sans dev ops :

developpeur (dev) collectent les exigences ensuite developpe le code, test le code puis livre l'application

a l'equipe de prod et explitation (ops operationals) met l'app en exploitation puis en production

pose des pb car deux equipe, dev + ops on des objectif diff:

1 coté, les dev doit faire evoluer rapidement les app pour repondre au demande du marché, c'est la recherche d'un time to market(delai entre idée initial et concretisatione en prod, time to market rapide permet a une entreprise de repondre plus vite au marché et a la concurrence : panne prod peut couter bcp a l'entreprise)

les dev et les ops sont opposés car si il y en a un qui est en retard l'autre équipe est ralenti

devops pour lier les objectifs : objectifs a court termes différent vers objectif commum sur le long terme(creation de valeur pour l'entreprise)

time to market plus rapide, produit de qualité et equipes plus efficace

5 grand principe : culture de la collaboration

l'automatisation

creation de valeur comme objectif

mise en place de métriques à tout les niveaux

partage comme valeur forte

donner une vision globale du systeme d'information

but ultime de fluidifié et de faciliter l'inter communication entre devs et ops

la recherche systematique de valeur ajoutée pour le client se décline a travers la mise en place de cucle d'amélioration continue et donc la suppression des taches qui n'apportent aucune valeur mais aussi a travers le rapprochement des différentes equipes en les incitant a se concentrer sur la valeur business et sr la satisfaction des utilisateurs.

le pricipe de partage : obectif commum au deux equipes

partage dans le travail quotidien: partager problemes, connaissances, et retours d'experience ce qui fait emerger un sentiment d'empathie et d'entraide mais aussi des nouvelles idées

la mise en place d emetriques est necessaire car on peut parler d'amelioration que si l'on est capable de mesurer l'etat initial , final et evolution entre les deux.

donc metriques metier pour mesurer la vitalité du business

metriques d'equipe pour performance des equipes

mais aussi mesure satisfaction clients pour améliorer les produits

**automatisation des procesus : 3 etapes : provisionning automatisé pour les dev,tests automatisés pour la qualité attendue, déploiement automatisé**

**outils : docker : provisionning auto d’envitonnement**

**selenium : test automatique**

**puppet : deploiement applicatif**

**automatisation de ses 3 processus va donc vers un deploiement continu**

c’est un approche qui considere qu’une appli doit etre construite de telle manière a ce quelle puisse etre envoyé en prod a tout moment c’est-à-dire qu’a partir du moment ou les applications reussissents les test d’acceptation automatisées elle doit etre automatiquement deploiyer en production

l’idée cest d’avoir un rythme de mise en production plus regulier, plus rapide et plus petit, ce qui est bcp moins risqué qu’une grosse mise a jour en mode big bang

acceleration est donc une acceleration du time to market, avec ce cycle de livraison plus rapide et plus regulier, il est facile pour l’entreprise d’avoir des retours des clients sur le produit livré et en exploitant ses donées il est plus facile d’abiutir par uitérations au produit qui sera le plus pres du besoin du marché, a chaque livraison, le produit repond d eplus en plus aux attentes des clients ce qui permet a l’entreprise d’avoir des produits d equalité et qui correspondent aux besoins des clients

donc entreprise devops : dev collectent les besoins métiers et deveoppe un nouveau code de l’appli pour repondre a ses besoins, l’app est automatiquement compilée et testee a diffenrents niveaux et si le code attent le niveaux d’exigences demandees en terme de qualité l’evolution est livrée en production auto et rapidement

le travail des ops est quant a lui facilité car les dev on compris les contraintes de l’equipe ops et ont integres ces problematiques d’exploitation et de production dans leur code

par exemple le dev a integre dans la conception d’application des metriques d’exploitation permettant desurveiller l’etat de snaté de l’application les equipes opérationnelles comprennent mieux les besoins des developpeurs, ce qui permet une meilleur optimisation des ressources de production , meilleur monitoring et une meilleure remontée d’erreurs

<https://www.youtube.com/watch?v=M6F6GWcGxLQ>

Tu ne m'en voudras pas, je corrige ici 2 erreurs de formulation de ton texte : **"Vous l'avez compris, l'objectif du déploiement continu est" la suppression des postes des "opérationnels". (...) "Le travail des ops est quant à lui" supprimé, et il ne leur reste plus qu'à devenir des devs (bon courage !), ou à pointer au pôle emploi. Ou à se suicider, ce qui arrangerait tout le monde, hein ?**

**stating : test automatisé**

**feeback : voir si ca marche bien ... sur serveur...**

**feedback client : ticket support et tu reviens debut**

[**https://www.youtube.com/user/GoAtlassian/videos**](https://www.youtube.com/user/GoAtlassian/videos)

**burnUP**