



HipTest

Comment échouer dans l'adoption de l'approche Behavior Driven Development ?



Présentation par



Laurent Py



Vincent Prêtre

Contexte

1. **Notre expérience**
2. **BDD Survey 2018**
3. **Discussion avec les utilisateurs**

Agenda

Introduction à BDD

Problèmes organisationnels

Problèmes pratiques : écriture et automatisation

Conclusion

Introduction à BDD

The bottom right corner of the slide features a decorative graphic consisting of several overlapping, semi-transparent geometric shapes. These shapes are primarily triangles and polygons in shades of light blue and teal, creating a modern, abstract design.

Objectifs de BDD

1. **Aligner** les membres d'un projet
2. Créer une terminologie métier **commune**
3. Définir le **périmètre** d'une fonctionnalité
4. Obtenir une **documentation vivante** du produit

Exemple de scénario avec Gherkin

Given I am a valid Hiptest user

When my account is deactivated

Then I should be logged-out of the application

BDD survey 2018

The number of teams already using Behavior Driven Development has increased from 38%(2017) to 44%(2018).



Within the next 3-5 years, more than 50% of teams are expecting to use BDD.

BDD survey 2018



BDD survey 2018

Key Takeaways

1

By 2023, 50% of teams are expecting to use BDD.

50%

2

The two main benefits of BDD are collaboration and improving quality.



3

BDD isn't just about testers and developers, it's about key stakeholders working together.



Problèmes organisationnels



Passer à BDD ? Bien sûr mais ...



... ça doit être rapide et peu couteux !

Manque de support

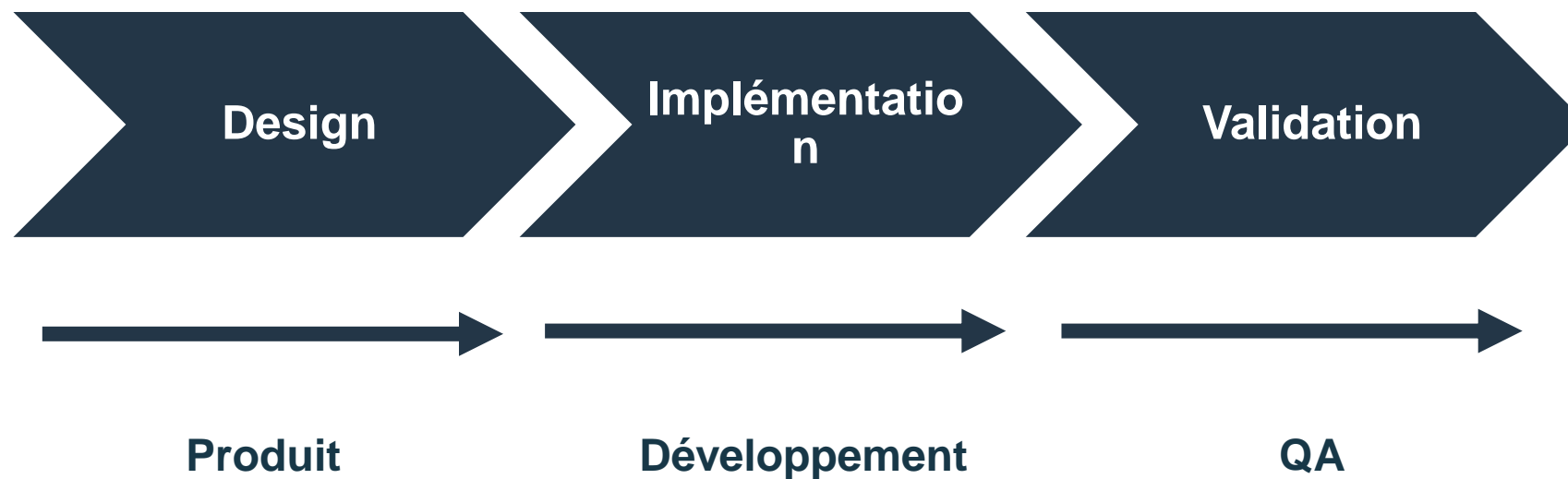
Estimer le temps

- Pris sur les équipes
- Pris pour être efficace avec BDD

Estimer les ressources nécessaires

- Accompagnement
- Outillage

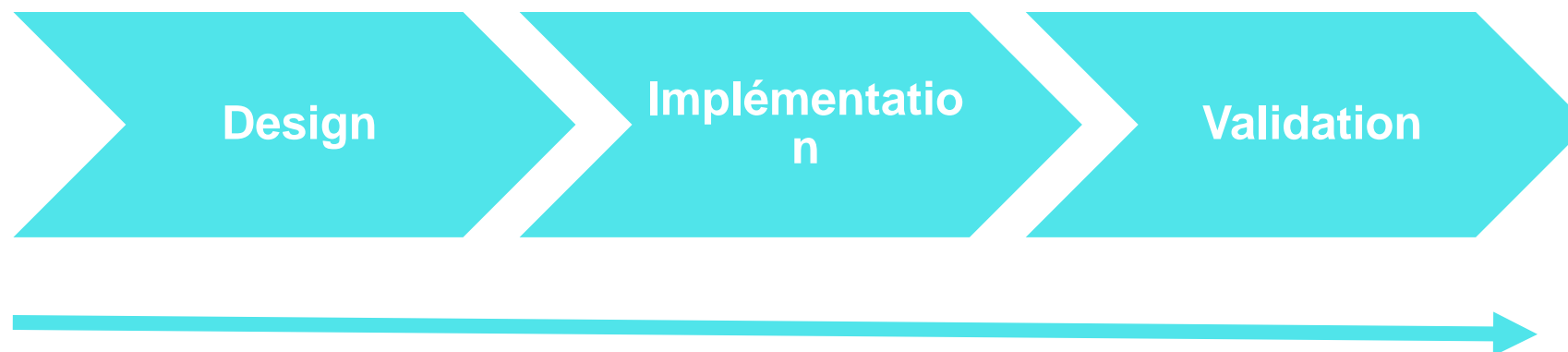
Temps investi : Cycle en V



Temps investi : méthode Agile



Temps investi : process BDD



Produit + Développement + QA

Temps investi : bénéfiques



Le temps investi au départ n'est pas perdu à développer une fonctionnalité qui ne correspond pas aux besoins initiaux.



Temps nécessaire à la transition

- La transition à BDD implique un **changement de paradigme**
- Corrolaire → "forcer" l'utilisation de BDD au sein de l'équipe

Temps nécessaire à la transition : solutions

- Faire un pilote de BDD par équipe
- **Laisser le temps** aux équipes pour prendre en main BDD → trouver le **bon équilibre**

Manque d'accompagnement : solutions

- **Former** les acteurs: développeurs, QA, business...
- **Se faire accompagner** par des experts

Outillage inadapté : un schisme entre les équipes



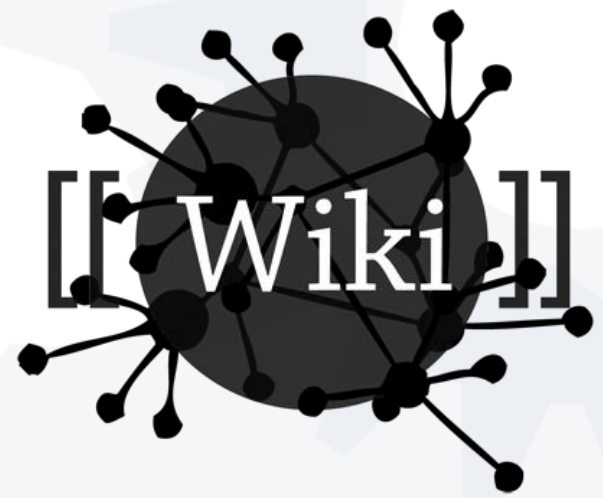
Equipes techniques



git



Métier



Outillage inadapté : un schisme entre les équipes

- **Cloisonnement** des définitions
- Problèmes de **synchronisation** entre les équipes

→ Normalement, les scénarios implémentés et automatisés doivent jouer le rôle de **documentation vivante**

Problèmes pratiques : écriture & automatisation



Des scénarios trop techniques

- Scénarios écrits par les développeurs
- Très faciles à automatiser
- Décrit une utilisation/cas aux limites mais pas de vision globale de la fonctionnalité
- Tous les tests doivent être écrits avec la syntaxe Gherkin



Scénario trop technique

Given users Harry and Hermione ↗

and project 'Coffee machine'

and scenario Simple use is selected

and JIRA issue panel is opened for ISSUE_ID from = type add-on

When current element is linked to current issue

Then JIRA issue panel contains Simple use

and tag JIRA: ISSUE_ID is displayed

When element Simple use is unlinked from current issue

Then no tag is displayed

When folder Coffee machine - Hiptest publisher sample is selected

and current element is linked to current issue

Then JIRA issue panel contains Coffee machine - Hiptest publisher sample

and tag JIRA: ISSUE_ID is displayed

Scénario vraiment trop technique

Given user Harry

and project F1/Sc1 and F2/Sc2

and folder My project is selected

and subfolders F1, F2 are selected

When folders manage tags modal is opened

and tag my_new:tag is added to the selection

Then tag my_new: tag is available in manage tags modal

and folder F1 has tag my_new: tag

and folder F2 has tag my_new: tag

When folder My project is selected

and subfolders F1, F2 are selected

and folders manage tags modal is opened

and tag my_new: tag is removed from the selection

Then no tag is available in manage tag modal

and folder F1 has no tag

and folder F2 has no tag

Des scénarios trop abstraits

- Scénarios écrits par l'équipe produit/métier
- Très haut niveau
- Difficiles à automatiser
- Manque de données utiles pour les développeurs (flux d'utilisation, données trop vagues etc ...)

Scénarios/définitions trop abstraits

Generate metrics by tags

[Export](#)

In Order to have an overview of my project

As a team member

I want to generate charts based on tag usage

[Add new tag](#)[Add attachment](#)

2 scenarios

[Create](#)

All / None Show tags

A chart can be generated based on a tag key

A chart can be generated to compare multiple tags

Bonnes pratiques BDD

- Collaborer
- Réutiliser la terminologie métier créée
- Utiliser la syntaxe déclarative
- Mettre en place des règles communes



Scénario compréhensible par tous

Given all test results are passed

and no execution cycle is started

When I start a new execution cycle

Then the test run Sprint 1 should contain a started execution cycle

and all test results should be not run

Durée d'exécution des tests

BDD n'est pas uniquement:



Tests Selenium/Appium: longs et fragiles

Durée d'exécution des tests

API +
Back



API +



Given all test results are passed

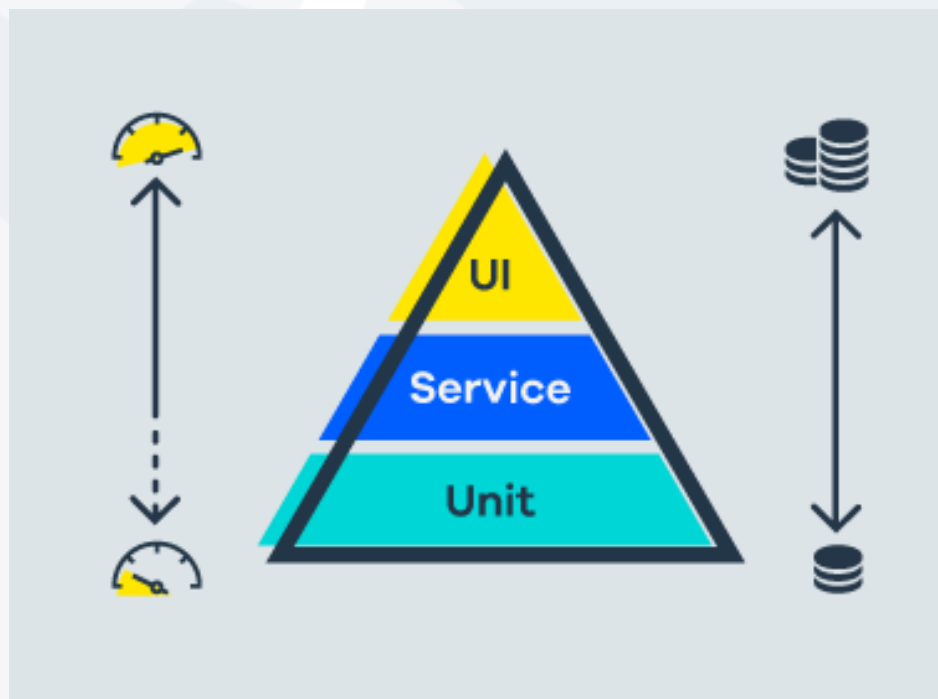
and no execution cycle is started

When I start a new execution cycle

Then the test run Sprint 1 should contain a started execution cycle

and all test results should be not run

Notre pyramide de tests



- 500 tests BDD
- 4300 tests Ember
- 7000 tests Rspec

Total tests: environ 12000

→ 32 minutes d'exécution

Conclusion



Conclusion



- 1** **Changement de culture**
- 2** **Collaboration**
- 3** **Trouver le bon équilibre d'écriture des scénarios**



Thank you!
