

# Les bonnes pratiques Agile

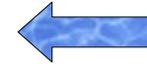


STS IRIS

Module 4.2 - « Gérer et organiser un projet informatique »

# Sommaire

Pair Programming  
TDD



# Introduction au Pair Programming (1)

- Imaginez ...
  - Deux personnes qui travaillent sur la même tâche.
  - Les deux ont le même objectif.
  - Les deux ont des expériences (expertises) différentes.
  - L'un exécute une tâche.
  - L'autre regarde, évalue la situation, le corrige et valide l'exécution correcte de la tâche.
  - Les deux personnes travaillent en équipe
  
- Vous avez des exemples en tête ?

## Introduction au Pair Programming (2)



# Introduction au Pair Programming (3)



## N'est ce pas une perte de temps ?

- Les deux développeurs font le travail d'un !
- Un débutant va **ralentir** le développeur confirmé !
- Moins de travail dans le même temps !
- Le **coût** est doublé !
- Pourquoi mettre deux personnes pour un job qu'une personne **seule** peut faire ?



Vous voulez d'autres exemples ?



...

## Les apports du pair programming

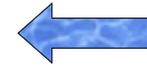
- Le code appartient à l'équipe
- Contrôle en continu du code
- Moins de défauts et détection précoce des défauts
- Aide à la résolution des problèmes
- Apprentissage et intégration des nouveaux développeurs
- Moins de distractions et meilleur productivité
- Satisfaction du travail en équipe
- Emulation positive
- Formation des équipes et communication
- Gestion des équipes

## En pratique

- On change les rôles régulièrement (~20 mn)
- Pas forcément en permanence mais régulièrement et progressivement (1h/jour, ½ journée par semaine, ...)

# Sommaire

Pair Programming  
TDD ←



# TDD : Introduction (1)

- TDD : Test-Driven Development (Développement dirigé par les tests)
- Pourquoi tester ?
  - Pour avoir un code :
    - Correct
    - Robuste
    - Complet
  - Trouver les bugs le plus tôt possible (€++)
  - Intégrer un processus de qualité
  - Tester un code aide à rester concentré sur le développement



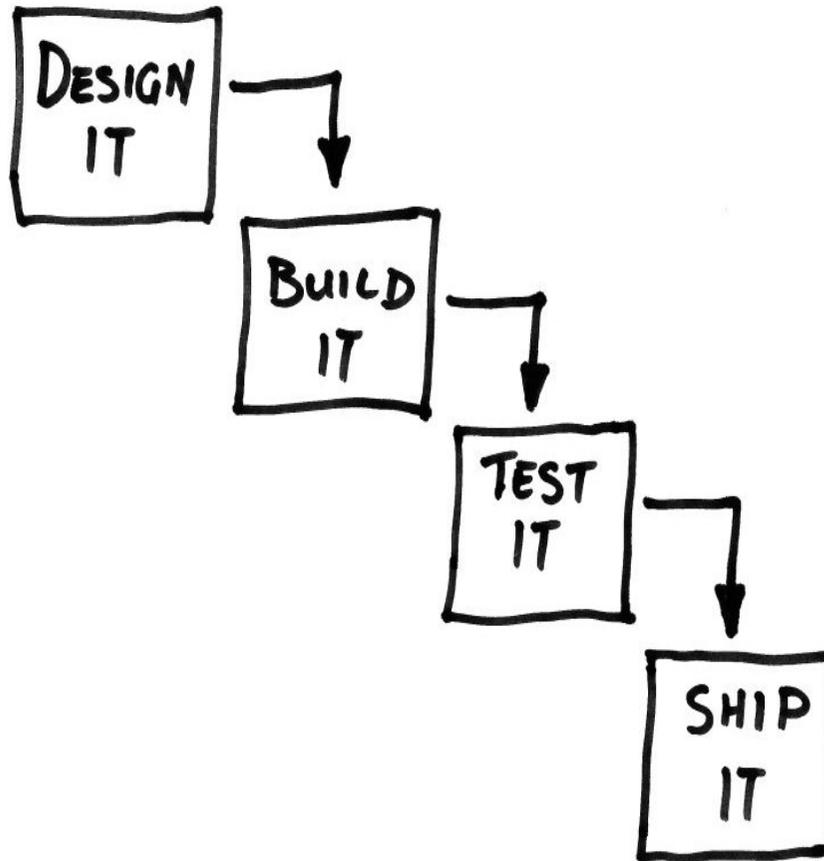
## TDD : Introduction (2)

- Pourquoi commencer par faire des tests ?
- Apprendre du passé : V88 ARIANE 501- 4 juin 1996



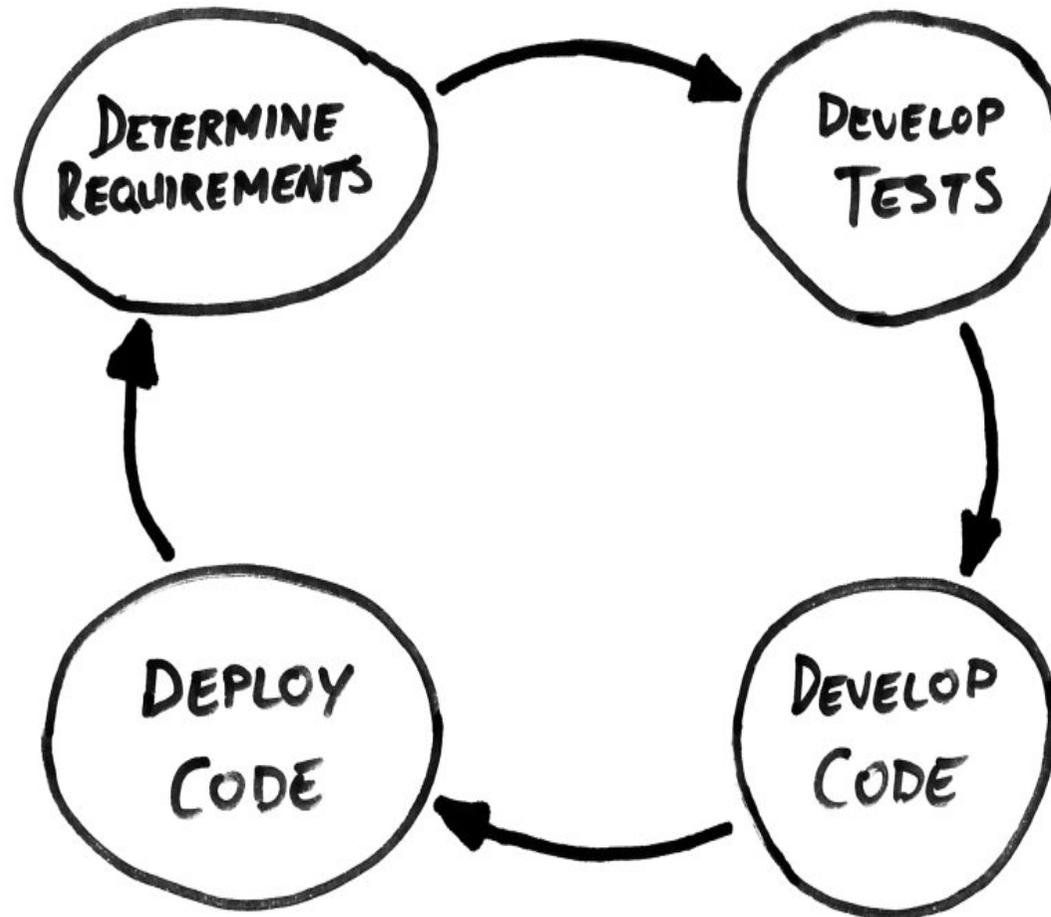
## TDD : Introduction (3)

- Développement et test : ancienne vision



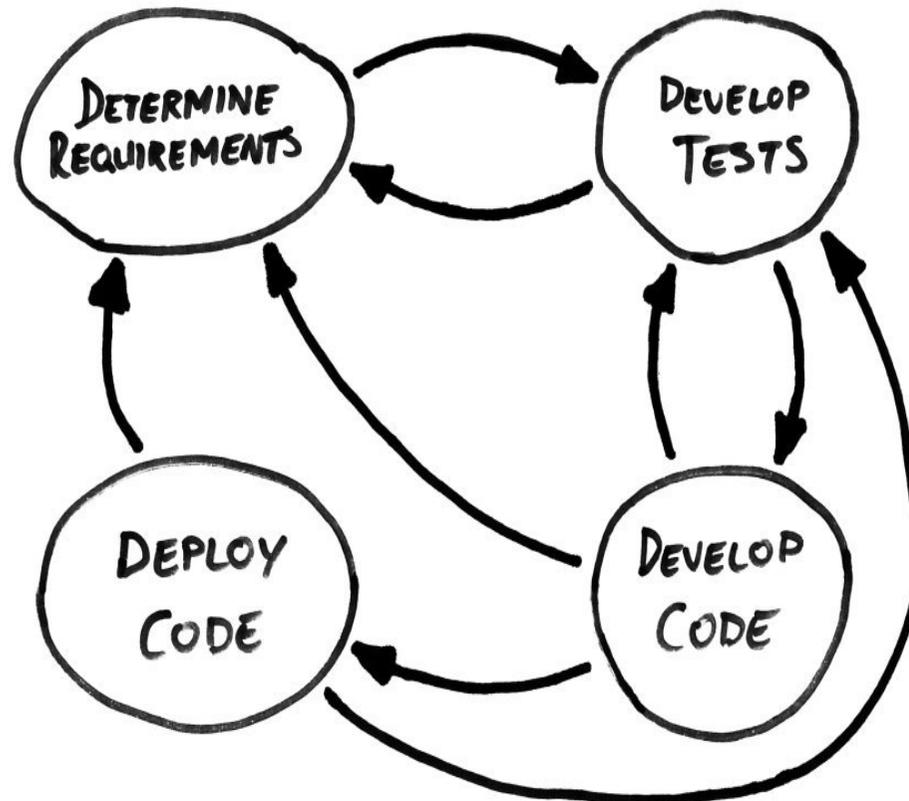
## TDD : Introduction (4)

- Une approche un peu plus Agile...



## TDD : Introduction (5)

- Une approche encore plus Agile...



# Le cycle TDD

