

Tests unitaires, fonctionnels et d'intégration

Mickaël Andrieu - SolvoLabs
mickael.andrieu@solvolabs.com

Qu'est-ce qu'un test ?

Un test valide un comportement

Un test confirme l'absence
de bugs.



Le risque 0 n'existe pas.*

* C'est pas une raison pour coder comme des gorets!

Et on ne peut pas tout
tester.

Un test **confirme** que ce qui
a été développé correspond
au **comportement attendu**.

Types de tests

1. Tests unitaires
2. Tests fonctionnels
3. Tests d'intégration

Tests unitaires

Le test unitaire est une procédure permettant de vérifier le bon fonctionnement d'une partie précise d'un logiciel ou d'un programme (dit aussi "composant").

Tests unitaires - en bref

- Trouvent les erreurs rapidement
- Sécurisent la maintenance
- Documentent le code
- Parfois impossible à écrire...

Tests unitaires - Démo

L'exemple d'une calculatrice en PHP avec PHPUnit

Tests fonctionnels

Le test fonctionnel est un service d'assurance qualité qui consiste à s'assurer qu'un système fonctionne adéquatement.

Tests fonctionnels - en bref

- Détectent des erreurs non révélées par les TU
- Sécurisent un comportement attendu
- Plutôt rapides

Tests fonctionnels - Démo

Tests fonctionnels sur un site web avec CasperJS

Tests d'intégration

Le test d'intégration valide un **scénario utilisateur** complet. Il confirme en tout point qu'une action utilisateur est **réalisable sans erreur**.

Les tests d'intégration sont aussi parfois appelés "tests fonctionnels", inutile de s'énerver en société :-)

Tests d'intégration - en bref

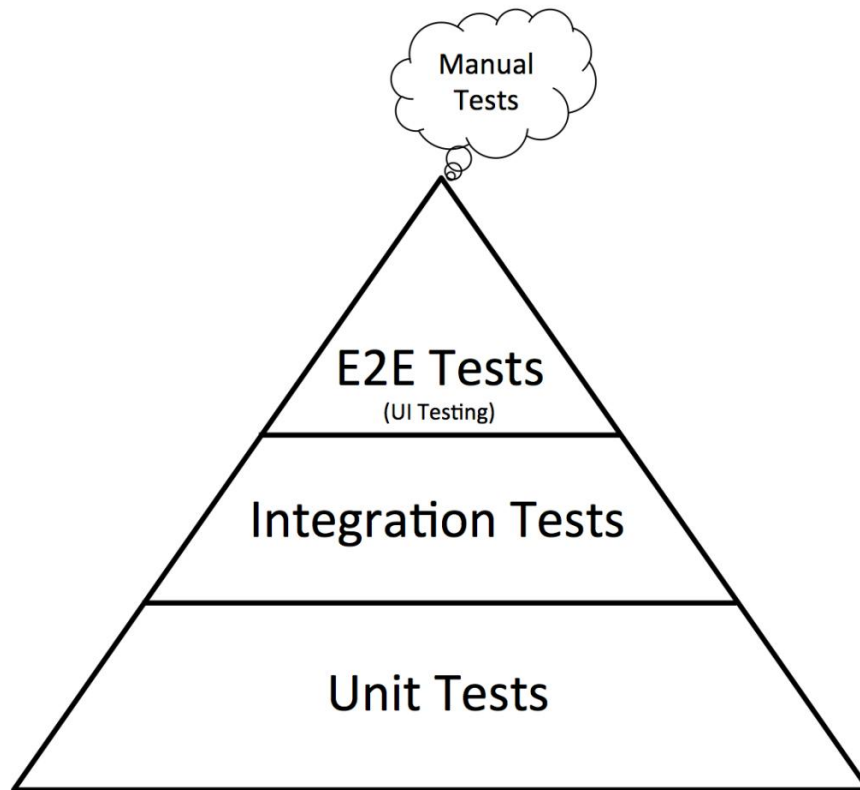
- Rassure l'équipe "produit" sur le comportement validé
- Très proche du comportement "réel"
- Super facile à mettre en place
- Très (trop?) lents

Tests d'intégration - Démo

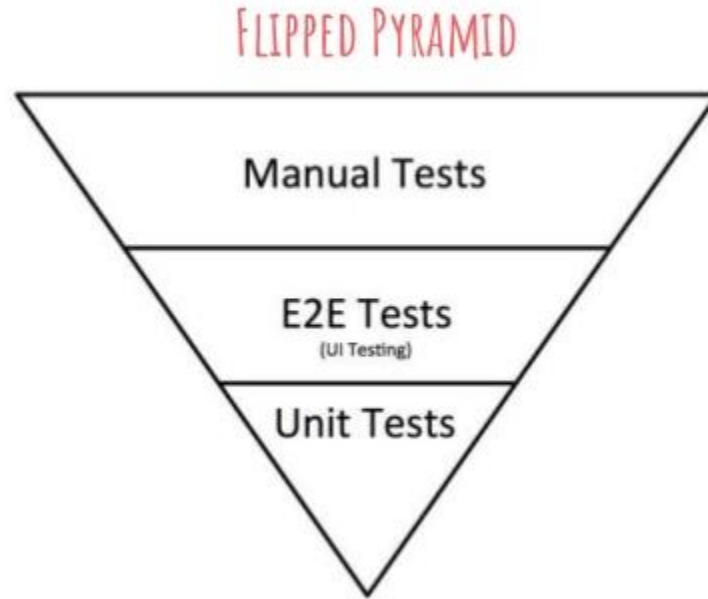
Tests de survie réalisés sur un site web avec CasperJs

Des tests oui, mais en
quelle proportion?

La pyramide de tests - cas idéal



La pyramide de tests - cas plus “réel”

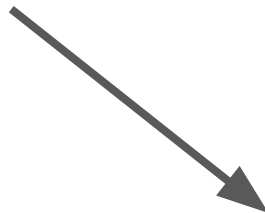


La pyramide de tests - Stratégies

- Si aucun test et beaucoup de code, commencez par quelques tests d'intégration pour sécuriser les principaux scénarios utilisateur/client;
- Si nouveau projet, un maximum de tests unitaires, ce qui ne peut être validé en unitaire en fonctionnel, et en cas de bug client test d'intégration pour éviter la régression;
- Si tests déjà existants, auditer les tests d'intégration et fonctionnels qui pourraient être validés par des tests unitaires;

Questions?

Questions?



Mickaël Andrieu - SolvoLabs
mickael.andrieu@solvolabs.com