

# Les tests

## Problématique

Pendant la réalisation de notre mandat, nous travaillons dans notre environnement de travail que nous connaissons bien. Petit à petit, nous construisons un produit qui satisfait les exigences de notre client. Comme nous sommes impatients de livrer le fruit de notre travail (ou alors qu'on est en retard), nous prenons parfois, plus ou moins consciemment, des raccourcis qui nous amènent plus vite au but. Pire que cela, lorsque le moment est venu de vérifier le fonctionnement de ce qu'on a fait, nous avons tendance à favoriser les tests que l'on sait qu'ils vont être couronnés de succès (c'est le [biais de confirmation](#)).

Le moment venu de mettre en service ou de publier notre système, nous mettons notre produit dans un nouvel environnement et entre les mains d'un autre utilisateur qui ne se comporte pas exactement comme nous.

Et soudain les problèmes surgissent...

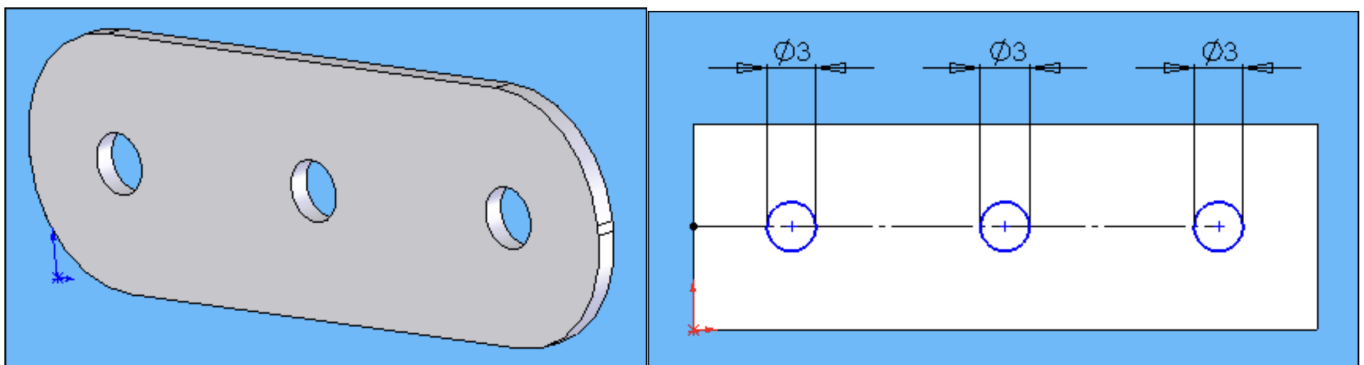
Comment faire pour minimiser, voire éliminer ces problèmes ? En portant une attention toute particulière à la manière de mener nos tests.

Commençons tout d'abord par introduire un peu de terminologie.

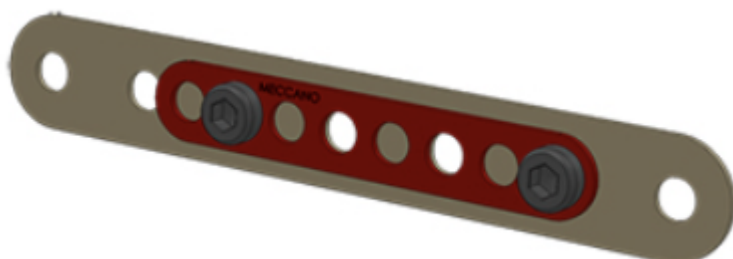
## Niveau de test

Le niveau d'un test est défini par l'environnement dans lequel il est exécuté.

- Le **Test unitaire** : Il vérifie le bon fonctionnement d'un composant réalisé par une personne. Exemple : on vérifie qu'une pièce de Mécano respecte les dimensions voulues



- Le **Test d'intégration** : Il vérifie que deux ou plusieurs composants testés unitairement fonctionnent correctement ensemble. Exemple : on vérifie que diverses pièces de Mécano peuvent bien être assemblées au moyen des vis et écrous.



- Le **Test Système** : Il vérifie le bon fonctionnement du système complet dans l'environnement où il va être vraiment utilisé. Exemple : on vérifie le bon fonctionnement de notre construction en Mécano complète.



## Type de test

Le type d'un test est défini par la caractéristique du produit qu'il vise à vérifier.

Nous nous contenterons ici de citer quelques-uns des types principaux :

- Le **Test fonctionnel**, qui vérifie que l'élément testé fait ce qu'on attend de lui ;
- Le **Test de performance**, qui vérifie que l'élément testé peut traiter la quantité d'information voulue dans le temps voulu ;
- Le **Test de robustesse**, qui vérifie que l'élément testé peut reprendre un fonctionnement normal après avoir rencontré une situation anormale.
- Le **Test de non-régression**, qui vérifie qu'un changement effectué à un endroit du système n'a pas "cassé" d'autres parties.

Exemple : considérons la machine ci-dessous.



Elle sert à imprimer des flyers, les plier, les insérer dans une enveloppe, fermer l'enveloppe et faire des paquets de 100 enveloppes fermées.

- Un test fonctionnel vérifiera que les flyers sont bien pliés en trois volets égaux et qu'il y a bien 100 enveloppes par paquet au final
- Un test de performance vérifiera que la machine est capable de produire 250 paquets de 100 enveloppes en une heure
- Un test de robustesse vérifiera que si on fournit des enveloppes qui sont déjà remplies, le système les écarte et reprend son fonctionnement dès que les enveloppes fournies sont vides
- Un test de non régression vérifiera que le système fonctionne toujours après avoir mis à jour l'OS du PC de contrôle de la machine

## Quels tests faut-il faire ?

Il est rare qu'il y ait des tests qui ne servent à rien. Par contre l'exécution des tests peut prendre un temps et des ressources très importants.

Nous nous trouvons donc rapidement dans la situation où il faut choisir quels sont les types et niveaux de test que l'on aura la possibilité d'exécuter, en fonction de la nature de notre produit et des moyens dont on dispose (en matériel, en personnes, en temps).

L'énoncé de ces choix se fait dans la stratégie de test.