointer un part un les centaines de fichiers manquants dans le répertoire contenant tous les fichiers.

Le serveur client étant hébergé sous Windows, je suis allé vérifier qu’il disposé de PowerShell et à partir de là j’ai demandé à ChatGPT de me construire un répertoire contenant les fichiers manquants.

Il me fait une proposition de script que je relie même si je ne connais absolument rien à PowerShell, c’est un shell Windows, même si je ne suis pas en mesure d’écrire un script, je suis en mesure de le comprendre, d’autant que ChatGPT propose toujours des scripts très commentés de toutes les instructions.

Une fois convaincu que le script pourrait correctement répondre à mon besoin, j’ai juste eu à le lancer et à constater que tout se déroulait correctement, le script a bien alimenté le répertoire avec les fichiers manquants que nous avons pu déposer dans le Workflow de mon client.

Le gain de temps est incroyable, la manière la plus brute aurait été d’aller chercher les fichiers un à un (il y en avait des centaines), je n’ai jamais envisagé cette options.

Mes outils de développement habituels ne répondent pas à ce genre de problématiques. Le fait que le client dispose d’un shell a permis d’avoir une option pour résoudre la problématique. Et donner les bons éléments de contexte à ChatGPT lui a permis d’écrire un script qui a résolu le problème directement.

Je pense avoir passé une demi-heure pour apporter une solution, entre le moment où j’ai retravaillé la liste de référence et le moment où j’ai lancé le script généré par ChatGPT pour récupérer les fichiers manquants.

Sans intelligence artificielle, je ne sais pas si j’aurai envisagé PowerShell qui est un langage de shell qui m’est totalement inconnu, là où le shell Linux est beaucoup plus naturel pour moi (même si je ne l’utilise quasiment jamais, il est quand même constitué de commandes que j’utilise plus fréquemment).

Une image contenant texte, capture d’écran, logiciel

Description générée automatiquement