



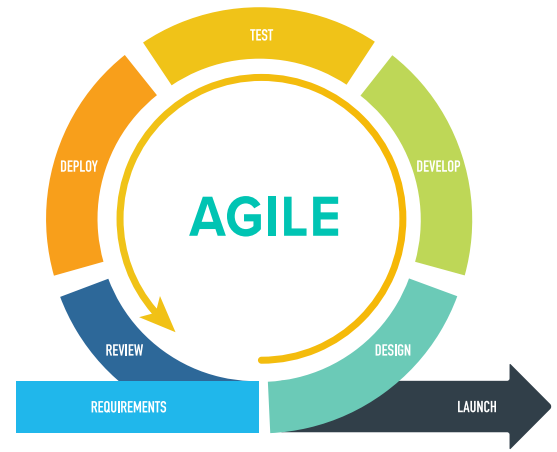
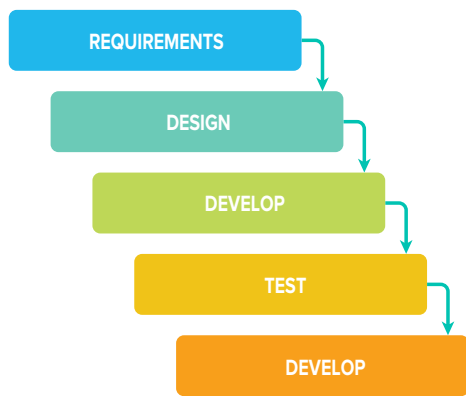
```
[cpu] system overload // dfc 330000x // stable
[cpu] done // code URxxx00xx
[cpu] done / done / password found / operation 120 227
[cpu] negative / negative / [not found] / operation 2289 99478
[cpu] done / negative / error 003378
[cpu] error / error / restart
[cpu] done / done / access / complete / operation 122 334
[cpu] error
[cpu] negative / analyzing / operation 552 390
[cpu] preparation complete / code xxx000x0xx0xx0
```

PROTOCOL RF --- accepted --- code 000xx0000 ---> stop --- order 7

4 façons d'organiser l'automatisation des tests avec le développement agile

Améliorer la qualité des produits en veillant à ce que les tests automatisés suivent les cycles de développement rapides.

WATERFALL



Passer de la méthode en Cascade, ou « waterfall » en anglais, à la méthode Agile nécessite de transformer votre approche. Il peut être difficile de tenir le rythme de tests sur des cycles plus courts, et recourir à l'automatisation des tests sur l'itération en cours peut ralentir le développement et la mise en production. Malheureusement, il n'existe pas de formule magique pour intégrer l'automatisation des tests dans chaque processus Agile, mais plusieurs facteurs peuvent aider.

1. Traduisez les nouvelles fonctionnalités de l'application en objectifs de test.

L'objectif de chaque sprint Agile est d'informer rapidement, de contrôler et de vérifier les nouvelles fonctionnalités et/ou caractéristiques. Les automatiseurs doivent être capables de comprendre ces objectifs tout en étant réactifs.

2. Adoptez d'autres pratiques modernes de développement et de test.

Des techniques comme le Behavior Driven Development (BDD) et des outils comme Cucumber et Gherkin peuvent rendre l'automatisation des tests plus efficace en fusionnant les exigences fonctionnelles aux résultats des tests.

3. Unifiez votre approche de l'automatisation des tests au sein des équipes de sprint.

Les automatiseurs devraient pouvoir communiquer et adapter leurs pratiques. Envisagez de renforcer vos sprints en appliquant nos méthodes, pour synchroniser le travail des automatiseurs et le partage des scripts et réduire les effets de répétition.

4. Promouvez une communication ouverte entre les équipes de test et d'exploitation.

Les opérations peuvent réutiliser les informations partagées par les automatiseurs de test pour surveiller plus efficacement la stabilité, la fonctionnalité et les performances de l'application dans les environnements réels.

Optimisez votre processus de test pour améliorer l'expérience utilisateur de vos services digitaux. Visitez notre site web.