

Experts en développement
logiciel et formation –
Approches AGILE

Software Development
and Consulting Specialists –
AGILE Approaches



Dites-moi, Monsieur le Client...

Techniques d'entretien

François Beauregard – fbeauregard@pyxis-tech.com



© Scott Adams, Inc./Dist. by UFS, Inc.

C'est la première fois que je donne cette présentation.

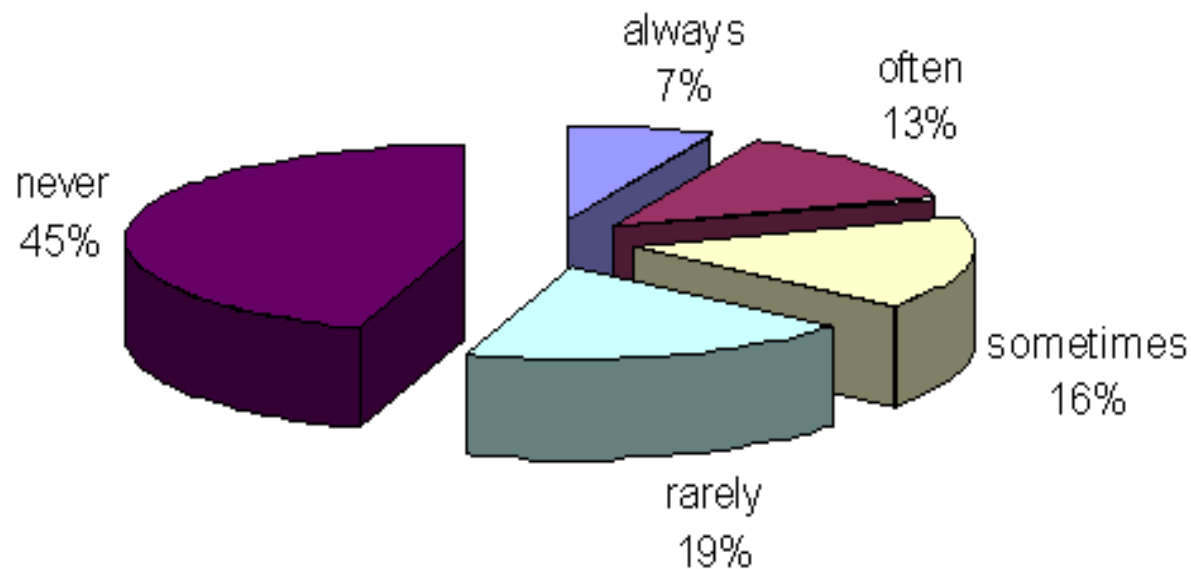
Cette présentation contient surtout des éléments très simples.

La collaboration est un sujet très vaste impossible à couvrir en 90 minutes. Nous allons nous concentrer sur des aspects précis.

- Présenter des techniques et des pratiques concernant les entretiens avec un client
- Vous sensibiliser à l'importance de ces pratiques
- Fournir des éléments de réflexion pour la mise en œuvre d'améliorations dans vos projets
- Vous donner l'occasion de mettre en pratique les éléments présentés

- **État des lieux**
- Défis de la gestion des exigences
- Rappels concernant l'approche Agile et les scénarios utilisateurs (*user stories*)
- Techniques d'entretien
- Biens livrables
- Atelier
- Conclusion

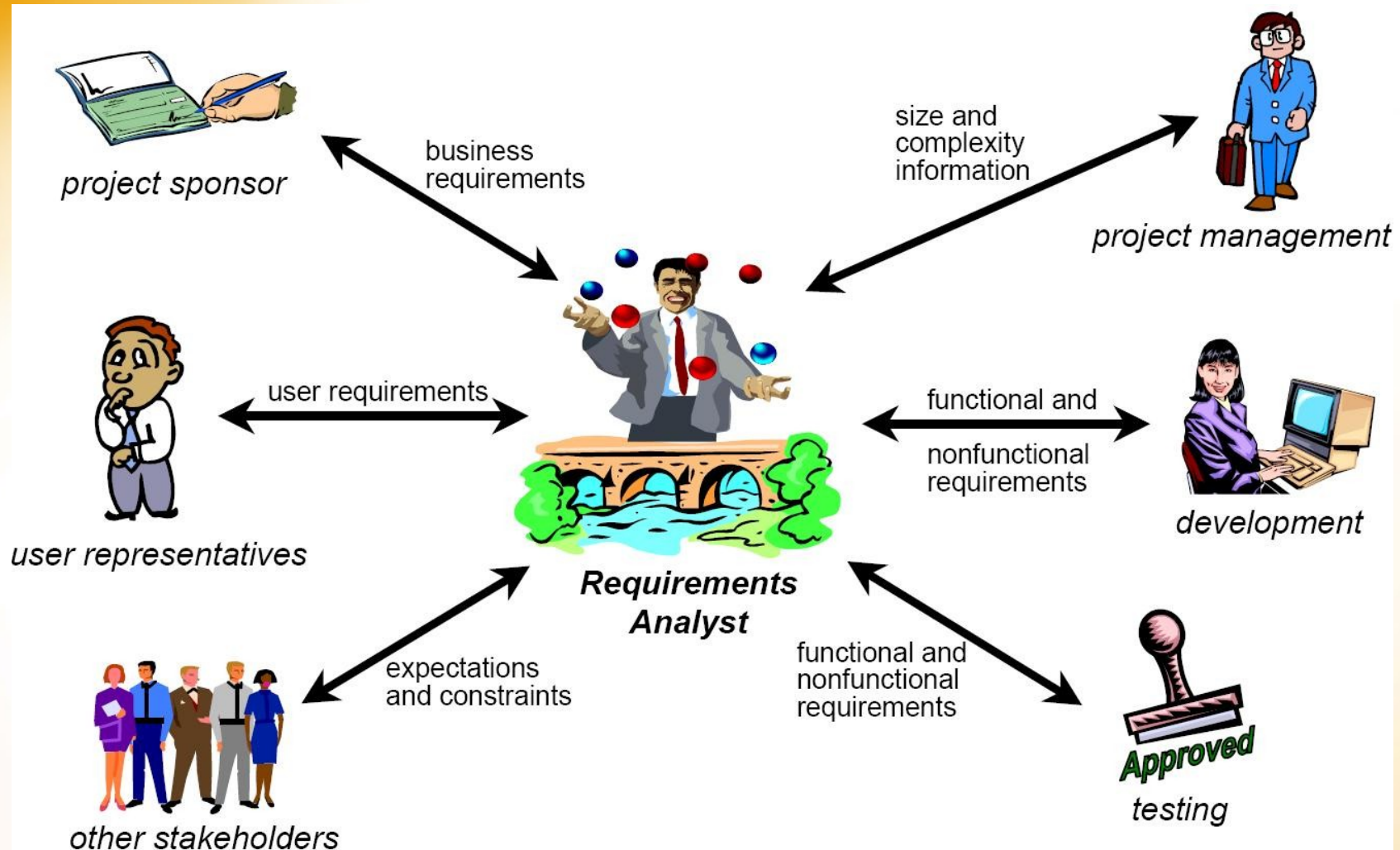
Actual Use of Requested Features



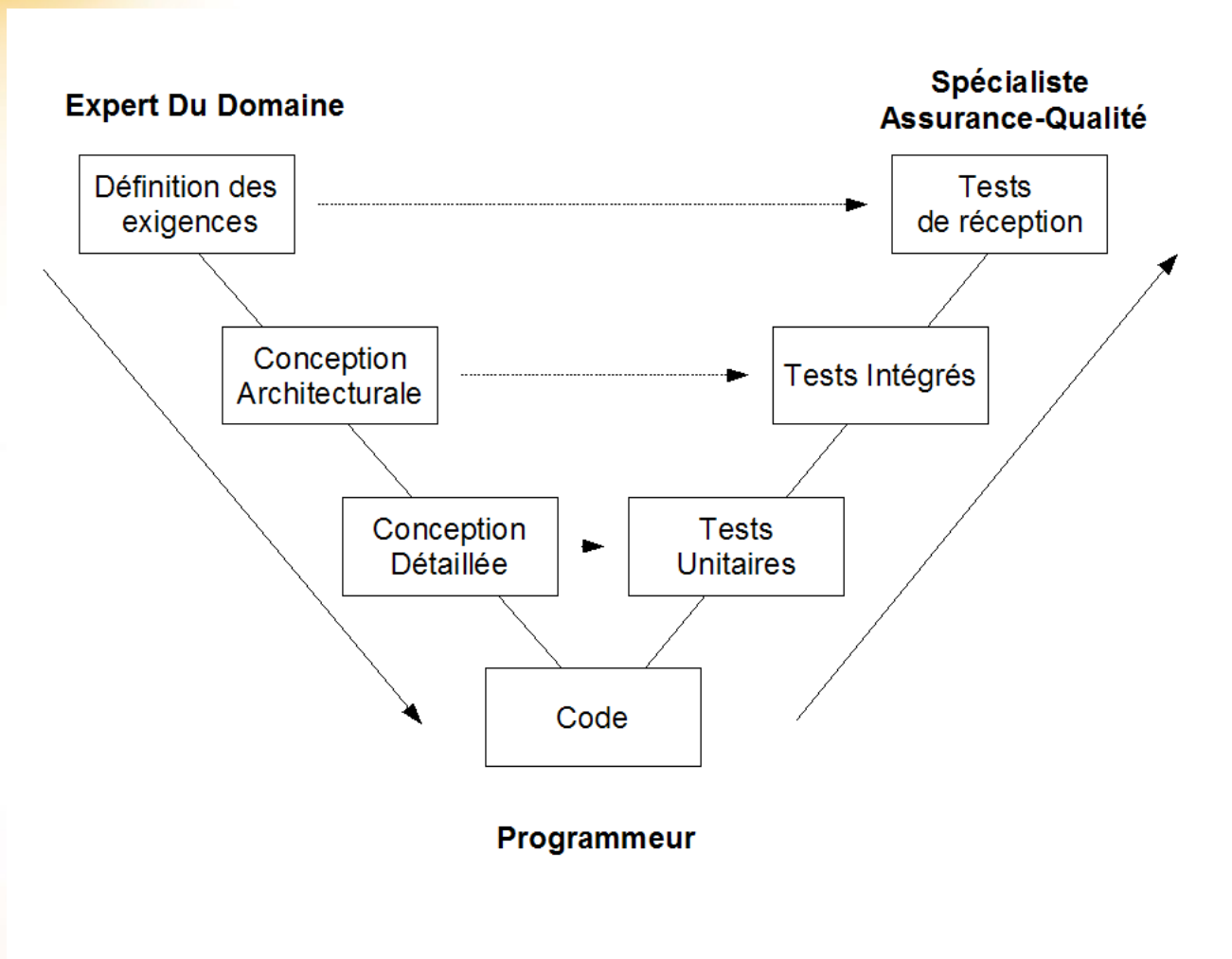
Dites-moi pourquoi

- **Le développement et la gestion des exigences logicielles se résument essentiellement à une problématique de communication.**
- Ceux qui demandent le logiciel doivent communiquer avec ceux qui le construisent.





Source : Karl E. Wiegiers



- Dans une approche traditionnelle ayant un cycle en V, la définition des exigences se fait dans un **document écrit** en langage naturel par un expert du domaine.
- Les scénarios permettant de valider le code développé sont écrits dans un autre document par des experts en assurance qualité, avec un **formalisme spécialisé**. De bons experts en assurance qualité doivent avoir une double compétence et à cause de cela ils sont rares et onéreux.
- Le code de l'application est développé après lecture des exigences fonctionnelles et validé par la suite par les scénarios de test, le plus souvent **manuellement**.
- Les sources d'information et les intervenants sont **multiples** et soulèvent la question de la **fiabilité de l'interprétation**, de la **synchronisation de l'information**, de l'**efficacité** et de l'optimisation du procédé.

- Les documents sont multiples.
 - Beaucoup d'effort pour s'assurer que tout est tenu à jour
 - Traçabilité difficile
- Il est difficile de tester des documents!
- La gestion des exigences et la gestion de projet sont souvent mal intégrées.



- État des lieux
- Défis de la gestion des exigences
- **Rappels concernant l'approche Agile et les scénarios**
- Techniques d'entretien
- Biens livrables
- Atelier
- Conclusion

Les approches Agiles telles que XP et Scrum donnent un rôle et des responsabilités au client.

Quels sont-ils?

- Les gens discutent, mais est-ce qu'ils écrivent?
- Écrivent-ils ce qu'ils pensent?
- Quel est le niveau de détail nécessaire?
- Est-ce qu'il y a vraiment UN client?

"... one of the most effective ways of specifying something is to describe (in detail) how you would accept (test) it..."

Bill Hetzel, <1990

User Stories :

Les trois C de Ron Jeffries

Carte

- Les scénarios sont traditionnellement écrits sur des cartes.
- Les cartes peuvent être annotées avec ce qui suit : estimation, notes, etc.

Conversation

Les détails concernant le scénario ressortent durant des conversations avec le client

Confirmation

Les conditions de satisfaction et les tests de réception confirment que le scénario a été développé correctement.

Source : XP Magazine 8/30/01, Ron Jeffries

En tant que client, je peux payer le contenu de mon panier d'achat avec ma carte VISA.

COS :

- Le panier est vide.
- Le détail de l'achat est sauvegardé.
- La transaction est acceptée par VISA.

Notes :

- Se reporter au document : Certifié par VISA

Devant

Derrière

- Tester avec un numéro de carte valide
- Tester avec un numéro de carte non valide
- Tester avec une carte expirée
- Tester un dépassement de limite (transaction refusée)

Utilisez plusieurs modèles pour
supporter les activités.

Il y a beaucoup plus que les
cartes!

- État des lieux
- Défis de la gestion des exigences
- Rappels concernant l'approche Agile et les scénarios utilisateurs
- **Techniques d'entretien**
- Biens livrables
- Atelier
- Conclusion

Faire de bons ateliers et avoir de bons entretiens avec un client c'est simple mais pas facile.

- **Préalables**
 - Avoir les bonnes personnes (Qui sont-elles?)
 - Avoir un bon facilitateur
 - Mettre en place un environnement qui favorise la collaboration
 - Respect et confiance
 - Processus de prise de décision
 - Règles de base
 - Endroit partagé (salles, murs, tableaux)
 - Outils appropriés (cartes, *Post-It*, caméra numérique, wiki, etc.)

■ Préparation

- Avoir un objectif clair. Il existe plusieurs types de d'ateliers (*workshop*) – « Charter »
Exploration du périmètre, établissement des priorités, précision des besoins, révision, etc.
- Transmettre l'information adéquate aux participants
- Préparer les activités (ouverture, déroulement, fermeture)
- Préparer la logistique avec soin
- Déterminer les biens livrables prévus

"Facilitation is the art of leading people through processes toward agreed-upon objectives in a manner that encourages participation, ownership, and productivity from all involved."

David Sibbet

- Conseils pour la discussion
 - Utilisez des formulations précises.
 - Lorsque l'expert dit : « Nous faisons généralement suivre X par Y. »
 - Poursuivez avec : « OK, donc vous faites toujours suivre X par Y. »
 - Il s'agit d'une forme d'écoute active.
 - Confirmez par la négation de l'opposé.
 - Lorsque l'expert dit : « Nous avons toujours X si Y est présent. »
 - Poursuivez avec : « OK, donc si Y est absent vous n'avez pas X. »
 - Il s'agit d'une façon qui permet de sortir notre expert du « pilote automatique ».
 - Cherchez un *pattern*.
 - Lors de l'exploration d'un nouveau processus, essayez d'appliquer le *pattern* d'un autre processus. Demandez ce qui fonctionnerait. Demandez aussi ce qui ne fonctionnerait pas et pourquoi.
 - N'ayez pas peur du ridicule!

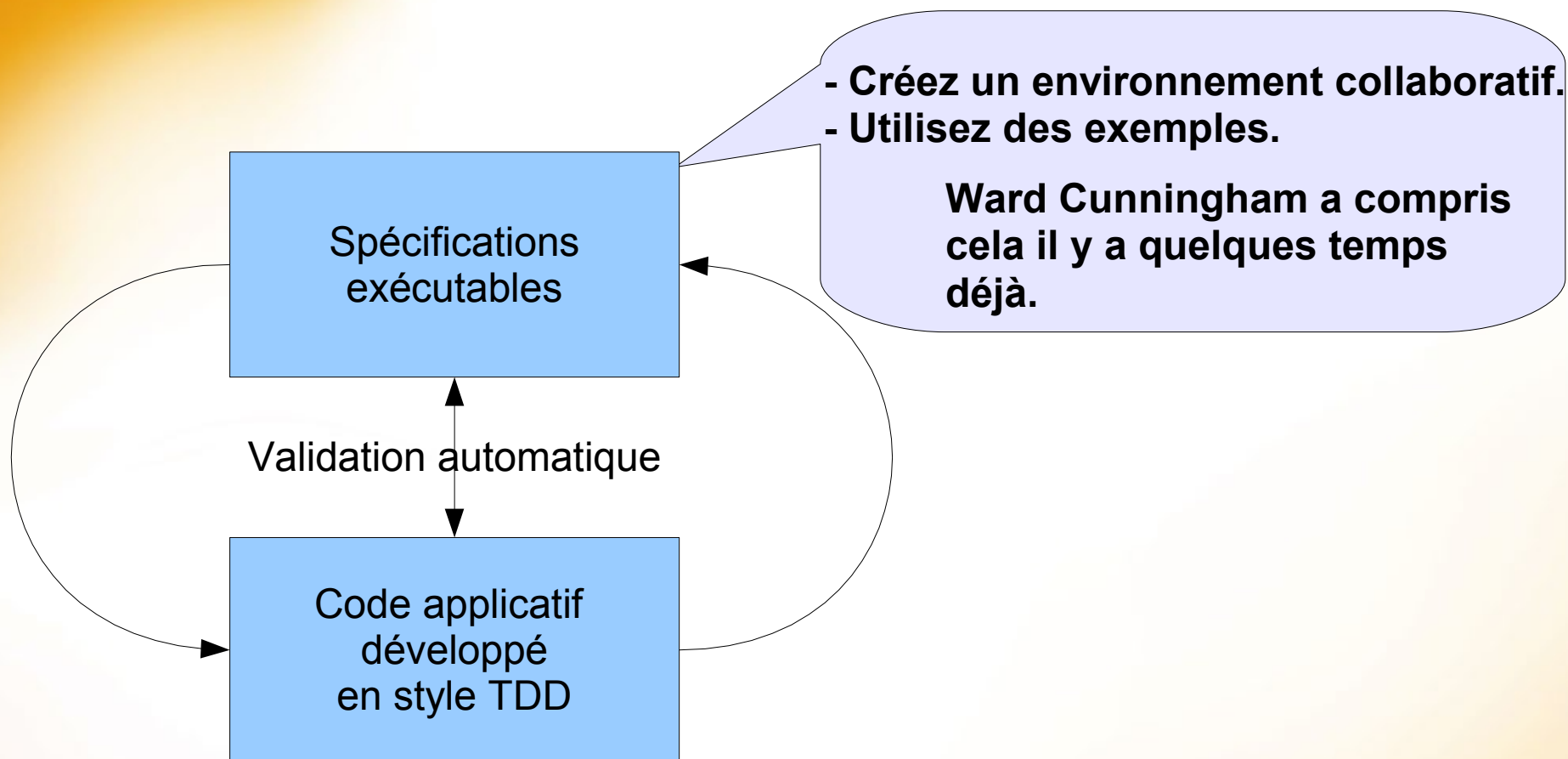
- **Conseils pour la discussion**
 - **Demandez des exemples.**
 - Commencez par demander des exemples de la situation la plus commune.
 - Demandez ensuite des exemples de cas moins fréquents et de cas limites.
 - **Créez des exemples.**
 - Proposez des exemples dans le but précis de valider une règle, de vérifier un résultat ou de valider une partie particulière d'un processus.
 - **Dessinez.**
 - N'hésitez pas à aller au tableau pour dessiner un diagramme, une table de vérité, etc.

Je reviendrai plus tard sur les exemples!

- Autres conseils
 - Utilisez un *parking lot*.
 - Variez les modes de collaboration.
 - Prenez soin de bien clore l'activité.
 - Ayez du plaisir!

- État des lieux
- Défis de la gestion des exigences
- Rappels concernant l'approche Agile et les scénarios utilisateurs
- Techniques d'entretien
- **Biens livrables**
- Atelier
- Conclusion

- Que doit-on conserver et où?
- L'important n'est pas tant la forme de l'information mais qu'elle soit partagée, compréhensible par les intervenants et qu'elle puisse évoluer et servir de soutien à des conversations futures.
- Dans le cadre de l'utilisation de scénarios, il est capital de consigner les conditions de satisfaction du client (confirmations).



- Le cycle optimisé vise à éliminer les inefficacités du procédé traditionnel
 - En améliorant la collaboration entre intervenants
 - En fiabilisant l'interprétation de l'information transférée
 - En fiabilisant la synchronisation des artefacts produits
 - En automatisant la validation du code
 - En rationalisant les sources d'information (principe de l'information à un seul endroit)
 - En détectant la non-conformité du code et des exigences
 - En augmentant la rétroaction

- Les tests de réception agissent comme la spécification des fonctionnalités (exigences).
- Les tests deviennent la documentation des exigences.
 - Un document **sans ambiguïté**
 - Qui ne peut se désynchroniser du projet
 - Qui est exécutable
- Une fonctionnalité n'est pas complétée tant que l'ensemble des tests de réception pour cette fonctionnalité ne s'exécute pas correctement.
- Il n'existe pas d'achèvement partiel.

- Les tests sont faits dans l'optique de spécifier (plutôt que de vérifier).
- Les tests servent à savoir à quel moment quelque chose est complétée. (Il devrait y avoir une intégration serrée dans la gestion de projet.)
- Les tests ne sont pas reportés à la fin du processus.
 - En fait, l'assurance qualité est déjà présente au début du processus pour aider à la spécification.



Toujours avec moi...

- Le système doit supporter l'addition de deux nombres naturels.
- Comment tester cela :
 - Tapez 3 dans le champ de saisie.
 - Cliquez sur le bouton **+**.
 - Tapez 7 dans le champ de saisie.
 - Cliquez sur le bouton **=**.
 - Vérifiez que le résultat affiché est **10**.

- Est-ce qu'il y a une meilleure façon?

Règle pour la calculatrice		
Opérande 1	Opérande 2	Total?
7	3	10
-3	7	4
6	0	6
-5	3	-2

- Le système doit également supporter la division.
- Est-ce qu'il y a une meilleure façon?

Règle pour la calculatrice			
Opérande 1	Opérande 2	Total?	Quotient?
7	3	10	2.33
-3	7	4	-0.43
6	0	6	Erreur
-5	3	-2	-1.67
0	5	5	0



Exemple plus significatif

Un crédit allant jusqu'à 1000 \$ est accordé à un client qui fait affaire avec nous depuis plus de 12 mois s'il a été un bon payeur durant cette période et s'il a un solde à payer de moins de 6000 \$.

Règle pour la vérification du crédit				
Mois	Fiable	Solde	Crédit accordé?	Limite de crédit?
14	Oui	5000 \$	Oui	1000 \$
0	Oui	0 \$	Non	0 \$
24	Non	0 \$	Non	0 \$
18	Oui	6000 \$	Non	0 \$
12	Oui	5500 \$	Non	0 \$



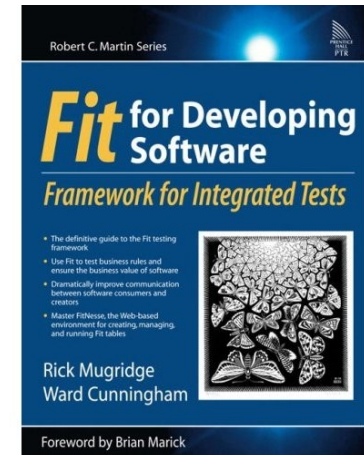
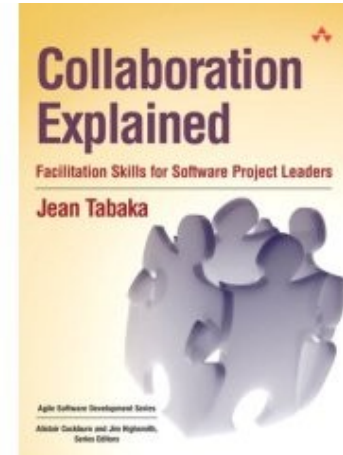
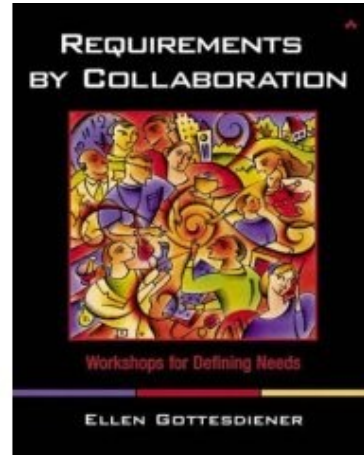
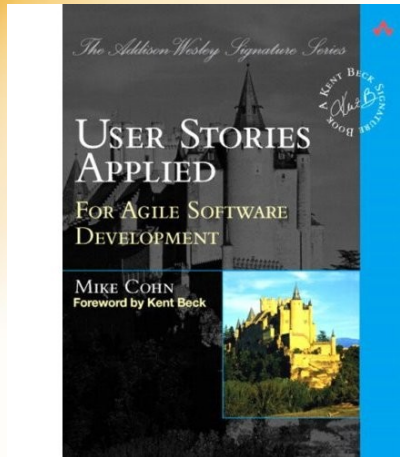
Atelier...

Échange sur les résultats

"When customers and developers recognize that they depend on each other, the team can steer the project so that there is always the most business value implemented at any given moment." and productivity from all involved."

XP Installed

- La **confiance** et la **collaboration** sont les clés du succès.
- Vous devez avoir les bonnes personnes **dans** l'équipe.
- Soyez créatif.
- Documentez ce qui est nécessaire.
- Attention : Les outils peuvent renforcer la collaboration mais ils peuvent aussi la réduire.
- **Concentrez-vous sur la valeur en livrant continuellement des logiciels fonctionnels.**



greenpeppersoftware.com

Merci!
Des questions?

■ **Centre de développement**

- Développement sur mesure
- Architecture et conception
- Méthodes de test
- Évaluation et gestion de projet



GreenPepper™ simplifie grandement la création concertée de tests de réception et de scénarios exécutables. De plus, il offre de grandes capacités de traçabilité des besoins.

■ **Accompagnement Agile**

- Gestion du changement — formation
- Transition Agile
- Scrum
- Modélisation Agile
- Tests (stratégie, fonctionnels, TDD, etc.)

■ **Services-conseils en développement**

- Développeurs chevronnés
- Accompagnement technologique (Java, .NET, Ruby)
- Exploration de solutions d'amélioration
- Sélection et configuration d'outils
- Gestion de projet, évaluation de mandat
- Audit et recommandations
- Rétrospective