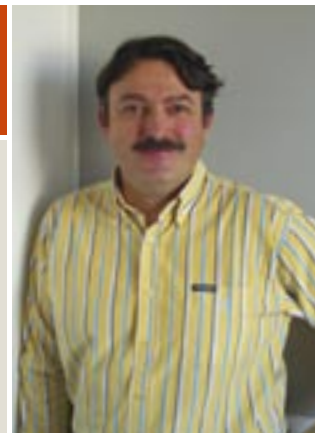


Alexandre Boutin, directeur de la start up AgiletoYou, spécialisée en formation et coaching Lean et Agile (et ex-responsable de la stratégie de développement logiciel pour Yahoo International)



Planifier moins pour produire plus !

Dans le monde du développement logiciel, il est d'usage de considérer qu'une personne atteint le maximum de sa productivité lorsqu'elle produit en permanence. Afin d'obtenir ce résultat, de nombreux managers choisissent d'optimiser la charge de chaque employé à 100 % ou plus. Honnêtement persuadés que cette pratique leur garantira l'absence de « temps mort », synonyme pour eux de perte de productivité.

Bien que cette approche semble frappée du bon sens, elle s'avère totalement remise en cause par certains principes prouvés du Lean^(*) et par d'autres méthodes mathématiques incontestables. Sur la base de ces éléments, il convient donc de mettre fin à cette pratique et de comprendre que pour atteindre l'optimum de productivité, il est judicieux de charger moins les personnes ou les équipes de réalisation.

Prendre le temps d'analyser son environnement

La métaphore que j'utilise pour illustrer ce propos est celle de l'autoroute. Si vous demandez à quelqu'un ce qui se passe lorsque le nombre de voitures atteint la capacité maximale de l'autoroute, il vous répondra naturellement : il y a création d'un bouchon ! Or ce bouchon n'est rien d'autre que la matérialisation physique d'une productivité faible. Chaque voiture est capable de rouler à 130 km/h mais, dans les faits, elle ne roule qu'à 30 km/h.

De même que pour l'autoroute, il est nécessaire de garder des « temps libres » à l'équipe de développement afin de garantir une productivité maximale.

Ainsi, lorsqu'un développeur continue à coder alors que le programme de compilation de son code (plate-forme de build) lui renvoie des erreurs système, il ne fait que rajouter de la complexité et des fautes dans son système. Il sera ensuite beaucoup moins performant car il passera une grande partie de son temps à corriger ces problèmes complexes.

« Il est nécessaire d'octroyer des " temps libres " aux développeurs afin de garantir une productivité maximale »

Logiquement, ce développeur devrait donc s'arrêter de coder, trouver ce qui pose problème, puis remettre en état de marche sa plate-forme de build. Pour être plus performant, il doit prendre le temps d'analyser son environnement et identifier des actions d'amélioration. L'équipe technique, qui peut régulièrement « lever le nez du guidon », trouvera donc plus facilement des solutions (outils, processus ...) qui amélioreront sa performance sur le moyen et long terme.

Réagir face aux urgences

De plus, toutes les équipes techniques sont confrontées à des urgences, qu'il convient de traiter immédiatement. Si aucun « temps libre » n'a été planifié, ces urgences viendront perturber le fonctionnement de l'équipe, laquelle,

pour tenir le délai, ne trouvera pas d'autres solutions que de dégrader les autres activités prévues (réduction de la qualité du code, architecture non robuste, tests non réalisés...). Ces décisions forcées réduiront d'autant la performance ultérieure qu'elles auront été prises dans l'urgence.

Enfin, plusieurs études ont prouvé qu'une équipe technique est plus performante lorsqu'elle travaille sur une seule activité à la fois. En procédant ainsi, elle évite les changements de contexte, qui consomment environ quarante-cinq minutes à chaque fois. Avec un planning rempli à 100 %, voire plus, l'équipe va essayer de tout faire en même temps, et changer de contexte en permanence, pour au final une perte sèche de plusieurs heures de productivité par jour.

Éviter l'effet bouchon

Pour être plus productif, il est possible de migrer vers des approches Lean pour le logiciel (Kanban...) et/ou des méthodes agiles (Scrum, XP...) car elles intègrent structurellement ces pratiques, alors mises en œuvre sans effort particulier. Pour les managers qui appliquent des méthodes plus traditionnelles comme le cycle en V, l'alternative consiste à planifier une occupation des équipes à seulement 80 %... pour produire plus sur le moyen et long terme ! Quel individu n'a jamais ressenti en fin de semaine, et sans savoir vraiment l'expliquer, la sensation d'avoir été peu productif alors qu'il avait énormément travaillé ? La réponse est toute simple : il se trouvait dans un bouchon. ■

(*) Le Lean pour le logiciel est une adaptation du système de production de Toyota, baptisé Thinking ou Lean Production System. Il propose une approche différente et souvent contre-intuitive pour la production industrielle.