

Software Architecture

Tests

**Ecrire des tests est une compétence
au même niveau que le
développement**

Ce n'est pas facile et ça s'apprend

Pourquoi on se fait ch*er à écrire des tests ?

Pour limiter l'erreur humaine

**Notre cerveau est trop petit pour
modéliser l'intégralité d'un
programme complexe**

Quoi tester ?

3 niveaux de tests

3 niveaux de tests

Niveau	Quels tests ?
UI	Les évènements click, checkbox, etc
Métier	Les règles de gestion
Protocole	le codage / décodage "bas niveau"

3 niveaux de tests

Niveau	Quels tests ?
UI	Les évènements click, checkbox, etc
Métier	Les règles de gestion
Protocole	le codage / décodage "bas niveau"

Ressemble à une logique mathématique

$$f(x) = y$$

Facile à tester

 **Si bien isolé de l'UI**

3 niveaux de tests

Niveau	Quels tests ?
UI	Les évènements click, checkbox, etc
Métier	Les règles de gestion
Protocole	le codage / décodage "bas niveau"

Toute la logique de vos écrans

“ J'active ce bouton dans la condition X ”

“ J'affiche cette view si l'utilisateur à le droit ”

3 niveaux de tests

Niveau	Quels tests ?
UI	Les évènements click, checkbox, etc
Métier	Les règles de gestion
Protocole	le codage / décodage "bas niveau"

Le moins critique

**Ecrire des tests,
c'est écrire du code**

Conséquence

Mettre à jour son code

Force à modifier ses tests

Le code UI
est le plus sujet à être modifié

Si nécessaire,
Commencez par tester un chemin critique

Elargissez si besoin

La couverture de test à 100%
tiens plus du trophée que de l'utilité
(C'est mon opinion, pas un fait absolu !)

**Le code le plus maintenable,
est celui qui n'existe pas**

**Le test est un outil,
pas un objectif en soi**