



es tests agiles : Ma quête des meilleures pratiques !

Qui suis-je ?

Artisan Développeur et Coach Agile.

Je suis persuadé que la technologie n'est rien sans les gens qui la font.

J'aide les équipes et leurs managers à rendre leur travail à la fois plus efficace et plus confortable.

Mes principales forces

Coaching d'équipes



Formation



Facilitation de groupe



Jeux sérieux



Frantz **Degrigny**

frantz.degrigny@conserto.pro



faisons connaissance

**Trouver une pratique qui
a été facteur de qualité 1 min**

Partage avec votre voisin 4 min

**Quels sont les facteurs
de qualité ?**



Mon intention

**Souvent les équipes sont bloquées
par leur mauvaise pratique de test**

**En 10 ans de coaching j'ai vu
le pire et le meilleur**

Partage de mes observations

**Selon 7 aspects de test,
du plus impactant au plus avancé
avec des exemples concrets**

Testeur

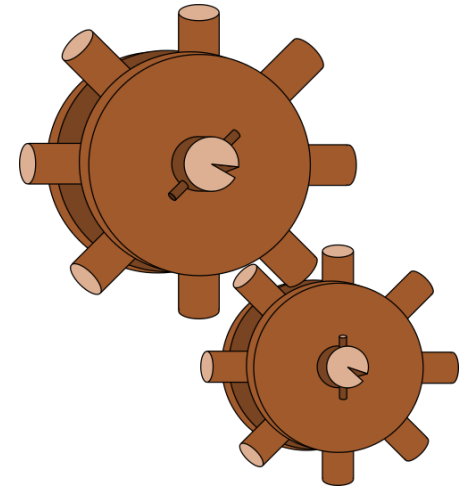


Défaut Bug



I - *Automatisation*

Nécessité d'automatiser tout ce qui peut l'être



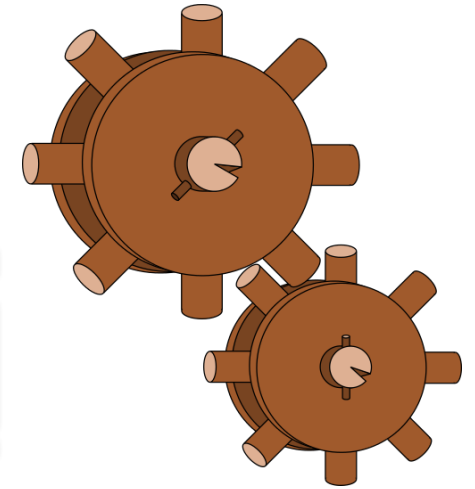
- Non-reg totalement manuelle
- Tirer les tests de la campagne au hasard
- Bug en prod et pourtant le test correspondant est dans notre patrimoine !



- Avec le TDD, couverture > 80%
- Refactor majeur en 2j :
moteur BDD : NoSQL => SGBDR

Tout auto => Full Exploratory Testing
La période de test finale est très confortable !

I - *Automatisation*



Nécessité d'automatiser tout ce qui peut l'être

**Sensibilisation
avec le jeu : CaTesTdrale**



<https://catestdrale.github.io>



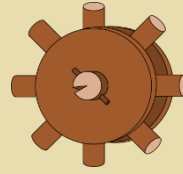
- Non-reg totalement manuelle
- Tirer les tests de la campagne
- Bug en prod et pourtant le test correspondant est dans notre



TDD, couverture > 80%
majeur en 2j :
BDD : NoSQL => SGBDR

> Full Exploratory Testing
de test finale est très confortable !

I *Automatisation*



II *Stratégie de test*

III *Patrimoine de tests*

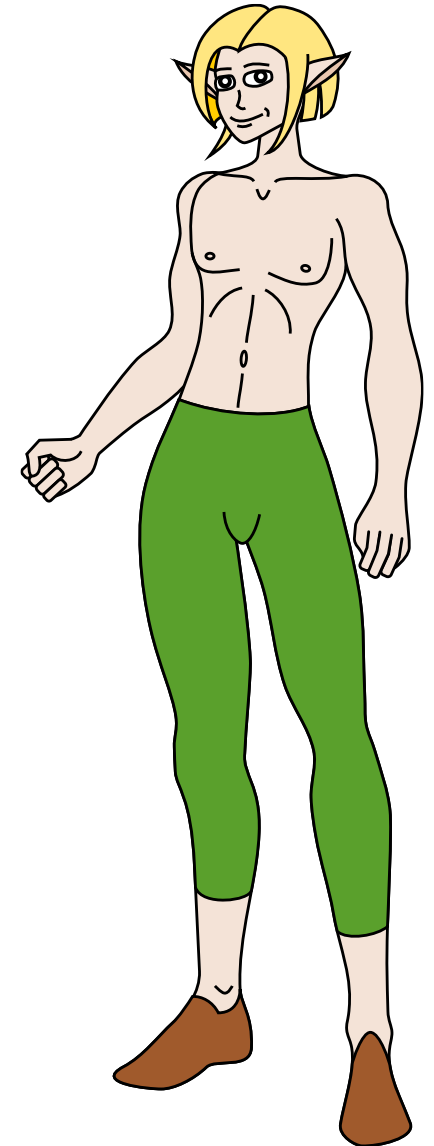
IV *Rôle du testeur*

V *Campagnes de tests*

VI *Qualité des tests*

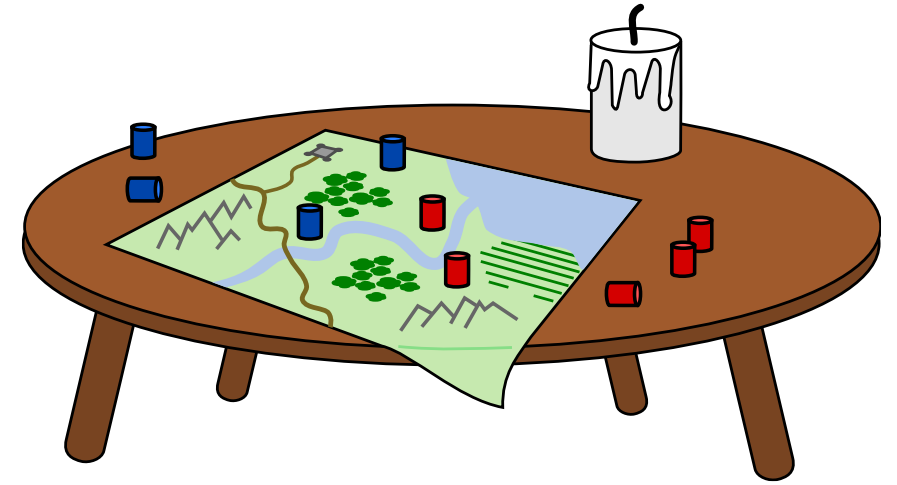
VII *Risque industriel*

Niveau I



II - *Stratégie de test*

partager une stratégie **délibérée**
avec tous les acteurs



- Stratégie de quoi ?
- Certaines choses ne sont jamais testées car personne n'y a pensé

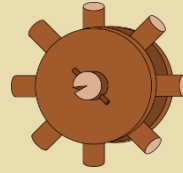


- 1/2 page rédigée en phase de préparation (sprint 0) avec le métier (et revue en rétro)
- combinatoire appareils ?
 - Accessibilité ?

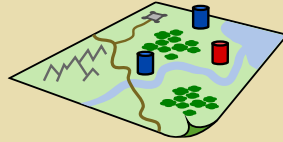
Equilibre : effort / risque

Le risque est accepté collectivement

I *Automatisation*



II *Stratégie de test*



III *Patrimoine de tests*

IV *Rôle du testeur*

V *Campagnes de tests*

VI *Qualité des tests*

VII *Risque industriel*

Niveau II



*Niveau
+1*



III - Patrimoine de tests

un patrimoine de test partagé et **lisible** par tous

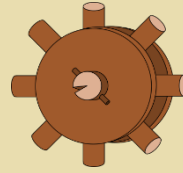


- Tests Unitaires codés par les Dev (Git)
- Tests de non reg manuels par la QA (GED)
- Tests de charge par une équipe dédiée
- Des choses testées plusieurs fois, des choses jamais
- Personne n'a de vue d'ensemble

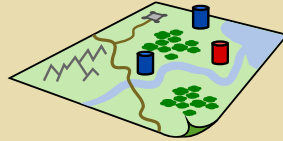


- Gherkin un DSL pour une façon unifiée de rédiger nos scénarios de test
- Les TU lisibles par le métier
- Atelier **Example mapping** 1x par semaine
- Un WIKI customisé pour éditer des fiches Gherkin pour les tests fonctionnels
- Un rapport de test en HTML lisible par tous

I Automatisation



II Stratégie de test



III Patrimoine de tests



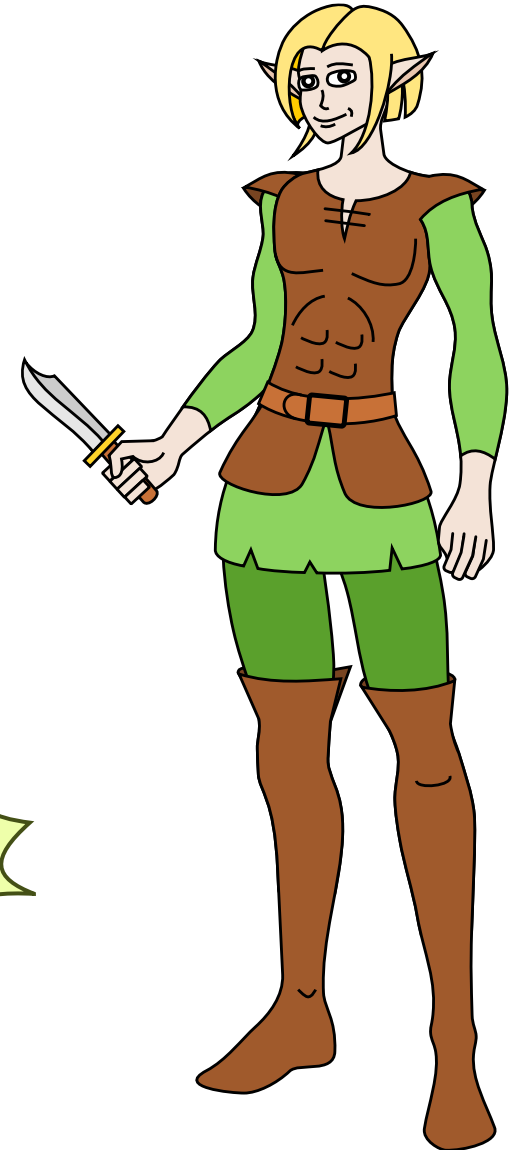
IV *Rôle du testeur*

V *Campagnes de tests*

VI *Qualité des tests*

VII *Risque industriel*

Niveau III

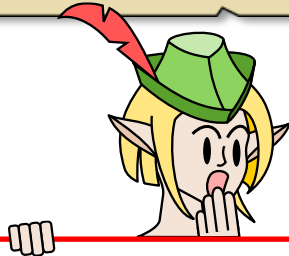


Niveau +1

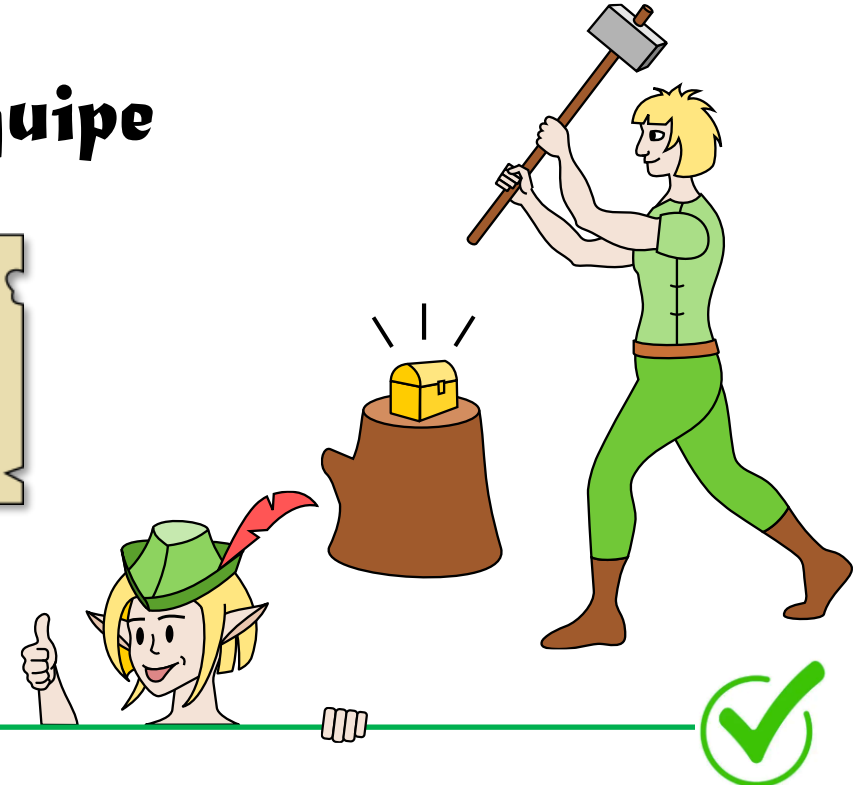


IV - Le rôle du testeur agile dans l'équipe

savoir bien tester est une **compétence**,
on doit l'avoir dans l'équipe



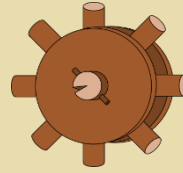
- Des « testeurs » dans un autre service, avec objectifs et rythme de travail différents
- Des tests écrits à partir des fiches fonctionnalités, sans dialogue
- Le PV de recette envoyé au métier sans passer par le PO !



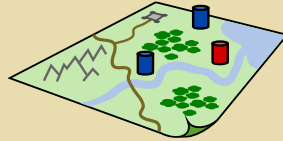
Rôle « testeur agile » coach de l'équipe :

- Garant de la stratégie de test
- Montée en compétence de l'équipe
- Collecter des jeux de données de test réalistes (ajoutées à la DoR)
- Ajout de fonctionnalités pour générer ou gérer des données de test
- Rôle porté par un binôme

I Automatisation



II Stratégie de test



III Patrimoine de tests



IV Rôle du testeur

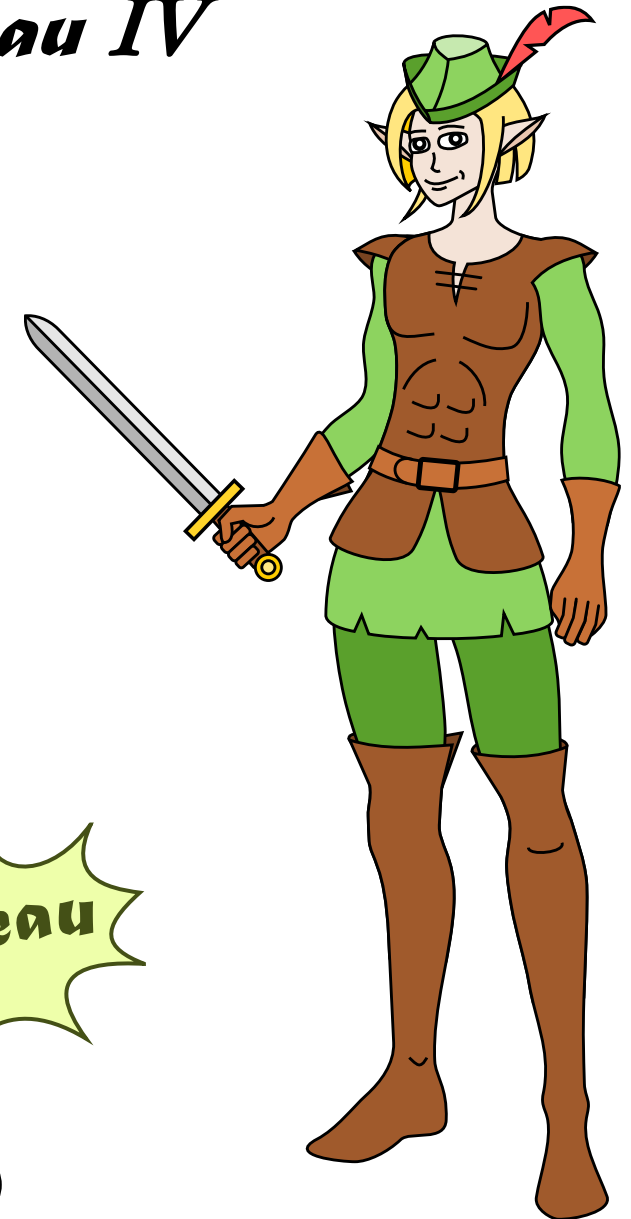


V Campagnes de tests

VI Qualité des tests

VII Risque industriel

Niveau IV



V - Gérer nos campagnes de tests comme des pros

chaque campagne doit être **définie**
et gérée avec **rigueur**



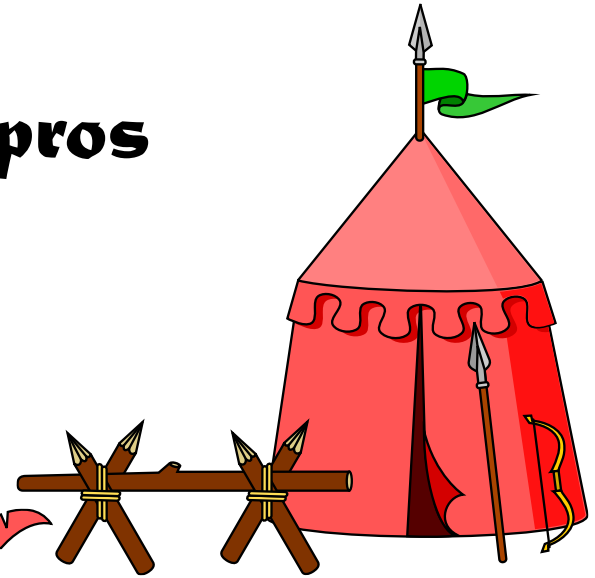
Campagne de quoi ?

On teste à l'arrache les scénarios
les plus évidents, le reste passe à la trappe

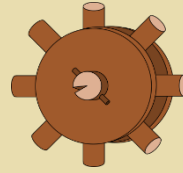


Outil dédié (XRay pour Jira) patrimoine et
campagnes, choisi avec le métier :

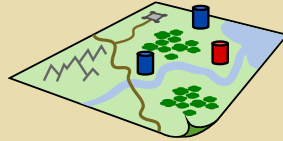
- des rapports de test agrégés (TU, Integ, ...)
- mesure de la couverture de test agrégée
(on voit ce qui n'a jamais été testé)
- métriques sur les tests partagés en Revue



I *Automatisation*



II *Stratégie de test*



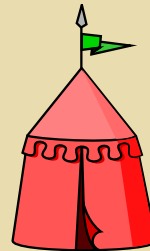
III *Patrimoine de tests*



IV *Rôle du testeur*



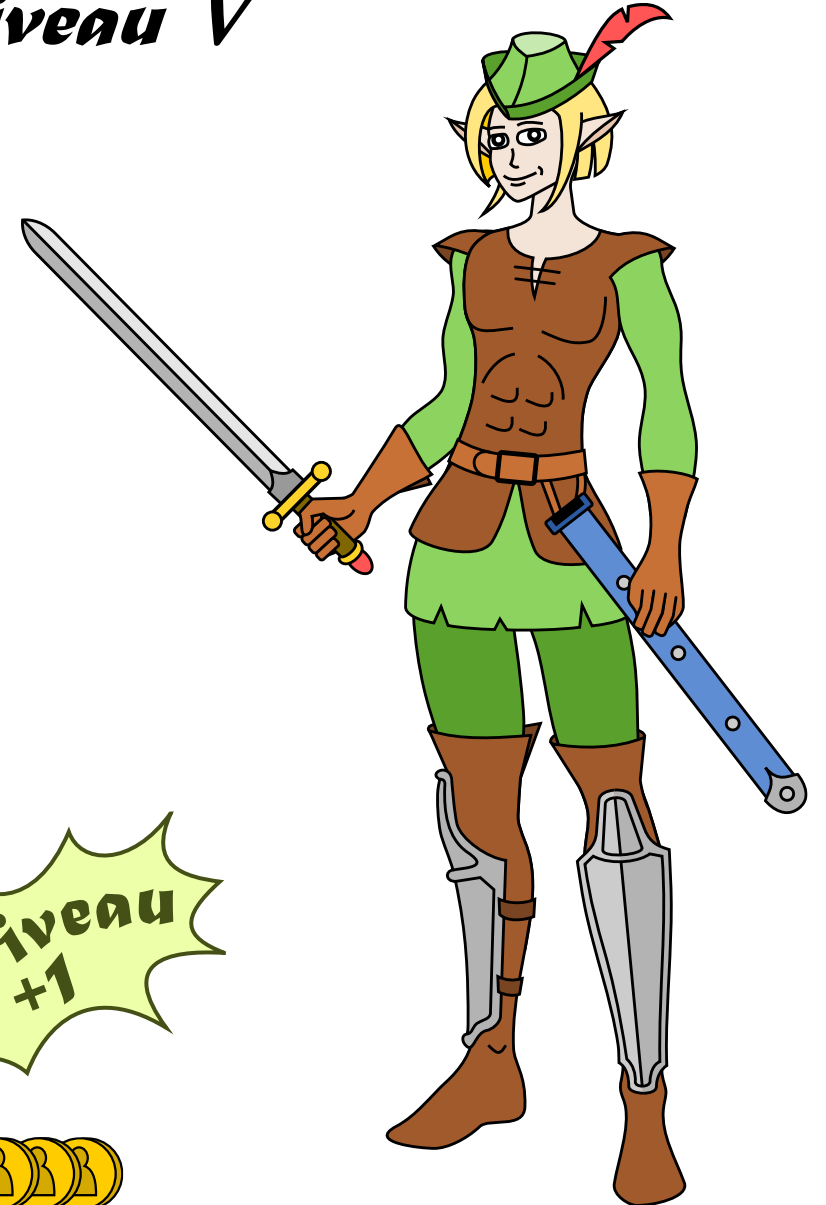
V *Campagnes de tests*



VI *Qualité des tests*

VII *Risque industriel*

Niveau V



*Niveau
+1*



VI - La qualité : sur les tests aussi !

pourquoi la qualité des tests est-elle généralement **plus faible** que sur le code ?



Refactor empêché par les tests auto,
de plus mauvaise qualité que le code !

L'analyse Sonar ne passe pas sur les tests

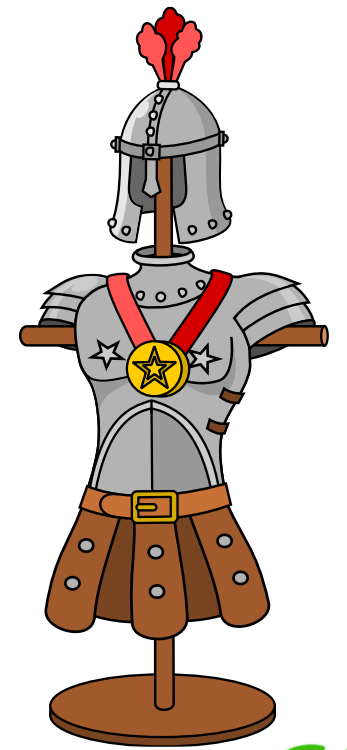
➤ Pourquoi ?



Mutation Testing pour vérifier la qualité de
nos tests dans un but pédagogique.

Une API de Test pour des tests maintenables :

- Les scénarios font quelques lignes
- Refactors sur l'API de test aussi



VI - La qualité : sur les tests aussi !

pourquoi la qualité des tests est-elle généralement **plus faible** que sur le code ?



Refactor empêché par les tests auto,
de plus mauvaise qualité que le code !

L'analyse Sonar ne passe pas sur les tests

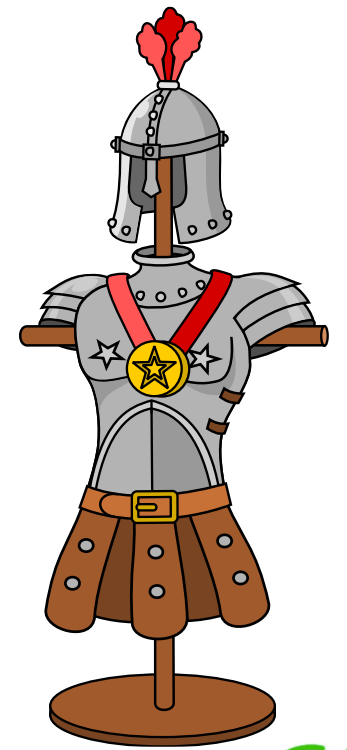
➤ Pourquoi ?



Mutation Testing pour vérifier la qualité de
nos tests dans un but pédagogique.

Une API de Test pour des tests maintenables :

- Les scénarios font quelques lignes
- Refactors sur l'API de test aussi



VI - La qualité : sur les tests aussi !

pourquoi la qualité des tests est-elle
généralement **plus faible** que sur le code ?



Refactor empêché par les tests auto,
de plus mauvaise qualité que le code

L'analyse Sonar ne passe pas sur les
➤ Pourquoi ?

Mutation Testing

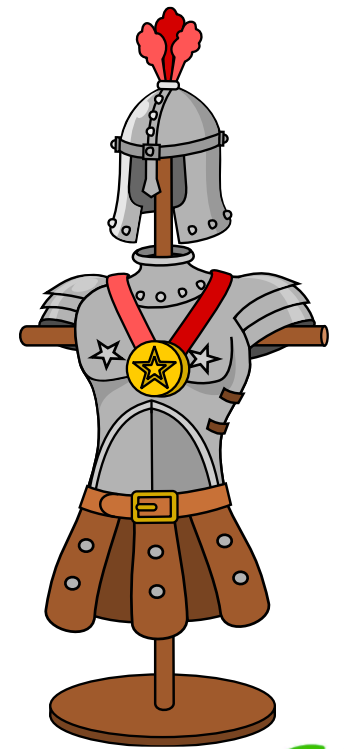


**Vérifier la qualité des tests
en cherchant des bugs connus**

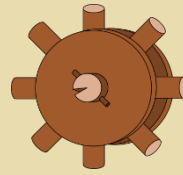


ing pour vérifier la qualité de
un but pédagogique.

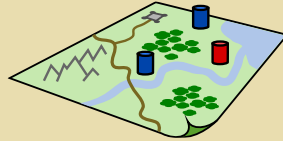
pour des tests maintenables :
s font quelques lignes
l'API de test aussi



I *Automatisation*



II *Stratégie de test*



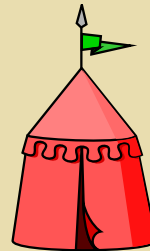
III *Patrimoine de tests*



IV *Rôle du testeur*



V *Campagnes de tests*

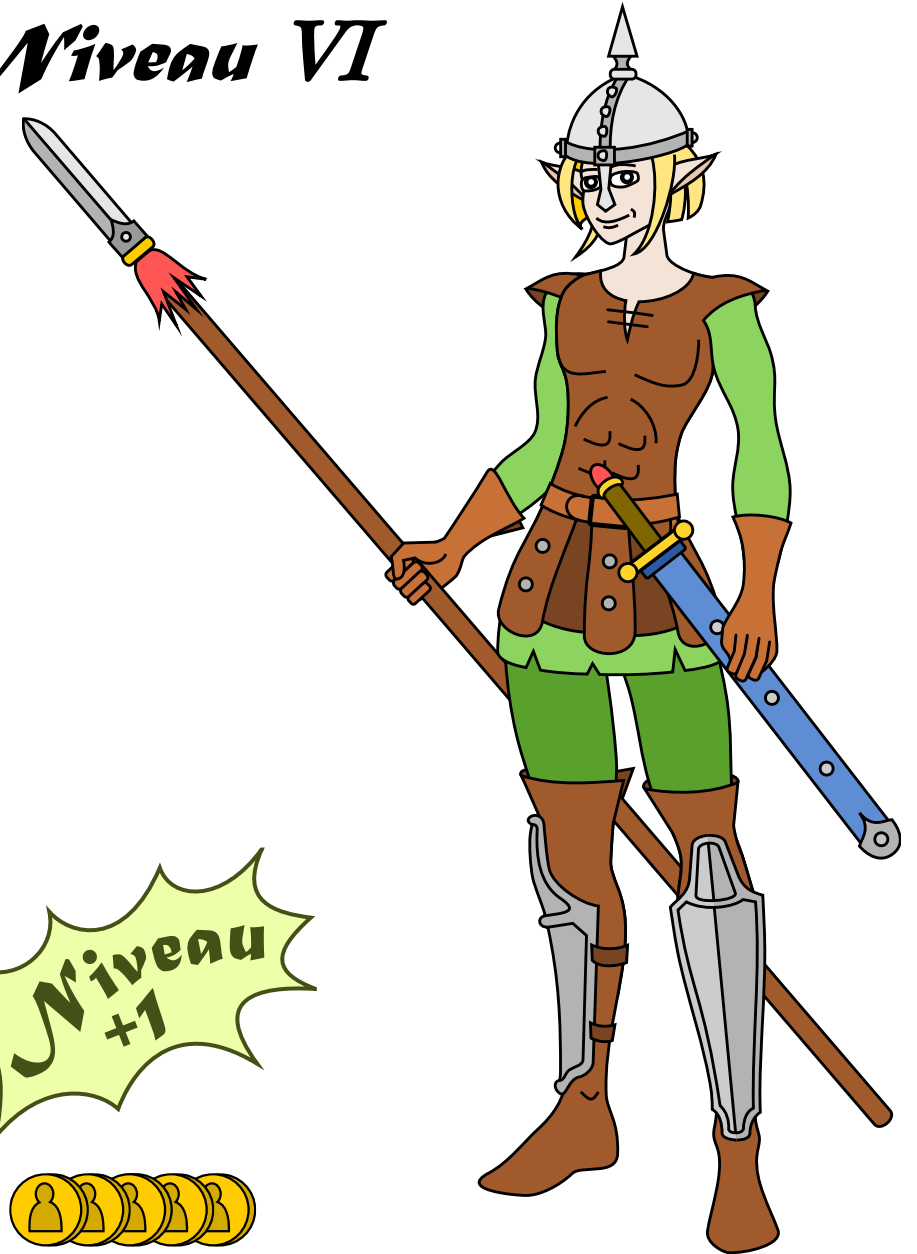


VI *Qualité des tests*



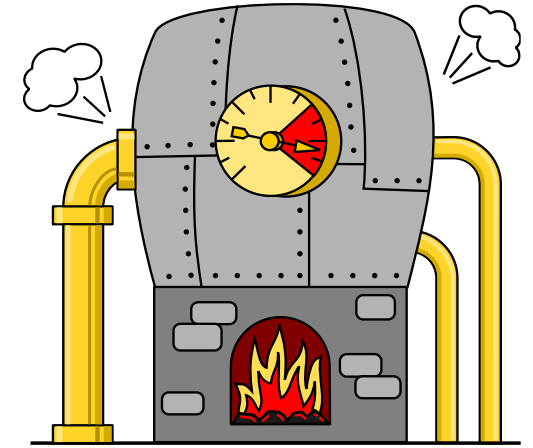
VII *Risque industriel*

Niveau VI



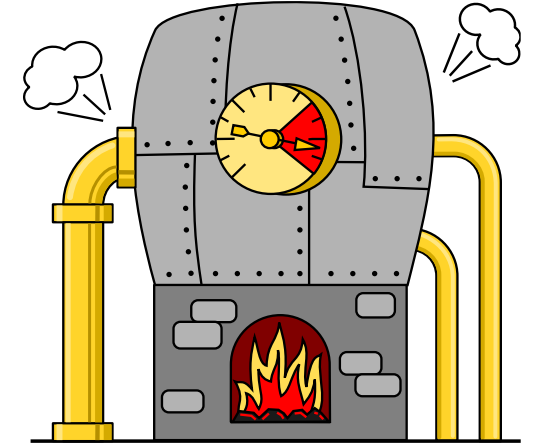
VII - Gérer le risque industriel

***A quel moment, un incident grave
a le plus de chance de se produire ?***



VII - Gérer le risque industriel

le risque industriel est maximal **après**
une longue **période de confiance**



On a rien prévu pour certains tests

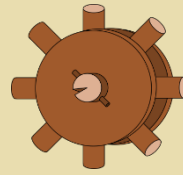
- on teste en prod (ex annuaire)
- Depuis quand n'avons nous pas testé la restauration de la sauvegarde ?
- Comment avez-vous testé les alarmes ?



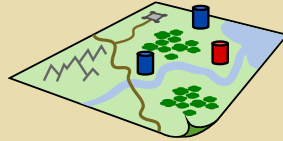
Concours « Chaos Monkeys » façon hackathon
(en hors prod) à intervalles réguliers :

- motivation / sensibilisation
- code++ poussé en production

I *Automatisation*



II *Stratégie de test*



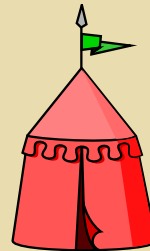
III *Patrimoine de tests*



IV *Rôle du testeur*



V *Campagnes de tests*



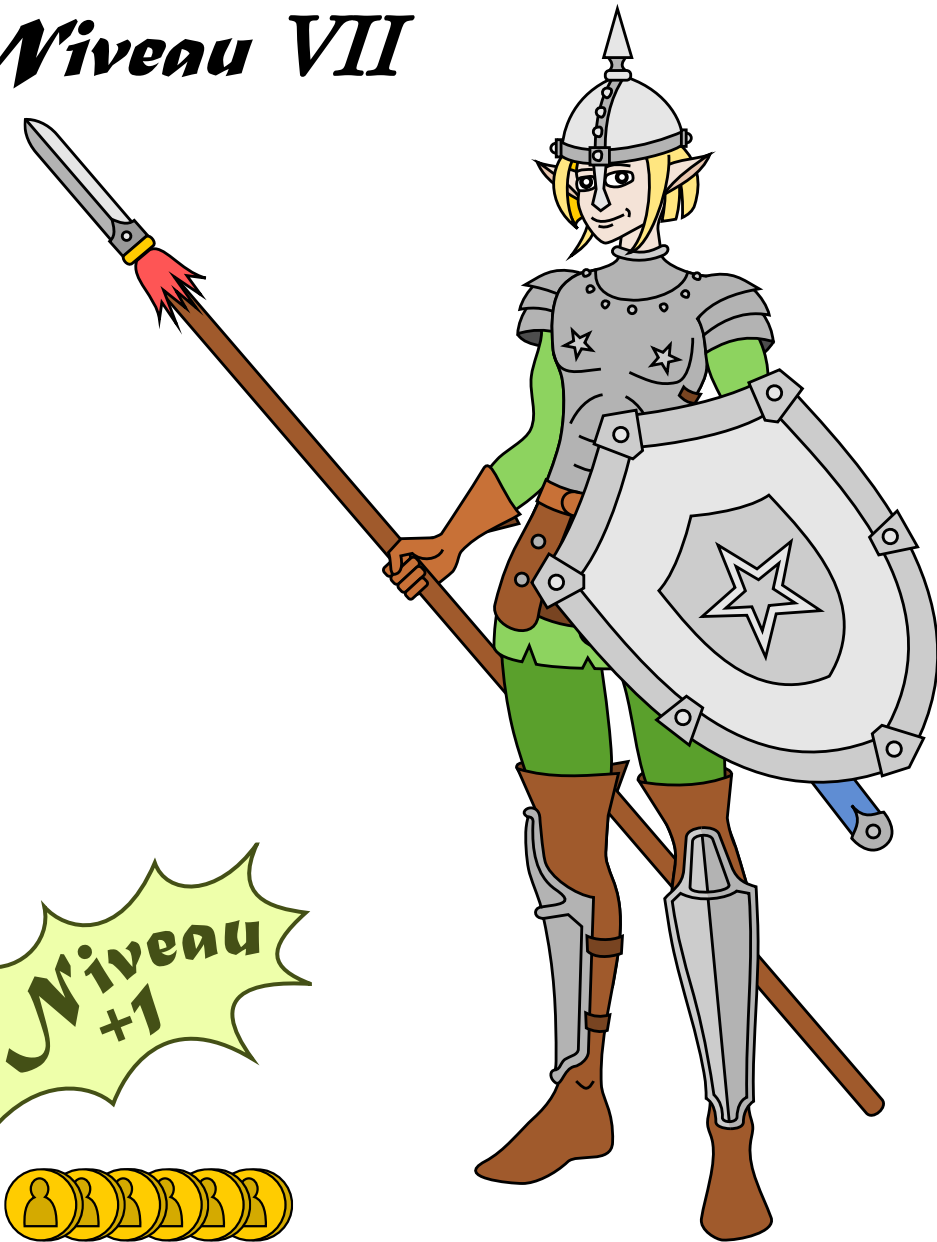
VI *Qualité des tests*



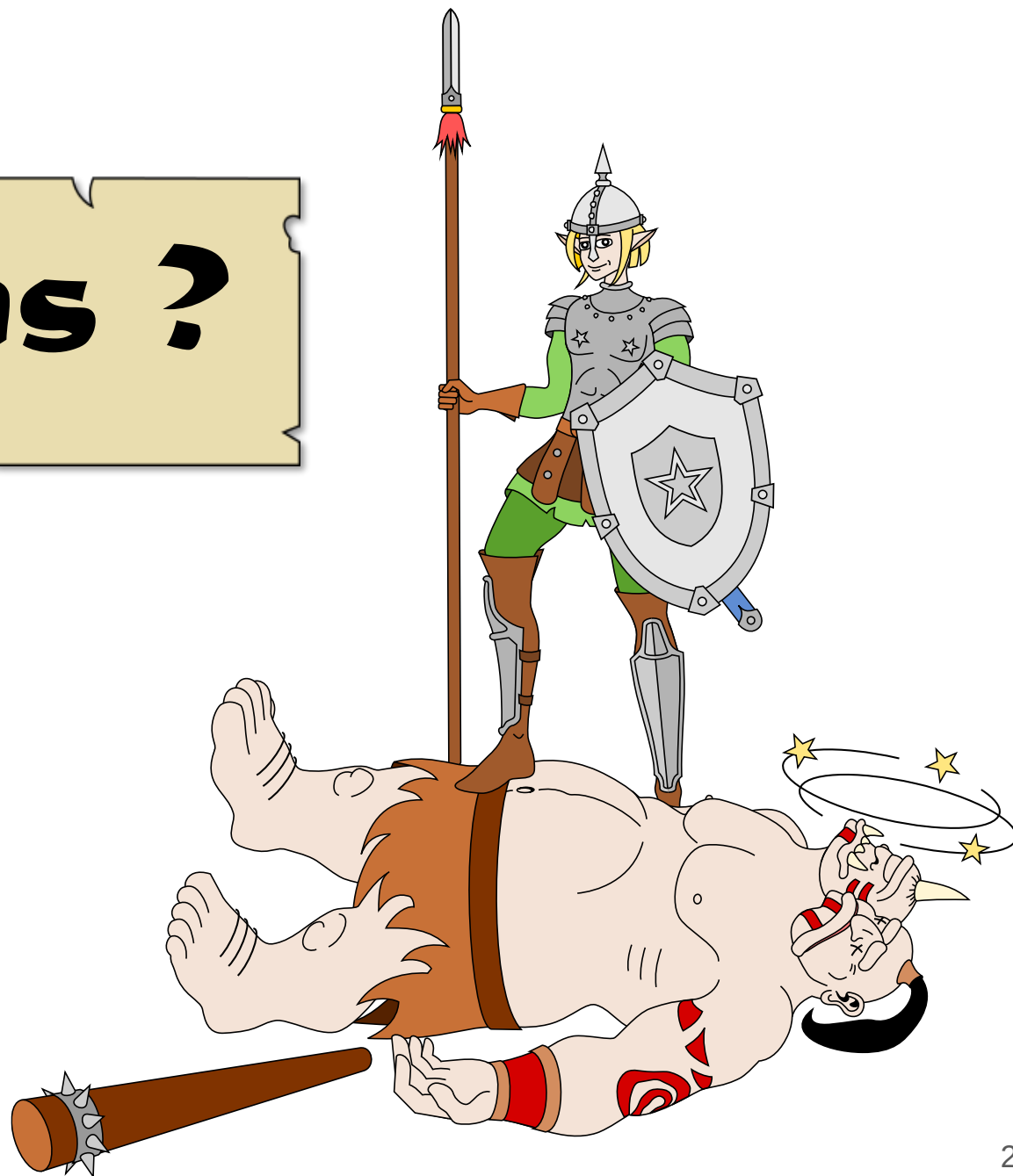
VII *Risque industriel*



Niveau VII



Des questions ?



7 choses à retenir :

- 1) Un prérequis : **automatiser** tout ce qui peut l'être !
- 2) Etablir **une stratégie de test** délibérée et partagée par tous
- 3) Le **patrimoine de test** doit être partagé (et lisible) par tous !
- 4) Le testeur agile comme **coach de l'équipe**
- 5) Gérer nos **campagnes de tests** comme des pros
- 6) **La qualité** concerne aussi les tests !
- 7) Gérer le risque industriel avec des pratiques type « **Chaos Monkeys** »

Take Away

Avec quoi je repars ?

- ***Qu'est-ce qui m'a marqué ?***
- ***Qu'est-ce qui va m'aider pour la suite ?***



En savoir plus

- Jeu caTestdrale : <http://catestdrale.github.io/>
- Gherkin : <https://cucumber.io/docs/gherkin/>
- Mutation Testing avec PIT : <https://pitest.org/>
- Chaos Monkey :
https://fr.wikipedia.org/wiki/Chaos_Monkey
- Article compagnon :
<https://conserto.pro/blog/les-tests-agiles/>

