**Semaine #4**

Pour clore ce mois d'explorations, nous allons nous concentrer sur quelques ingrédients clés pour améliorer notre performance et notre bien-être. Nous commencerons par des stratégies d'amélioration continue et d'automatisation pour affiner notre efficacité. Nous verrons comment le travail hybride peut optimiser notre productivité et découvrirons comment les IA peuvent nous aider à devenir des développeurs augmentés. Nous terminerons en abordant l'importance de prendre soin de soi pour aller plus loin dans notre parcours professionnel.

**Jour #13 - Améliorer en continu et automatiser**

Je trouve essentiel de réfléchir à des actions d’amélioration continue et d’automatisation. Mon objectif est de m'assurer qu'un problème ne se répète pas et d’éliminer les tâches récurrentes à faible valeur ajoutée. L’idée est de creuser, de chercher les problèmes sous-jacents afin d’éliminer ces problèmes qui sont une perte de temps, optimiser son temps et éviter le gaspillage.

**Chez Scrum, on initie une réflexion sur la qualité au moment de la rétrospective**

Chez Scrum il existe une réunion d’équipe qui se déroule à la fin du sprint et qui consiste à faire un point sur les choses qui se sont bien passées, et aussi celles sur lesquelles on a été confronté à des difficultés. L’idée de cette réunion, est de réfléchir aux actions que l’on pourrait mettre en place pour améliorer ce qui s’est moins bien passé. Une fois que l’on a conscience des difficultés, il faut s’intéresser aux actions qui permettent de réduire ou supprimer ces problèmes. Définir des actions n’est que la première étape. Il faut ensuite les réaliser au cours des sprints suivants. Une fois réalisées et mises en place, il faut vérifier l’impact de ses actions et voir si elles ont porté leurs fruits ou non. Si ce n’est pas le cas il faut se reposer la question si l’on estime que cela en vaut la peine, de poursuivre ces actions, les ajuster ou les arrêter.

**Dans le domaine de la qualité logicielle, on utilise la roue de Deming**

Une image contenant texte, diagramme, capture d’écran, cercle

Description générée automatiquement

Il existe un concept connu sous le nom de roue de Deming visant à améliorer la qualité en continu en se basant sur son expérience.

L’idée est relativement simple. Il s’agit de prendre un point de blocage, quelque chose qui cause une frustration quotidienne et décider de la supprimer en effectuant des actions.

* La première étape consiste à planifier les actions à mettre en place pour résoudre notre problème
* La deuxième à réaliser le point d’action qualité
* La troisième à évaluer le résultat de notre action
* La quatrième à agir en fonction de l’évaluation. Le point a-t-il été résolu totalement, partiellement ? L’action a-t-elle eu l’effet attendu ? Peut-on arrêter ? Faut-il persister ? Faut-il changer l’action d’amélioration ?

Et ainsi de suite, jusqu’à améliorer la situation.

**On pourrait faire une analogie avec l’amélioration de la qualité du code pour les développeurs**

On pourrait imaginer qu’une difficulté réside dans un code de mauvaise qualité. Pourquoi ce point pose-t-il un réel problème ? Parce qu’on a du mal à comprendre le code, le maintenir, le faire évoluer ? Pourquoi pourrait-on parler de difficulté ? Parce qu’on a des objectifs d’évolution ? Parce qu’on a des délais ?

* La première étape pourrait être d’analyser le code pour essayer de l’améliorer en réalisant des actions de refactoring
* La phase d’action serait de réaliser du refactoring de notre code
* La phase de vérification consisterait à évaluer si notre action de refactoring rend le code plus facile à comprendre.
* La phase d’action pourrait être de se poser la question si les actions de refactoring sont suffisantes ou s’il faut persister car le code n’est toujours pas assez lisible

**Des actions d’amélioration dans le cadre de mon organisation**

Je prends un peu de temps lorsque je détecte un problème ou une tâche répétitive, afin de voir ce que je pourrais mettre en place pour supprimer cette tâche de faible valeur qui me fait perdre du temps dans mon quotidien.

J’ai par exemple opté pour des tableaux de bord, plutôt que pour des tableaux Excel ou ma simple mémoire pour gérer mon activité. Le problème d'Excel est qu'il est parfois trop lourd et peu adapté à la gestion quotidienne. Et la mémoire nous joue souvent des tours. Il m’arrivait parfois d’oublier un traitement ou de reporter un correctif sur une version de programme, ce n’est pas évident lorsque l’on travaille sur plusieurs projets, et plusieurs développements de fonctionnalités, plusieurs programmes et plusieurs versions de ne jamais rien oublier.

Le problème était de trouver une solution afin de ne rien oublier du traitement d’une fonctionnalité.

La mise en place de tableaux de bord, et de carte pour chaque développement de fonctionnalité m’a permis de résoudre ce problème. Je les trouve plus facile à maintenir qu’un fichier Excel, avec un rendu de meilleure qualité et plus facile à partager. Ils résolvent mon problème initial de perte d’information.

**Cela m’a permis de déterminer quand ma fonctionnalité est réellement terminée**

Le tableau de bord et l’idée de définir des cartes pour le développement de fonctionnalité était un premier pas, mais je me suis aperçu que ce n’était pas suffisant. J’avais besoin de détailler plus finement les différentes étapes qui me permettaient de valider l’ensemble des éléments à mettre en place.

D’où l’idée d’ajouter à mes cartes un Definition of Done (en français on pourrait dire une liste de traitements que je dois réaliser pour m’assurer que ma fonctionnalité a été réalisée entièrement) avec comme élément final la mise en production et la validation dans l’environnement de mes utilisateurs.

Mon Definition of Done contient des actions d’analyse, de réalisation de documentation, de développement, d’archivage et de mise à disposition. Elle est flexible en fonction des fonctionnalités que je traite.

Ainsi chaque fois que je dois développer une fonctionnalité, je crée une carte, qui vient s’intégrer à un flux de traitement, que je fais évoluer, et pour laquelle je suis capable à tout moment de partager à tout moment son avancement.

**J’utilise des idées d’amélioration qualité pour l’ensemble de problèmes récurrents**

Cela peut être très simple grâce à la mise en place de favoris, parce que je n’ai plus envie perdre du temps à de rechercher des liens, ou la mise en place de système de gestion de mots de passe pour éviter de passer mon temps à les réinitialiser.

Je le mets en place dans le traitement de résolution de tickets pour lesquels les actions précédentes n’avaient pas permis de mettre fin à la problématique. Il s’agit dans ce cas de chercher le problème sous-jacent pour assurer une solution durable, en utilisant des méthodes telles que les cinq pourquoi qui est une approche d’analyse des causes profondes.

Je la mets en place en automatisant des actions de sauvegarde de mes environnements, de mes favoris, de mes sauvegardes ou de mes fichiers de configuration pour ne pas prendre le risque de perdre mon travail et passer du temps à faire des récupérations manuelles fastidieuses et peu ludiques.

**Conclusion**

A mon sens la recherche d’actions d’amélioration qualité est un des outils les plus puissants pour optimiser son temps. Elle permet de supprimer les petites actions chronophages du quotidien, telles que rechercher des liens que l’on utilise fréquemment, sauvegarder automatiquement des données essentielles à notre travail, pérenniser les solutions sur lesquelles je travaille. Toutes ces actions d’amélioration permettent de dégager du temps pour des actions qui ont une plus grande valeur.

Je fais régulièrement de la veille pour découvrir de nouvelles solutions, ce qui me permet de trouver des outils et méthodes de plus en plus efficaces. C’est ainsi que l’amélioration continue devient un cercle vertueux dans mon travail.