

T.1

Tester le logiciel réalisé

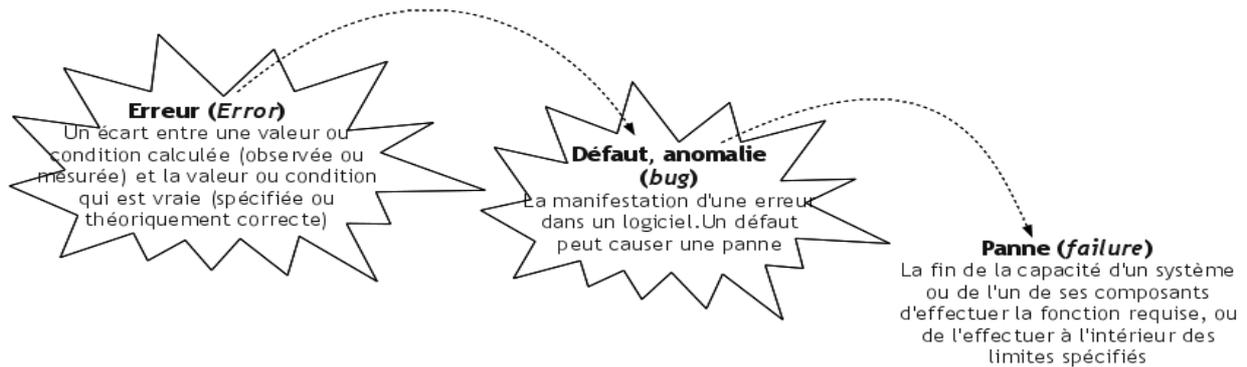
Objectif

Rechercher une anomalie (défaut, appelé souvent bug) dans le comportement d'un logiciel.

Préambule

Principe d'une panne (IEEE 729)

A l'origine, il y a la Faute (mistake) : c'est la cause d'une erreur, puis :

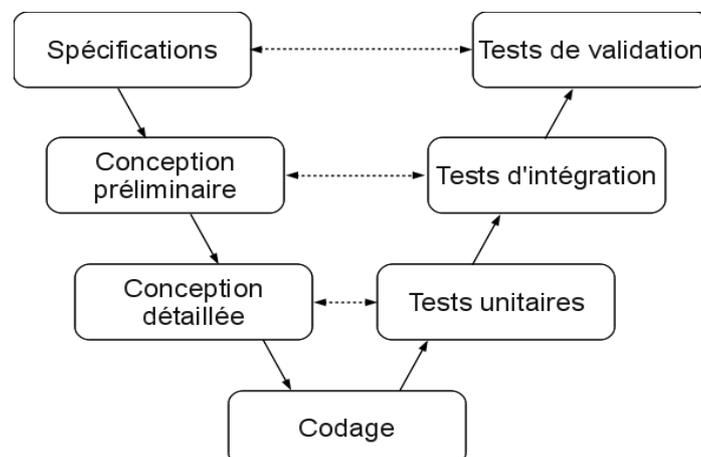


Méthode

Si les tests se déroulent parmi les activités finales du projet, ils se préparent dès le début. Il existe différentes formes de tests :

- Tests de fonctionnements (**unitaires et intégrations**) : pour vérifier que le produit est bien fait (sans anomalies)
- Tests de **validation** : permettent de vérifier que le produit réalisé est le bon
- Tests d'architecture : pour vérifier que le produit est utilisable dans son environnement d'exploitation.

Ces tests, quels qu'ils soient, sont **planifiés à l'avance**, dans l'étape correspondante à leur niveau de détails. Cela signifie qu'il faut, dans chaque étape du processus, spécifier quels tests seront effectués, à quel moment, par qui, dans quelles conditions, avec quels moyens et surtout de quelle manière.



Remarques

Si une batterie de tests ne montre pas de défaut cela n'implique pas que le logiciel est quand même exempt de défaut ...

Le test n'a pas pour objectif : de diagnostiquer la cause des erreurs, de corriger les fautes ou de prouver la correction.

On évalue à environ 40% la part des tests dans le coût d'un logiciel (et plus pour des logiciels critiques).

L'investissement dans les tests a donc un coût très important dans un projet logiciel.

