

Le BDD: Une conversation pour découvrir et spécifier les besoins d'affaires



FÉLIX-ANTOINE
BOURBONNAIS

B.ING., M.SC, PSM

&

PASCAL ROY

ING., CSM, PSM, PMP





Diapositives et références

<http://conferences.elapsetech.com/bdd-spec-affaires/>



Cette présentation s'adresse à un public **débutant** désirant **s'initier** au BDD.

Le **BDD** n'est pas une technique de test

Le BDD peut être fait **sans outils**
comme Cucumber, SpecFlow, etc.

Le **BDD** est un processus pour
découvrir, spécifier et documenter les
requis d'affaires

bienvenue 

Cadre de la conférence

Public cible

Cette présentation s'adresse à un public **débutant** désirant **s'initier** au BDD.

Nos objectifs

- Comprendre (survol) ce qu'est le BDD et pouvoir continuer à explorer le sujet
- Le BDD n'est pas une technique de test
- Le BDD est une approche pour la découverte et la spécification

Félix-Antoine Bourbonnais

B.ing., PSM, M.Sc.



Pascal Roy

Ing., PSM, CSM, PMP





Formations



Mentorat



Diagnostics



Conférences

Conférenciers

Formateurs

Mentors & coaches

Scrum

QA Agile

BDD

Essais automatisés

TDD

Architecture évolutive

DDD

...

Équipe & Affaires

Tech.

Gestion

Conseils stratégiques

Gestion de projets

Agilité

Quels sont les défis typiques
concernant la documentation ?

Problèmes concernant la documentation

> Transmission et distorsion de l'information

Passage de documents souvent sans explications et contexte

> Interprétations différentes

Impossibilité de transmettre directement un modèle mental

> Documentation pas à jour

Quelle est la quantité réaliste que l'on pourra garder à jour?

> La qualité \neq la quantité

Plus de texte \neq plus clair

Le BDD

Est-ce que je produis la bonne chose?

(« Building the right thing »)

BDD ?!?

15

BDD :
« Behavior-Driven Development »

Un processus

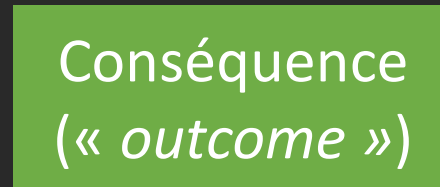
pour

Explorer
Découvrir
Spécifier
Piloter

le

comportement
souhaité d'un
système

Comportement ?!?



Comment doit se
comporter le système*.

* Système au sens large et peut englober plusieurs systèmes « physiques » ou applications dans certains cas.

Explorer
Découvrir
Spécifier
Piloter

utilisant
→

la conversation et
des exemples concrets

pour forger
→

une compréhension
commune du problème

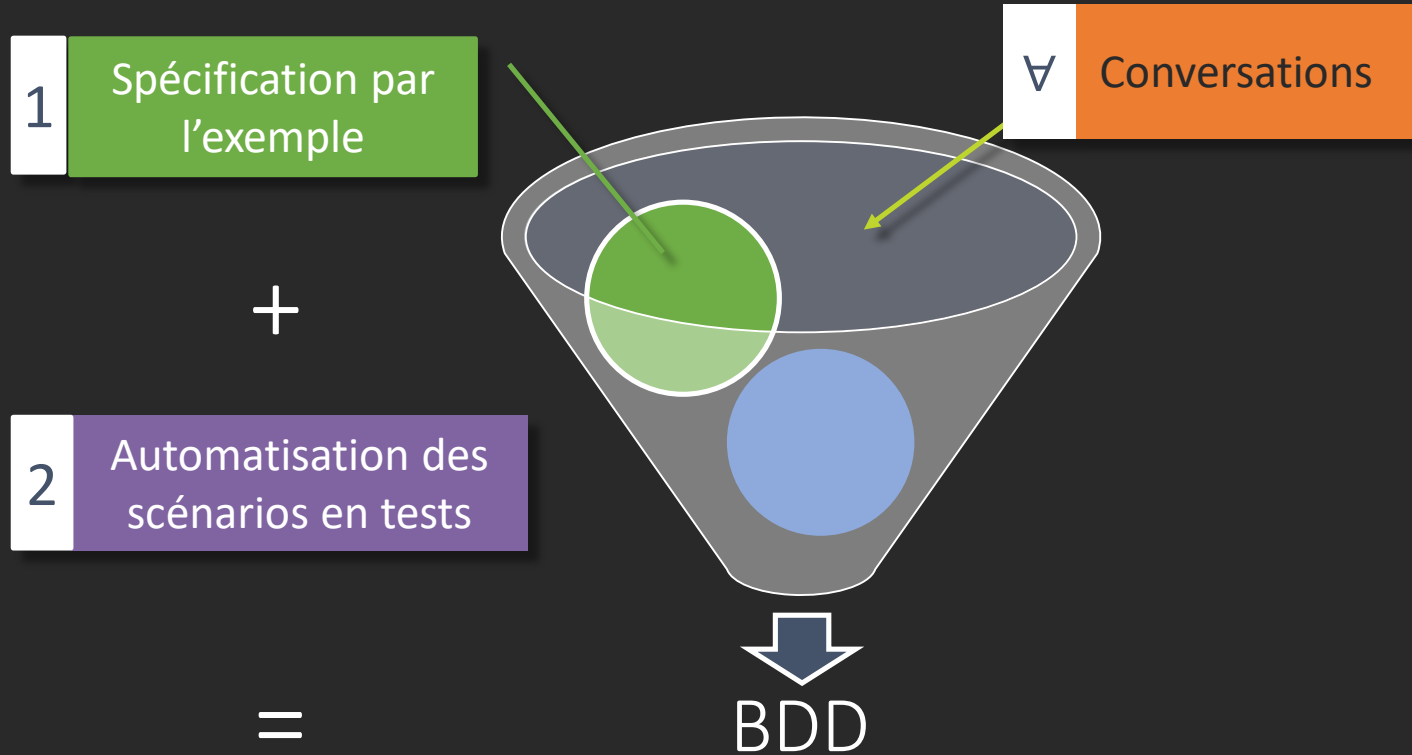
et piloter
→

le développement avec
des tests automatisés

=
→

une spécification
exécutable

Les ingrédients



BDD versus Tests

Tests

Fonctionne comme prévu? + vérifie et valide le développement

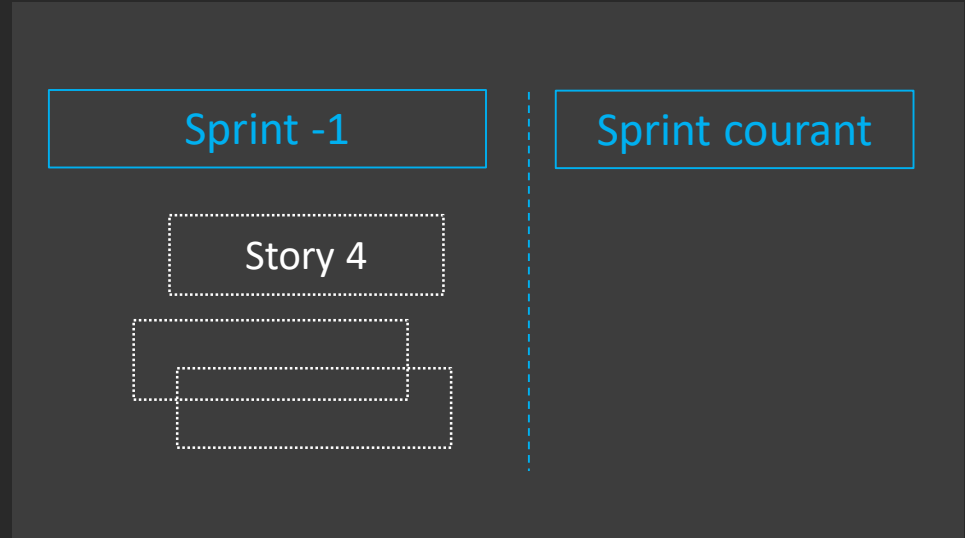
BDD

Bon comportement/produit? + guide le développement

Il était une fois...

- Alice --> Propriétaire du produit
- Jean --> Assurance qualité
- Bob --> Développeur

- La Story 4 devrait être réalisée dans le prochain sprint



Explorons la Story 4...

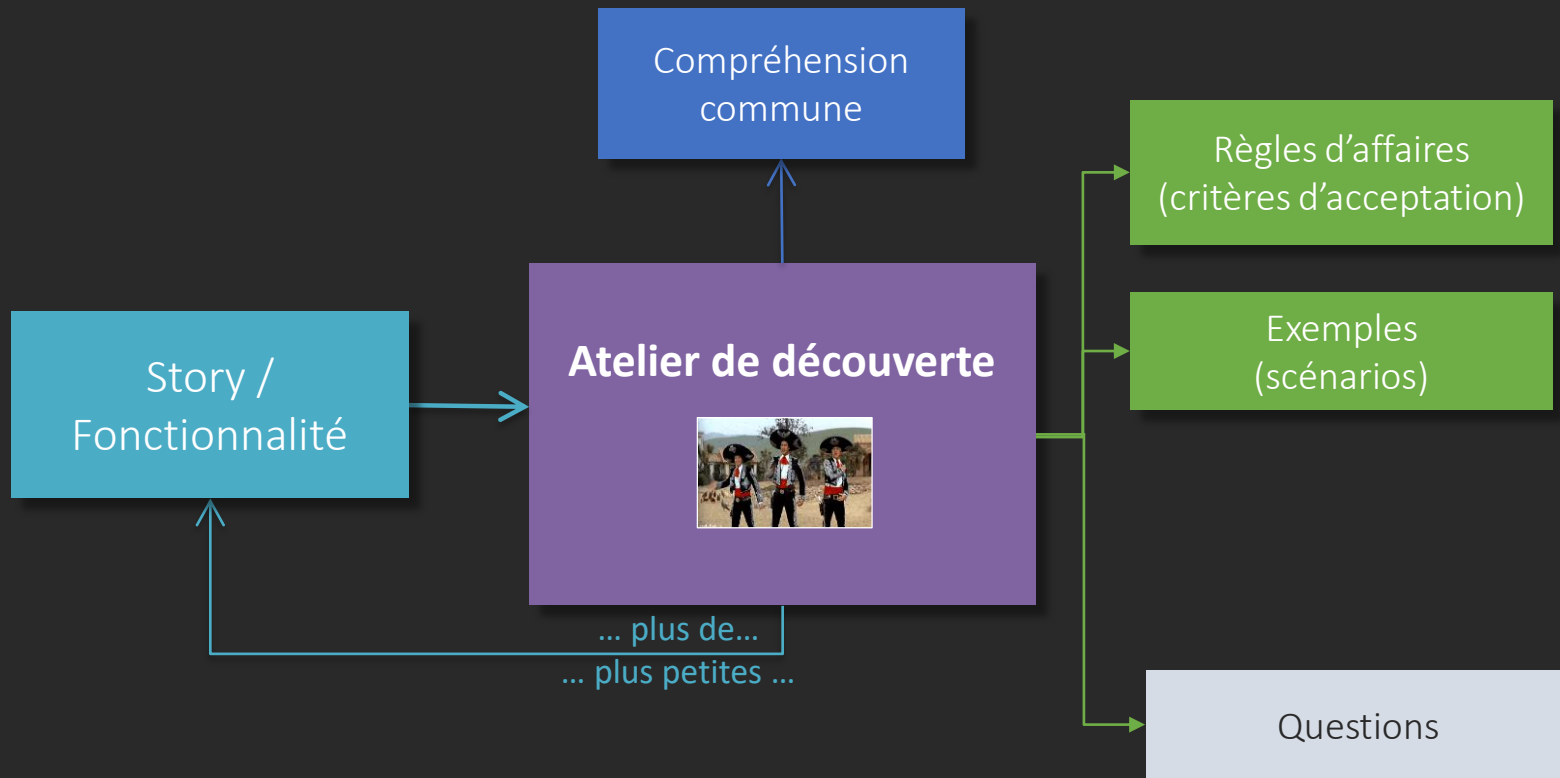
- Sa portée
- Ses critères d'acceptation et règles d'affaires
- Etc.

Découvrir et comprendre

Découverte

Où ?

Atelier de découverte



Découverte

Qui ?



Affaires / Produit
- Valeur -



QA / UX
- Utilisabilité & détails -



Dev / Ops / ...
- Faisabilité -

3 Amigos

3 Amigos \neq 3 personnes



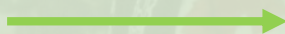
Affaires



A le **contexte**, le *pourquoi* et connaît la **valeur**.



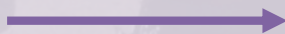
QA / UX



Pense naturellement aux **détails** et à l'**utilisabilité**



Dev & Ops



Pense à la **faisabilité** et aux **aspects techniques** et à la **complexité**

Découverte

Comment ?

User Story / Fonctionnalité

décrite par



- Règles d'affaires; *et/ou*
- Critères d'acceptation;
- Conditions de succès

Mais avons-nous compris la
même chose ?

Est-ce que ça veut dire X
ou Y ?

BDD

centré autour

Des
conversations



~~Voici les scénarios~~



~~J'ai écrit mes « BDDs »~~



Le BDD permet à l'équipe de
découvrir et **s'approprier** la
spécification

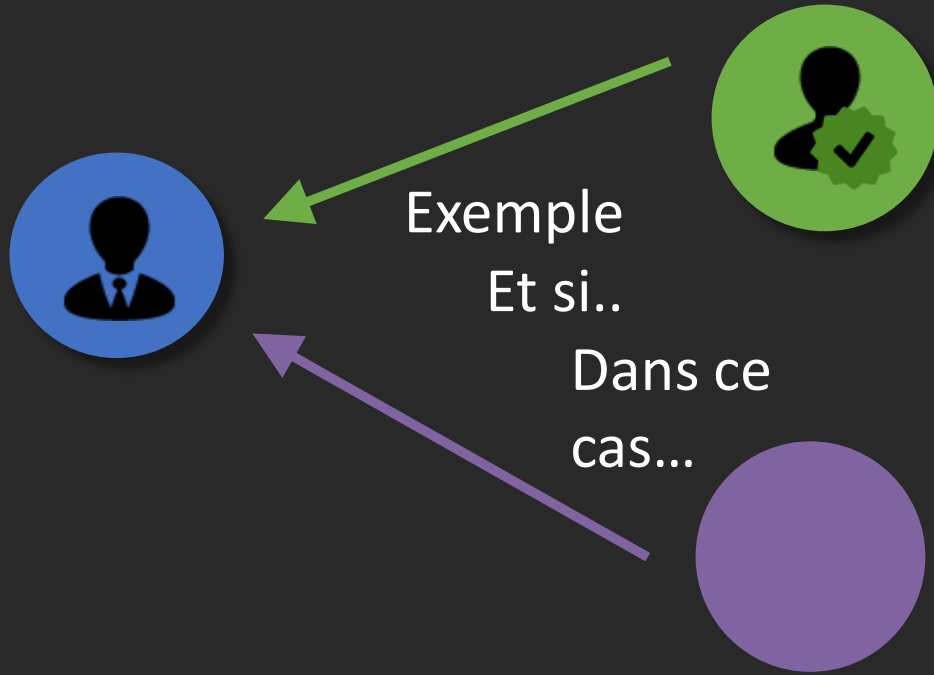
Voici
les scénarios



Donne-moi un **exemple!**


On utilise des **exemples** pour
communiquer et **découvrir** les règles.

Déroulement



Questions
sous forme
d'exemples

Ainsi, on valide que tous ont la **même compréhension** et qu'il ne manque pas de règles importantes.



Pour une activité ludique sur l'interprétation des règles à faire avec votre équipe:

<https://www.slideshare.net/Reloadk/bdd-how-to-solve-communication-problems>

Le principe



Découverte



Concrètement ?

Story: Retirer au guichet

Limite quotidienne de retrait

Si limite=1000
Et déjà retrait =500 le 7 Nov.
Montant=500 le 7 Nov.
->OK

Si limite=1000
Et déjà retrait =500 le 7 Nov.
Montant=501 le 7 Nov.
->Refusé

Si limite=1000
Et déjà retrait =500 le 7 Nov. 23h59
Montant=501 le 8 Nov. 00h01
->?????

Story: Retirer au guichet



Limite quotidienne de retrait

Si limite=1000
Et déjà retrait =500 le 7 Nov. 23h59
Montant=501 le 8 Nov. 00h01
->Refusé

Si limite=1000
Et déjà retrait =500 le 7 Nov. 23h59
Montant=501 le 9 Nov. 00h00
->OK

Si limite=1000
Et déjà retrait =500 le 7 Nov. 13h00
Montant=500 le 7 Nov. 14h00
->OK

Si limite=1000
Et déjà retrait =500 le 7 Nov. 13h00
Montant=501 le 7 Nov. 14h00
->Refusé

Autre règle

Exemple 1

...

Il existe des activités pour mener les ateliers

(ex.: Example Mapping, Feature Mapping, ...)

Activité: « Example Mapping »

Nouvelle Story

Fonctionnalité /
Story

Règle

Exemple

Exemple

Exemple

Règle

Exemple

Exemple

Règle

Exemple

Questions à
valider

Example Mapping

Story:
Retirer au guichet

Par tranche de 20\$

20\$ -> OK

21\$ -> Err

40\$ -> OK

Limite **quotidienne**

1000\$ @ 13h
1\$ @ 14h → Non

1000\$ @ 13h le 7
1\$ @ 13h le 8 -> OK

1000\$ @ 23h59
1\$ @ 00h01 -> NON

...

Doit avoir assez
de fonds

...

Si carte 123 est bloquée
Et j'essaie de retirer
????

Autre Story ->
Fraude

Limites (scope):

Distribution \$\$
exclue

Déjà authentifié

(Q) Si le client a une
marge ??



EXAMPLE MAPPING

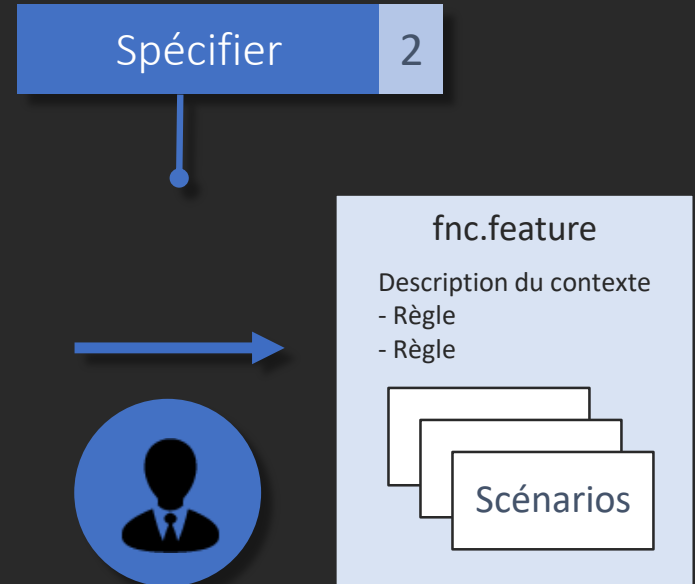
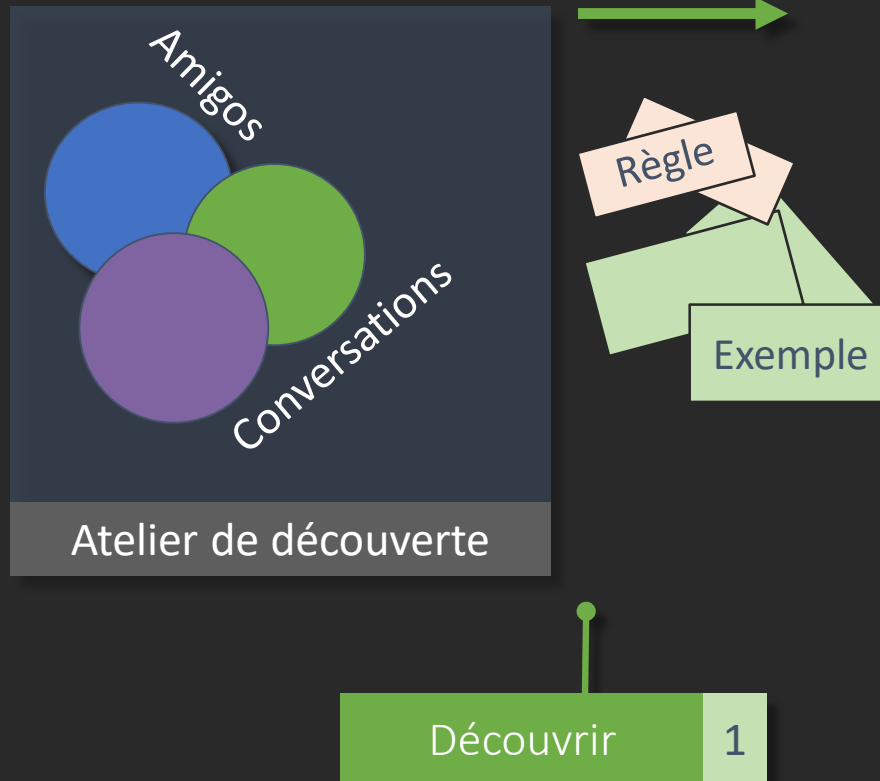
Pour en savoir plus sur *l'Example Mapping*
<https://cucumber.io/blog/2015/12/08/example-mapping-introduction>

Spécifier

Spécification

Quand ?
Qui ?

Déroulement



Écrire la spécification

1. Prendre les règles et les exemples trouvés lors de la découverte
2. Mettre au propre (pas en groupe!) --> **Spécification**
3. L'équipe s'entend sur le comportement

Spécification

Sous quel format?

Des exemples sous différentes formes!

SCÉNARIO: RETIRER AVEC UN SOLDE SUFFISANT
ÉTANT DONNÉ UN COMPTE AVEC 600\$
QUAND JE FAIS UN RETRAIT DE 400\$
ALORS LE RETRAIT EST AUTORISÉ
ET L'ARGENT EST RETIRÉ DU COMPTE

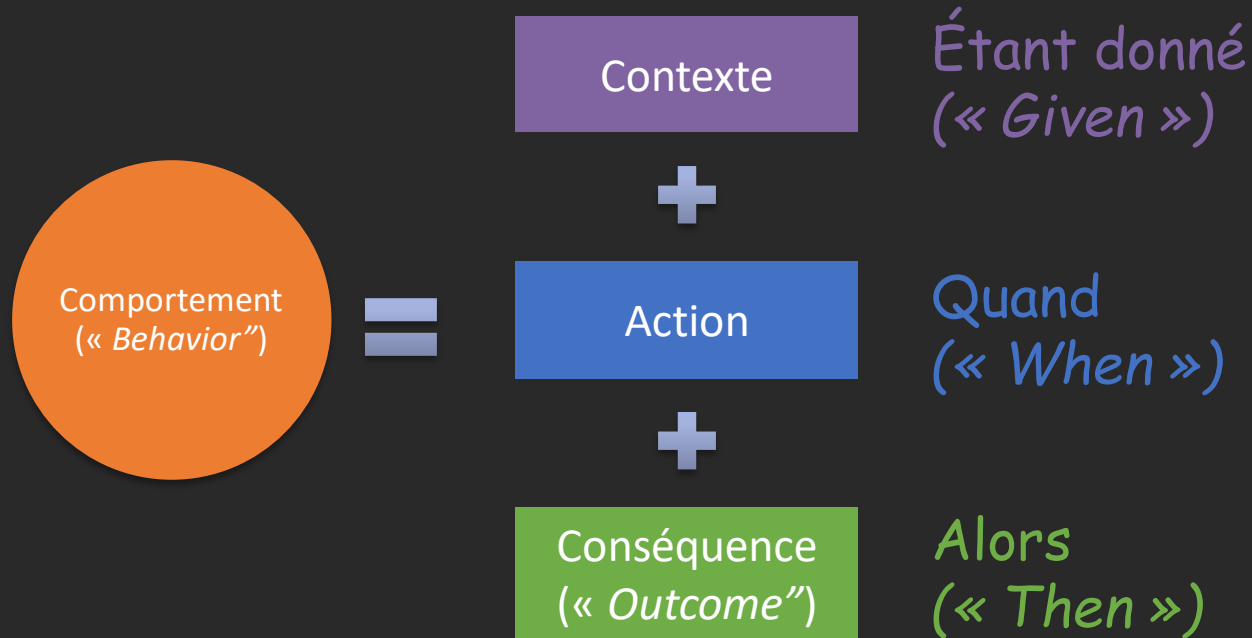
eg.Division

numerator	denominator	quotient?
10	2	5.0
12.6	3	4.2
22	7	~=3.14
9	3	<5
11	2	4<_<6
100	4	33



On utilise généralement le **Gherkin** : un langage **simple** et **textuel**

Le Gherkin



Des exemples concrets

Gherkin



Scénario: Retirer avec un solde suffisant

Étant donné un compte avec 600\$

Quand je fais un retrait de 400\$

Alors le retrait est autorisé

Et le solde du compte est de 200\$

Scénario Gherkin versus tests

Test: Retirer avec un solde suffisant

1. Se connecter avec compte 'C37362'
2. Aller dans les opérations > Retrait
3. Faire un retrait de 400\$
4. => un message montre que c'est accepté
5. Aller dans le sommaire
6. => Le solde doit être de 200\$

Impératif

Étapes --> Actions

Déclaratif

Étapes --> Comportements

Scénario: Retirer avec un solde suffisant

Étant donné un compte avec 600\$

Quand je fais un retrait de 400\$

Alors le retrait est autorisé

Et le solde du compte est de 200\$


```
remboursement.feature x
1 # language: en
2 Feature: Remboursement d'une facture
3
4 // Contexte: processus de remboursement
5
6 Pour éviter les erreurs et les clients mécontents, le remboursement se déroule en étapes:
7 1. Une proposition de remboursement est proposée au client
8 2. Le caissier rembourse le client (manuel pour l'instant)
9 2. Le remboursement est confirmé
10
11 [Proposition] --> [Remboursement] --> [Confirmation]
12
13
14 // Règles d'affaires
15
16 - R1: Le montant total des articles avec taxes est remboursé. On ne rembourse pas les frais.
17 - R2: Uniquement les factures payées sont remboursables
18 - R3: Quand le remboursement est confirmé comme effectué
19     --> La facture est marquée comme remboursée
20
21 // Exemples
22
23 @scope=api @focus
24 Scenario: Celui où la facture n'a pas de frais
25   Given la facture 'F1111' avec les articles:
26     | description | prix |
27     | Lait        | 3.00 |
28     | Oeufs       | 4.00 |
29   And des taxes de 3.50$ facturées pour la facture 'F1111'
30   And aucun frais pour la facture 'F1111'
31   When le client demande un remboursement pour la facture 'F1111'
32   Then la proposition de remboursement est de 10.50$
33
34 Scenario: Celui où la facture a des frais de transport
35   Given la facture 'F2222' avec un article de 5.00$
36   And des taxes de 2.00$ facturées pour la facture 'F2222'
37   And les frais de transport de 10.00$ pour la facture 'F2222'
38   When le client demande un remboursement pour la facture 'F2222'
39   Then la proposition de remboursement est de 7.00$
```

« Feature File »

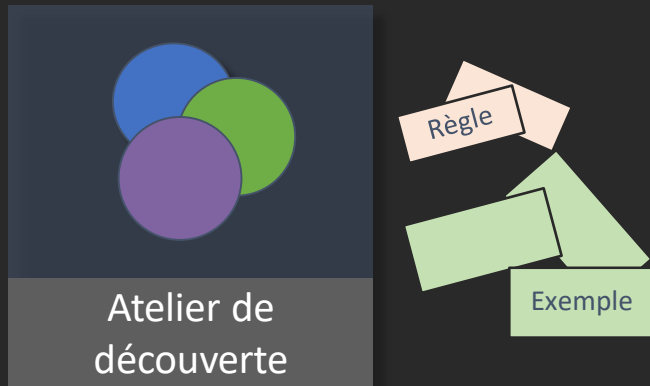
48 # ...

COMMAND MODE, Line 13, Column 1

Nos fonctionnalités composées de règles et de scénarios constituent maintenant une documentation

Piloter le développement

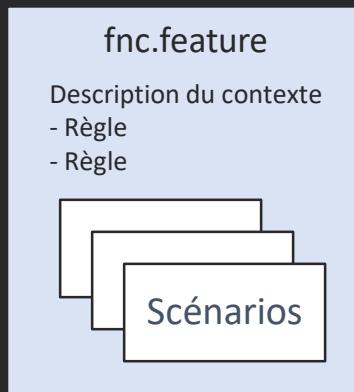
Déroulement



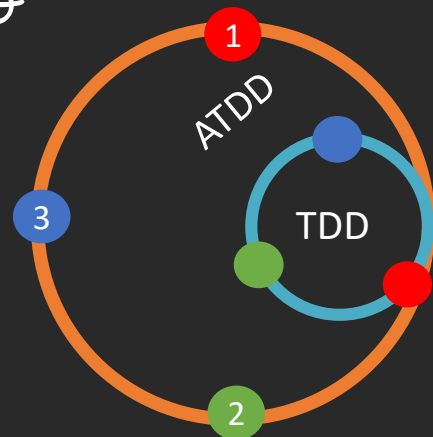
Découvrir 1



Spécifier 2

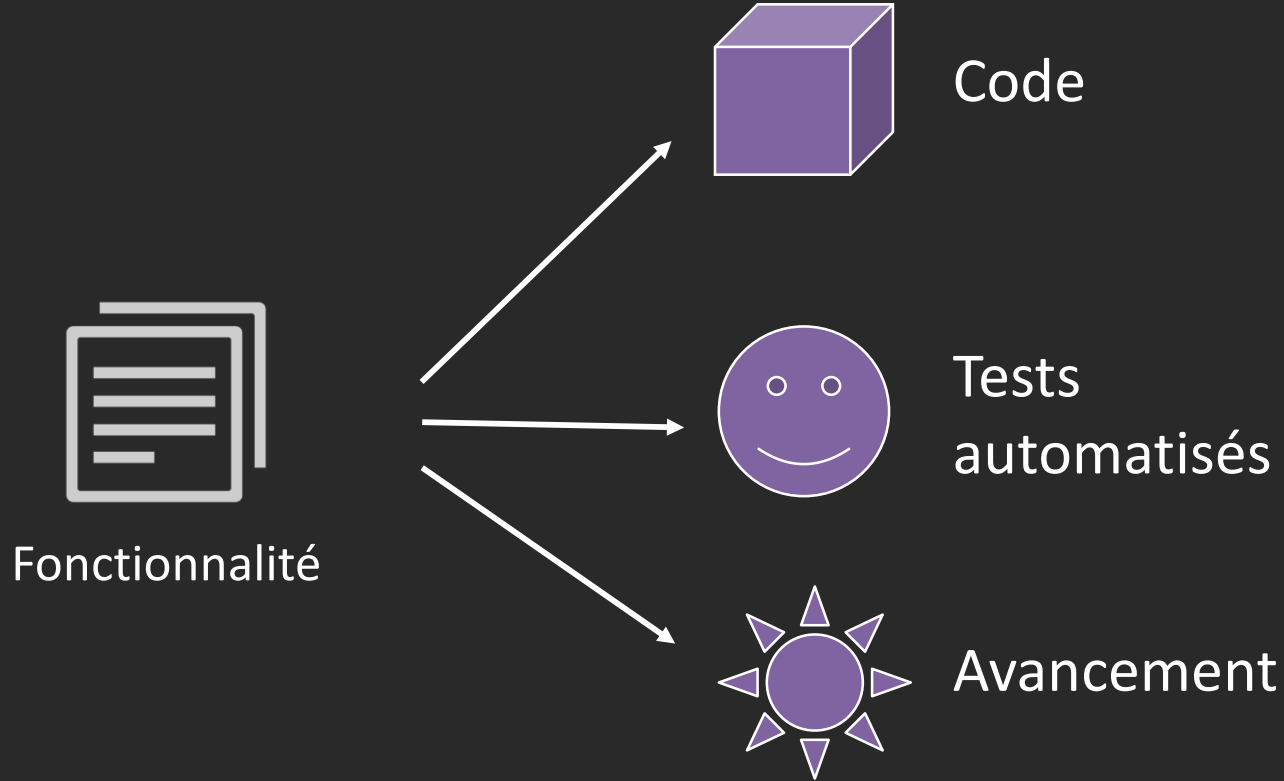


Développement & tests



Piloter 3

Piloter



Automatisation des tests

Feature:

Scenario: ...

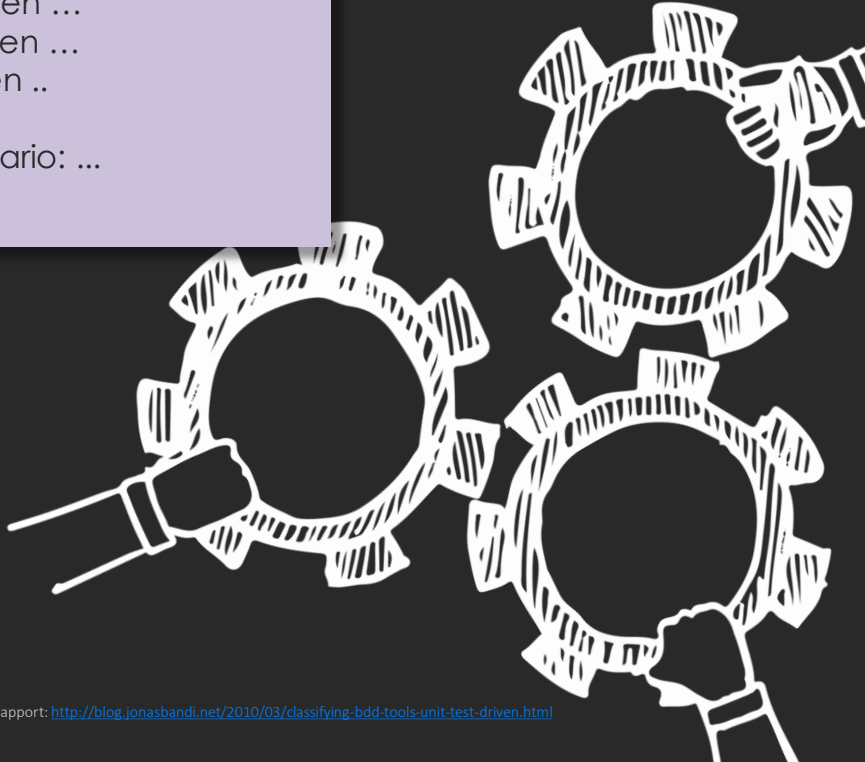
Given ...

When ...

Then ..

Scenario: ...

Automatisation



Cucumber Features

9 scenarios (1 failed, 8 passed)
31 steps (1 failed, 30 passed)
Finished in 0m36.456s seconds

Feature: Book search

In order to find books I might buy
As a potential customer
I want to search for books by different criterias

Background:

Given the following books				SearchSteps.theFollowingBooksWithTable(Table)	
Author	Title	Year	Publisher		
Martin Fowler	Patterns of Enterprise Application Architecture	2002	Addison Wesley		
Eric Evans	Domain Driven Design	2003	Addison Wesley		
Gerard Meszaros	xUnit Test Patterns	2007	Addison Wesley		

Scenario: Search for title

When I search for title 'Patterns'	SearchSteps.searchForTitle(String)
Then the result list should contain 2 books	SearchSteps.checkSearchResultCount(int)

Scenario: Search for author

When I search for author 'Fowler'	SearchSteps.searchForAuthor(String)
Then the result list should contain 2 books	SearchSteps.checkSearchResultCount(int)

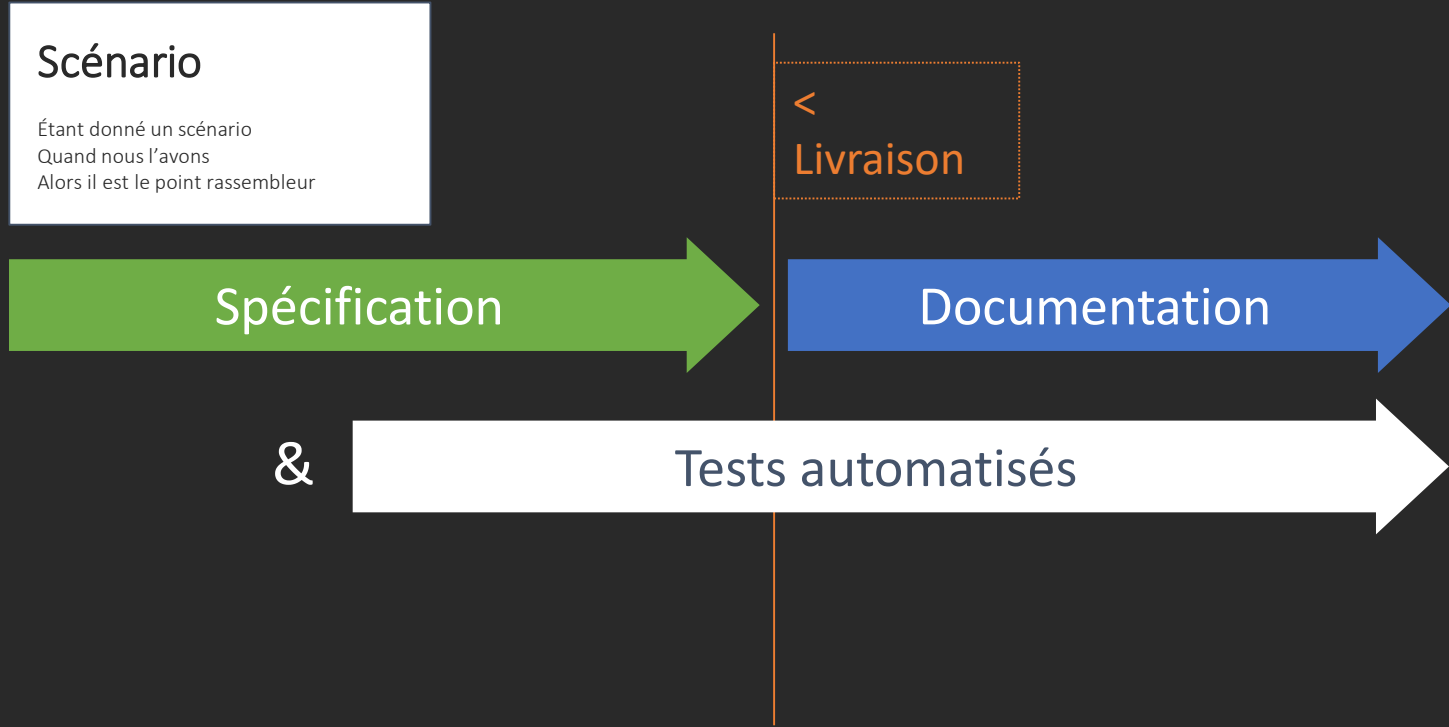
java.lang.AssertionError: Result contains wrong count. expected:<2> but was:<1>



=



Un scénario pour tous!



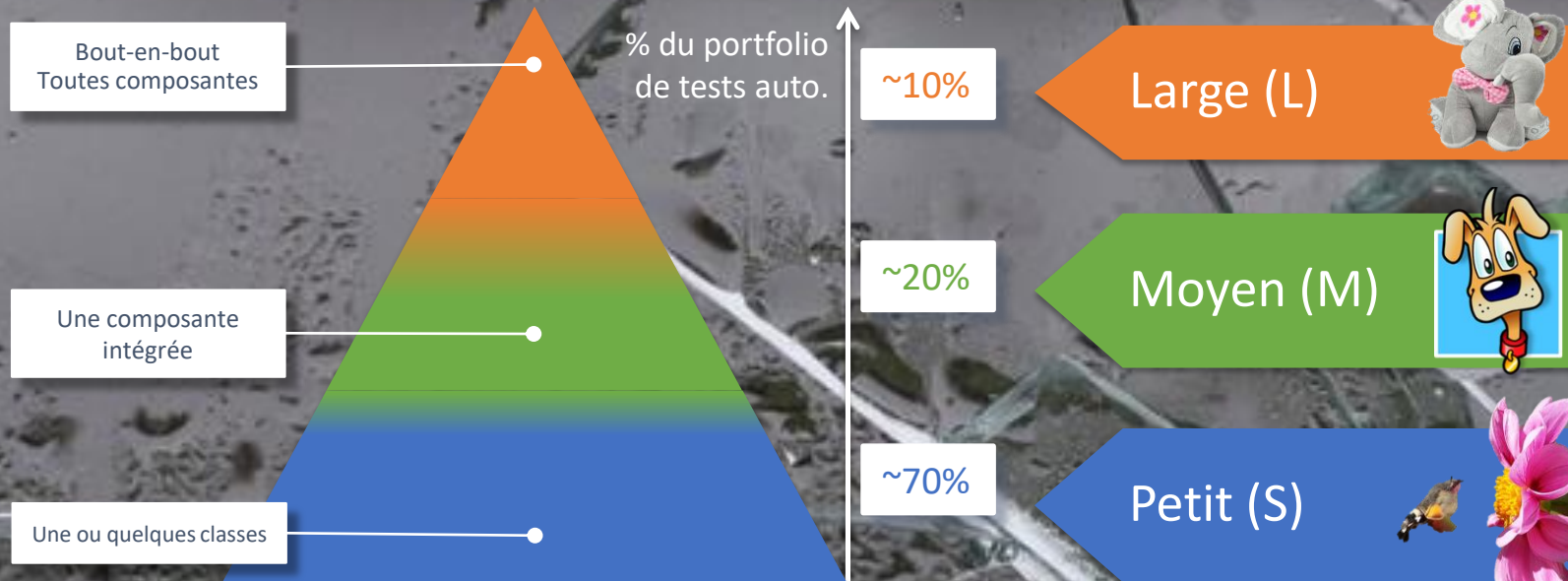
Tests fonctionnels automatisés

≠

Tests bout-en-bout via l'interface et la BD

Attention en automatisant !

Fragilité des tests



Conclusion

En résumé...

1

En découvrant **ensemble** les scénarios et les règles, nous bâtissons une **compréhension commune** et **forte**.

2

Les scénarios servent **d'exemples** pour **piloter le développement**.

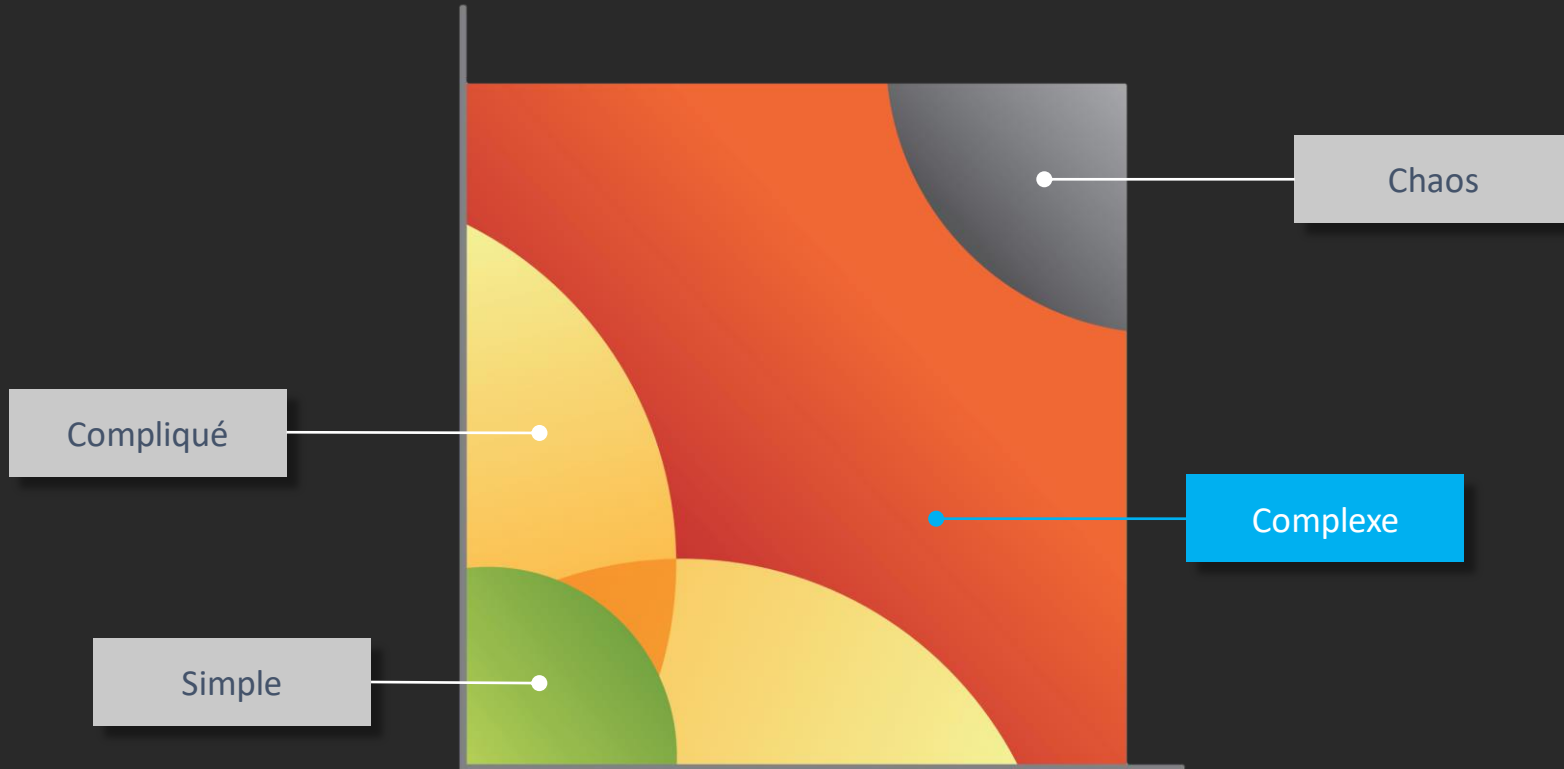
3

Les scénarios sont attachés à des **tests automatisés** qui démontrent **l'avancement** et **préviennent la régression**

4

Les scénarios et règles **documentent** la **fonctionnalité** de manière **permanente** et **vivante**...

Quand utiliser ou ne pas utiliser le BDD?



Le BDD est particulièrement utile pour découvrir
et spécifier des règles impliquant de la logique
d'affaires complexe

merci .



Toutes nos présentations

conferences.elapsetech.com

Diapositives et références

[conferences.elapsetech.com
/bdd-spec-affaires](https://conferences.elapsetech.com/bdd-spec-affaires)

Félix-Antoine Bourbonnais

Pascal Roy

Site

elapsetech.com

Twitter

[@fbourbonnais](https://twitter.com/fbourbonnais)

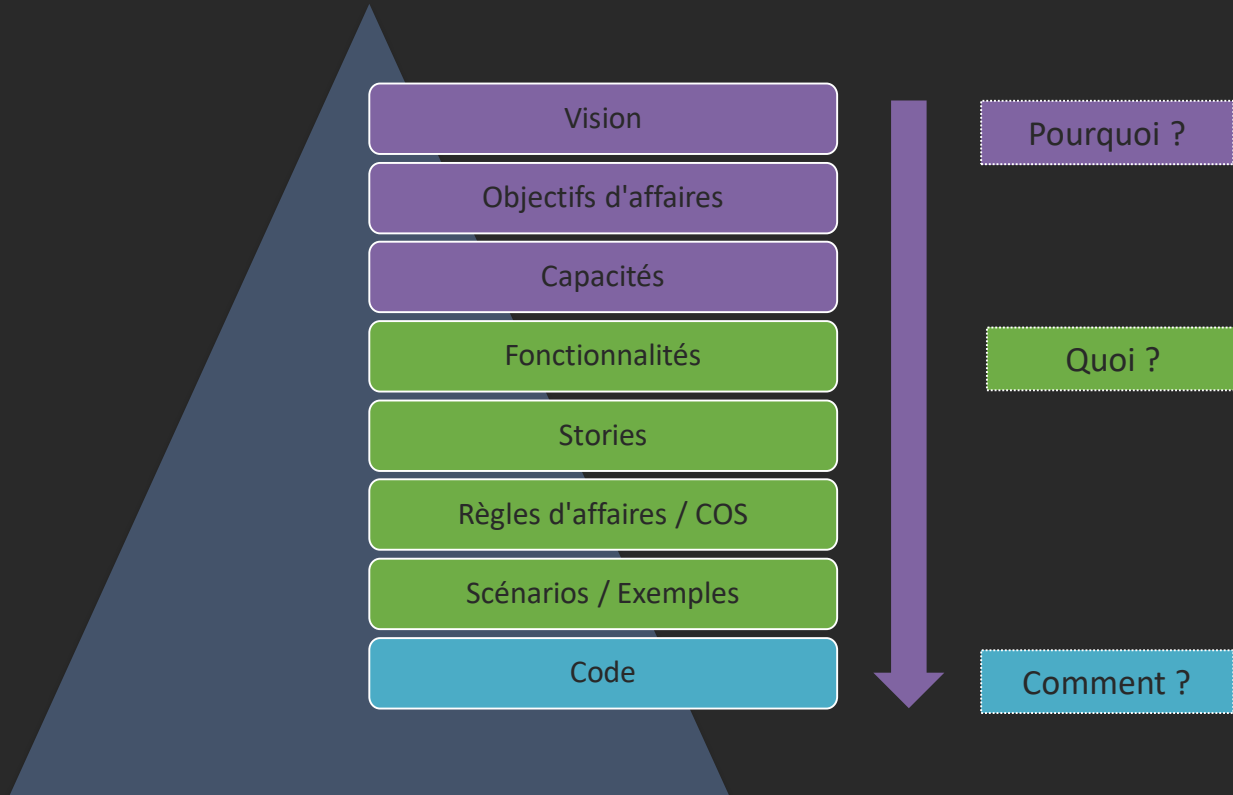
Courriel

fbourbonnais@elapsetech.com

pascalroy@elapstech.com

Annexes

Découpage vers une fonctionnalité



Quand et où dans mon processus de développement?

