

Tests et Développements



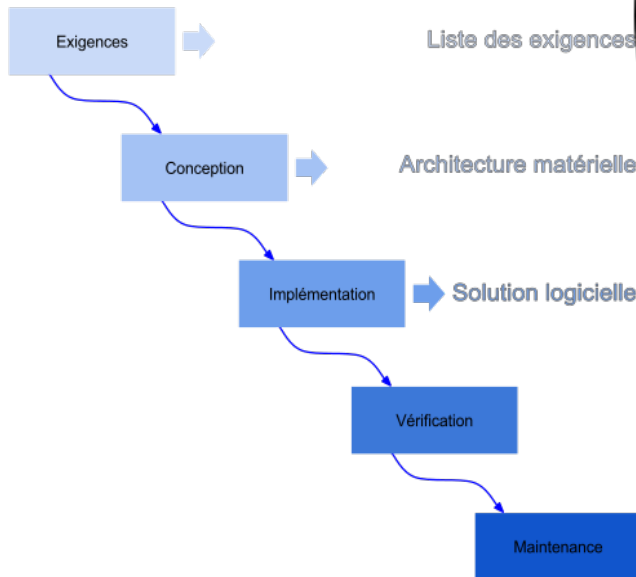
Cycles du développement

Qu'est ce qu'un **test** ?

TDD : Test Driven Development

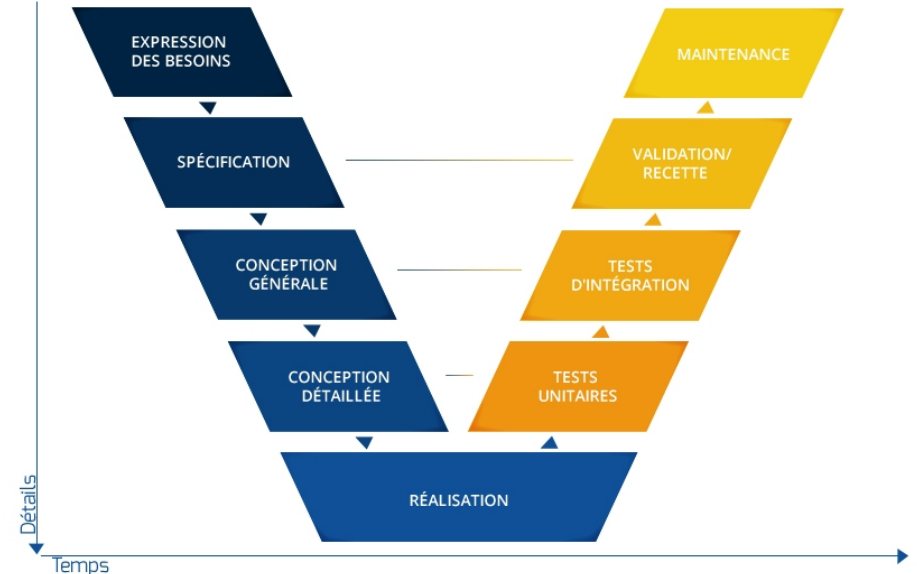
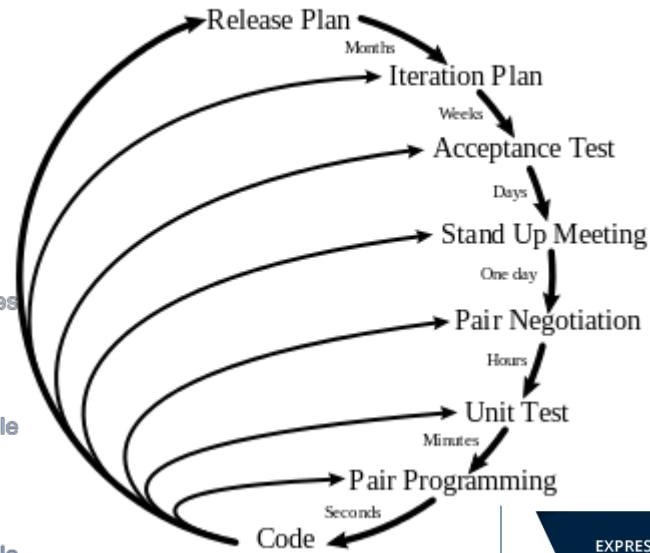
Tests et Développements

Cycles:



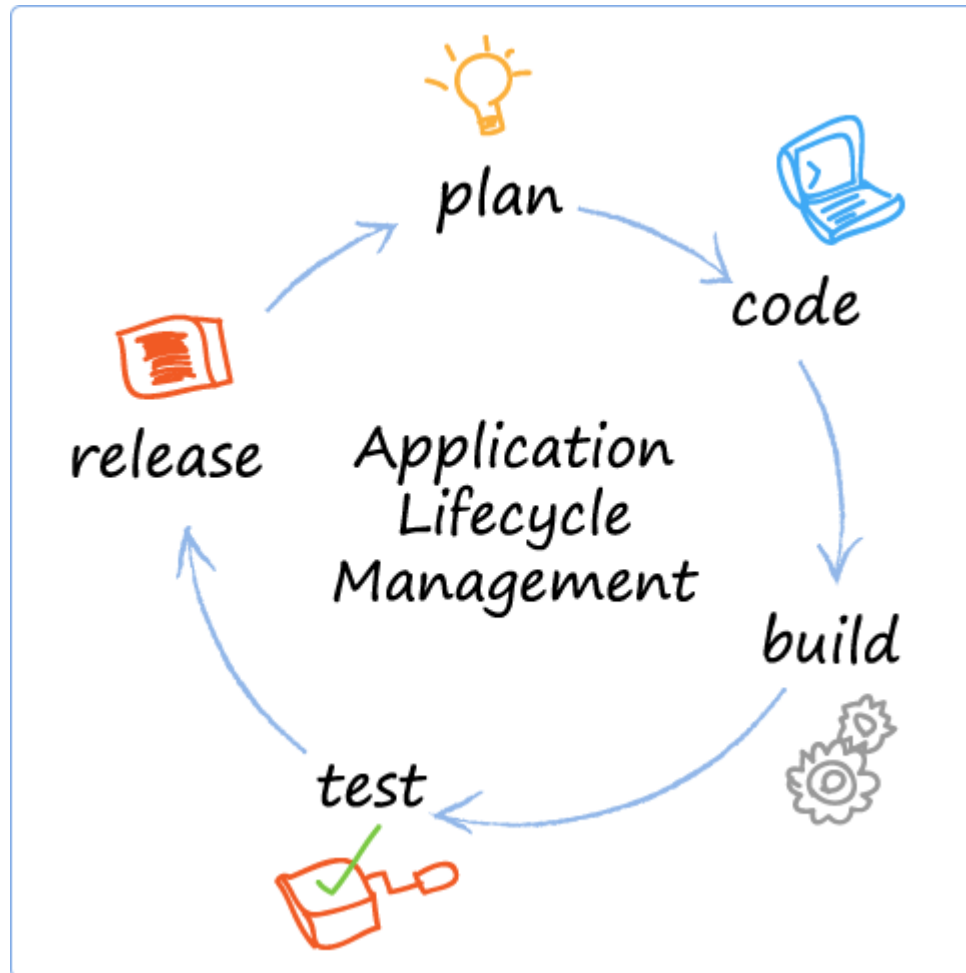
wikipedia

Planning/Feedback Loops



Tests et Développements

Cycle de vie d'un code:



Tests et Développements

Résoudre les problèmes

le plus tôt

- plus complexe en prod
- mauvaise image
- commit plus serein

THE RELATIVE COST OF FIXING DEFECTS

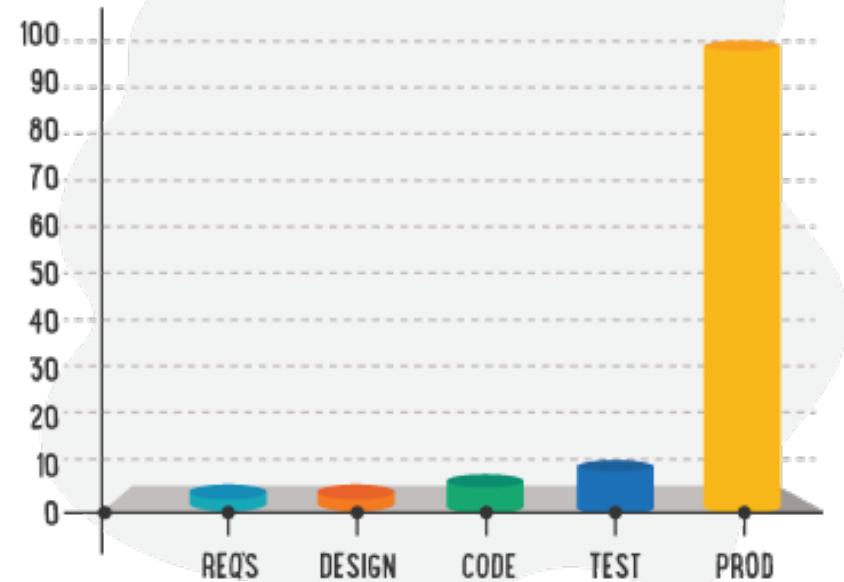


ILLUSTRATION BY SEGUE TECHNOLOGIES

Tests et Développements

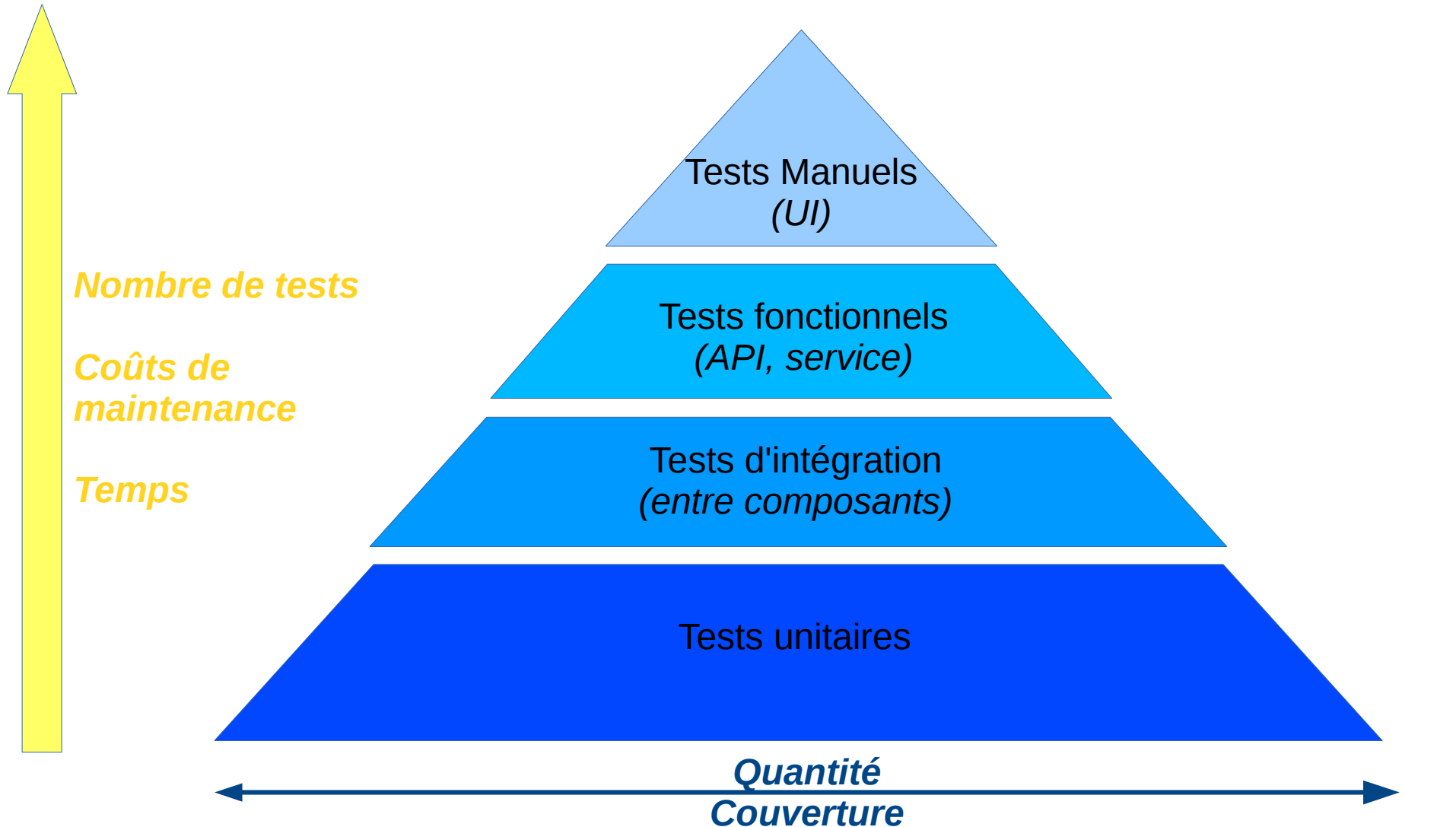
- Oblige une prise de recul avant de coder
- Etude d'IBM sur l'impact de la productivité
 - Temps de développement **plus long** : 15-30%
 - **Moins de bugs** : 40-90%
- Assurance de **non-regression**

Types de tests



- Nouvelle fonctionnalité, cas simples reproductibles
 - tests automatisés : unitaires, d'intégration
- Tests manuels (exploratoires) : charge, robustesse, sécurité

Types de tests



Test unitaire

- Chaque méthode/fonction a son test
 - Vérifier l'action effectuée selon divers paramètres d'entrées



Test unitaire

- Couverture maximum :
 - Tester chaque cas particulier individuellement
 - Gestion des erreurs



Green = class in focus
Yellow = dependencies
Grey = other unrelated classes

CLASS IN UNIT TEST



Green = class in focus
Yellow = mocks for the unit test

Outils pour les tests

EASYMOCK



JUnit



googletest
Google C++ Testing Framework



PHPUnit



CPPUnit



Portail



https://en.wikipedia.org/wiki/Portal:Software_testing

TDD



- Test Driven Development :
 1. Ecrire **UN test** et sa doc :

quel cas particulier ? qu'est-ce qu'on attend ?
 2. Ecrire **le code** qui correspond pour que le test passe
 3. Réorganiser l'ensemble des tests et le code
+ recommencer

TDD



- Un fichier/classe **dédié au test** (ex : CalculTest)
 - Une fonction (ex : testSomme) par fonction dans le code
- **TestSuite** : Ensemble des tests combinés
- Exemple :

```
assert.typeOf(foo, 'string');  
assert.equal(foo, 'bar');  
assert.lengthOf(foo, 3)  
assert.property(tea, 'flavors');  
assert.lengthOf(tea.flavors, 3);
```