

Mise en oeuvre d'une méthode Agile



<http://www.sebastienchabal.fr/wp-content/uploads/2010/08/meteoBlackBlue.jpg>

Scrum

(Ce cours suppose les principes d'XP connus)

M. Blay-Fornarino

63

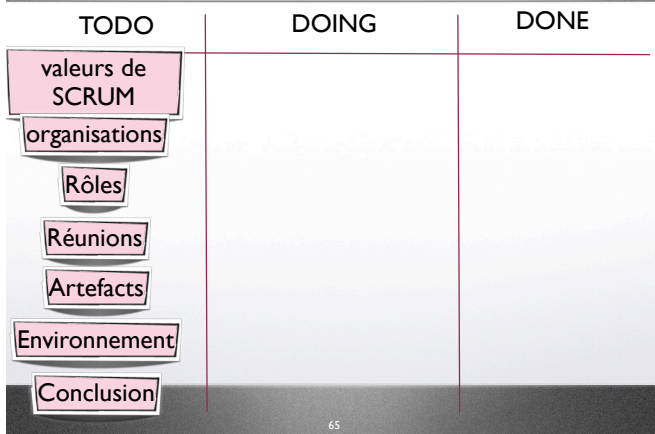
Contrôler le Chaos ...

SCRUM

- Met l'accent sur la gestion de projet
- Met l'accent sur les caractéristiques livrées et l'ajustement selon les résultats
- L'objectif est de trouver un équilibre entre permettre au métier de changer d'approche et à l'équipe de développement de faire un travail de qualité à sa portée.

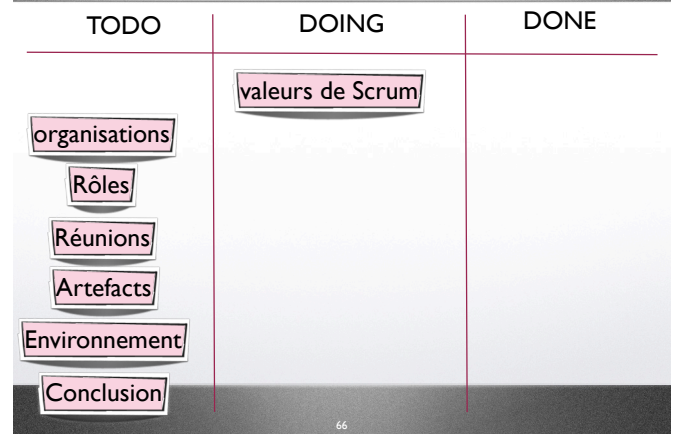
64

Backlog



65

Backlog



66

Scrum en 100 mots

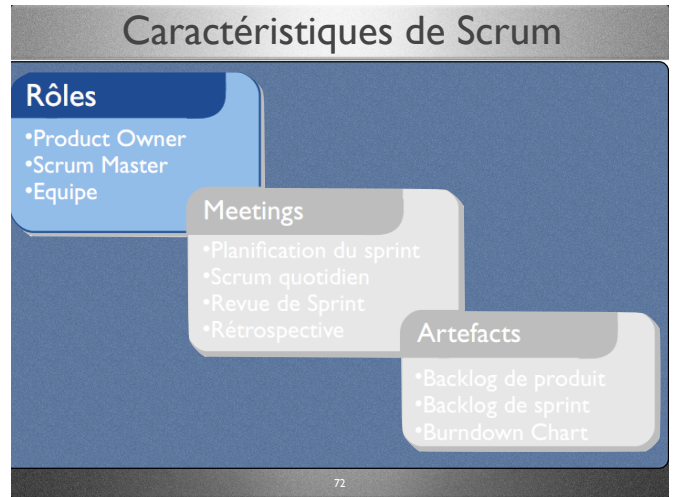
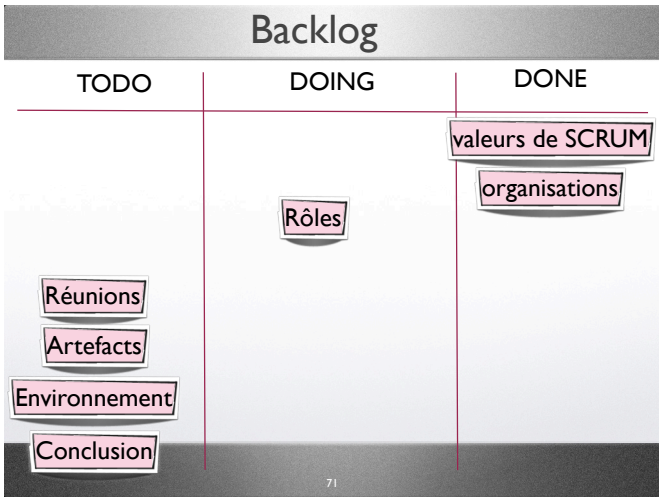
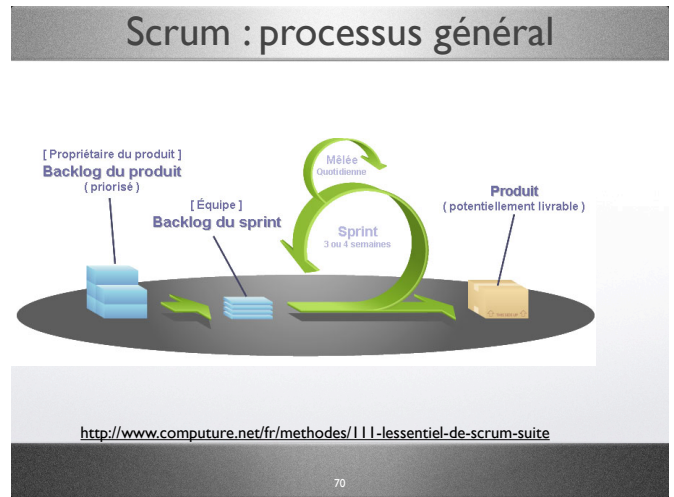
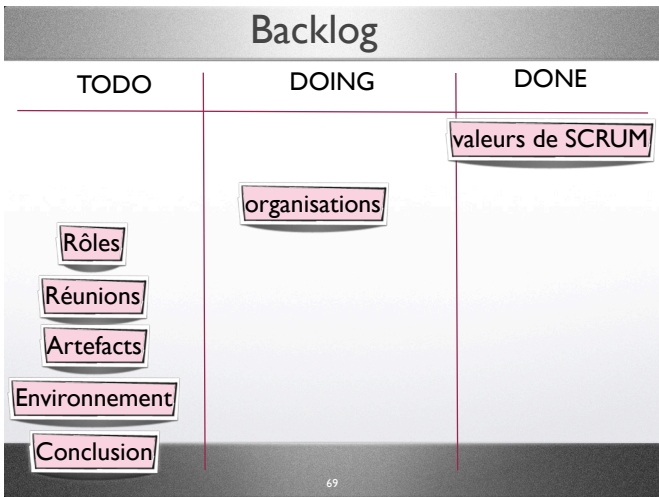
- Scrum est un processus agile qui vise à produire la plus grande valeur métier dans la durée la plus courte.
- Un **logiciel qui fonctionne est produit à chaque sprint** (toutes les 2 à 4 semaines).
- Le **métier définit les priorités**.
- L'équipe s'organise elle-même pour déterminer la meilleure façon de répondre aux exigences les plus prioritaires.
- A chaque fin de sprint, tout le monde peut voir fonctionner le produit courant et décider soit de le livrer dans l'état, soit de continuer à l'améliorer pendant un sprint supplémentaire.

67

Les valeurs de SCRUM

- **Engagement.** Soyez prêt à vous engager sur un objectif. Scrum assure aux développeurs l'autorité dont ils ont besoin pour remplir leurs engagements.
- **Focus.** Faites votre travail. Concentrer tous vos efforts et vos compétences à faire le travail que vous vous êtes engagé à faire. Ne vous inquiétez pas d'autre chose
- **Transparence.** Scrum laisse tous les éléments d'un projet visibles à tous.
- **Respect.** Les individus sont façonnés par leurs antécédents et leur expérience. Il est important de respecter les différentes personnes qui composent une équipe.
- **Courage.** Ayez le courage de vous engager, d'agir, d'être ouvert et d'attendre du respect

68



Les Rôles : le Product Owner

- Définit les fonctionnalités du produit
- Choisit la date et le contenu de la livraison
- Responsable du retour sur investissement
- Définit les priorités dans le backlog en fonction de la valeur « métier »
- Ajuste les fonctionnalités et les priorités à chaque sprint si nécessaire
- Accepte ou rejette les résultats

© Scrum.org

73

Les Rôles : le Scrum Master

- Représente le management du projet
- Responsable de faire appliquer par l'équipe les valeurs et les pratiques de Scrum
- Résout des problèmes
- S'assure que l'équipe est complètement fonctionnelle et productive
- Facilite une coopération poussée entre tous les rôles et fonctions
- Protège l'équipe des interférences extérieures

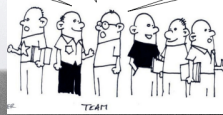
© Scrum.org

74

Les Rôles : l'équipe

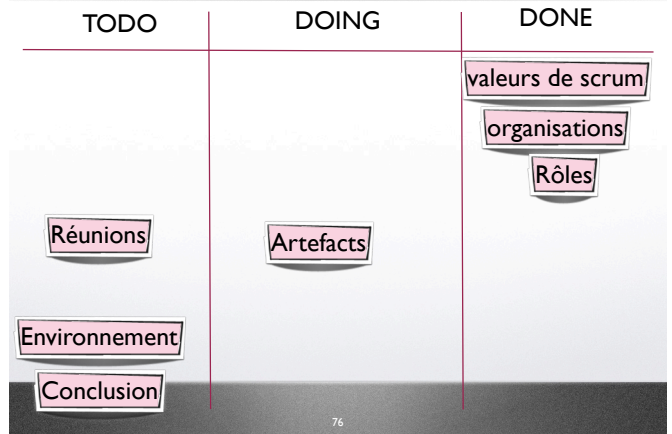
- De 5 à 10 personnes
- Regroupant tous les rôles
 - Architecte, concepteur, développeur, spécialiste IHM, testeur, etc.
- A plein temps sur le projet, de préférence
 - Exceptions possibles (administrateur, ...)
- L'équipe s'organise par elle-même
- La composition de l'équipe ne doit pas changer pendant un Sprint

Martin (2003): "The team is in it for the long term. They work hard, at a pace that can be sustained indefinitely. They conserve their energy, treating the project as a **marathon** rather than a **sprint**."



75

Backlog



76

Caractéristiques de Scrum

Rôles

- Product Owner
- Scrum Master
- Equipe

Meetings

- Planification du sprint
- Scrum quotidien
- Revue de sprint
- Rétrospective

Artefacts

- Backlog de produit
- Backlog de sprint
- Burndown Chart

77

Artefacts : le Product Backlog ou PDL (Plan de Développement Logiciel)



- Recueil des exigences et des "User Stories"
 - « En tant que <rôle>, je veux pouvoir <besoin>, afin de <bénéfice, plus-value> »
- Etabli et maintenu par le Product Owner
- Chaque élément doit apporter une valeur ajoutée au produit final (pas de superflu)
- Les "importances" sont définies par le Product Owner
- Les "importances" sont revues à chaque Sprint
- Les plus importantes sont les plus détaillées.

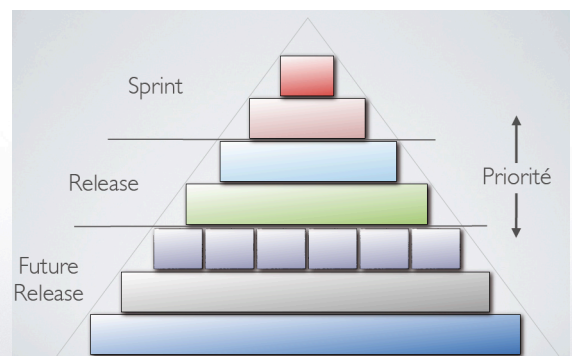
78

Backlog



79

Iceberg du product backlog



80

BACKLOG DE PRODUIT

- Les exigences du produit
- Toutes (Idées, fonctionnalités, Epic, Thème, etc.)
- Exprimées en User Stories
- Le PO le maintient organisé
- Toujours estimé et avec les priorités

81

Yannick Quenec'hdu
yquenechdu@gmail.com

Les fonctionnalités sont décrites
sous forme de :
USER STORIES

**Je suis <rôle>,
Je veux <fonction>**

82

Yannick Quenec'hdu
yquenechdu@gmail.com

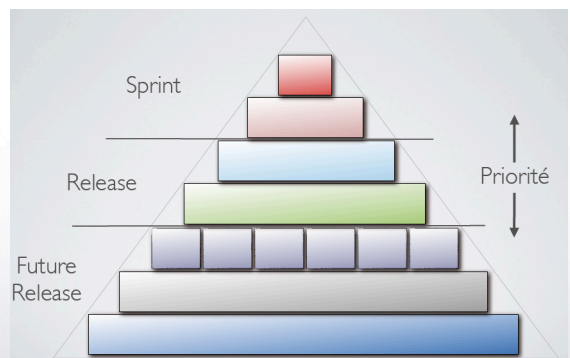
Les fonctionnalités sont validées par des
Critères d'acceptation

**Je suis <rôle>,
Je veux <fonction>
et <un résultat>**

83

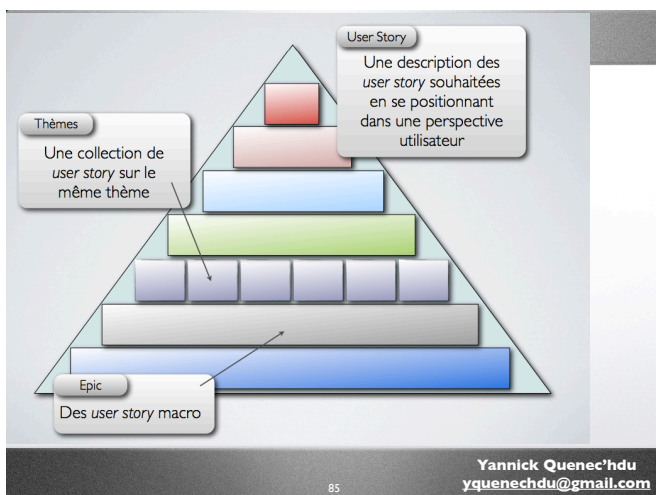
Yannick Quenec'hdu
yquenechdu@gmail.com

Iceberg du product backlog



84

Yannick Quenec'hdu
yquenechdu@gmail.com



85

Yannick Quenec'hdu
yquenechdu@gmail.com

User stories / (Histoires d'utilisateur)

- Agile est associé à l'expérience utilisateur
- On raconte une histoire : User stories
- On montre notre histoire : Prototype

86

Yannick Quenec'hdu
yquenechdu@gmail.com

Recommandation

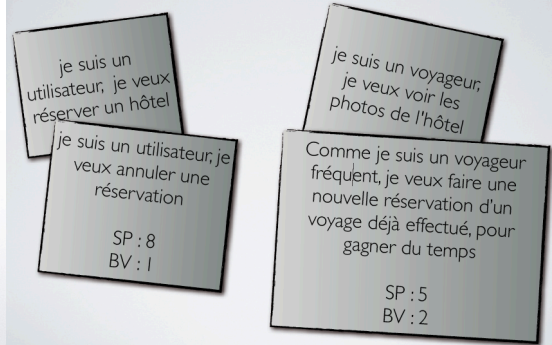
Utiliser ce masque

Je suis « Rôle de l'utilisateur »,
je veux « But »,
les « conditions » pour ...

87

Yannick Quenec'hdu
yquenechdu@gmail.com

EXEMPLE DEPUIS UN SITE DE VOYAGES



88

Yannick Quenec'hdu
yquenechdu@gmail.com

Détails

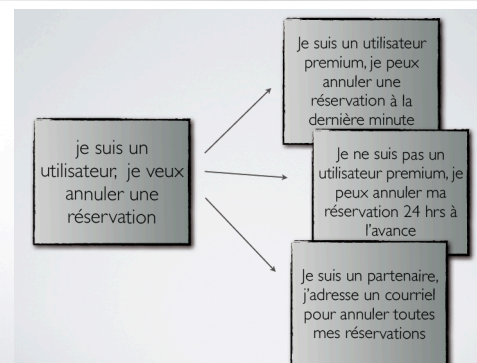
Je suis un utilisateur, je veux annuler une réservation

- Le client reçoit un remboursement complet ou partiel
 - Je rembourse directement sur son compte ou à l'intermédiaire
- Comment doit fonctionner l'annulation d'une réservation ?
 - C'est le même principe pour tous les hôtels ?
 - Un voyageur fréquent peut-il annuler après sa réservation
- Une confirmation est adressée à l'utilisateur ?
 - Comment ?

89

Yannick Quenec'hdu
yquenechdu@gmail.com

LES DÉTAILS SONT AJOUTÉS EN PETITES USER STORIES

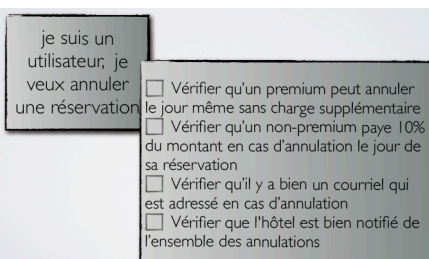


90

Yannick Quenec'hdu
yquenechdu@gmail.com

LES DÉTAILS SONT AUSSI UNE CONDITION DE SATISFACTION

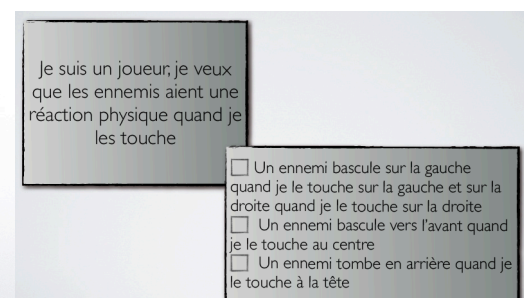
- Le product owner peut ajouter aux users stories des conditions de satisfaction
- Ce sont essentiellement des vérifications



91

Yannick Quenec'hdu
yquenechdu@gmail.com

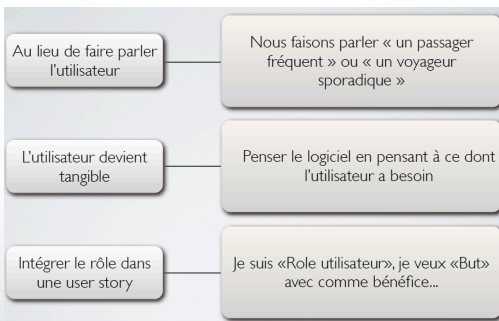
UN EXEMPLE DE JEUX DE DÉVELOPPEMENT



92

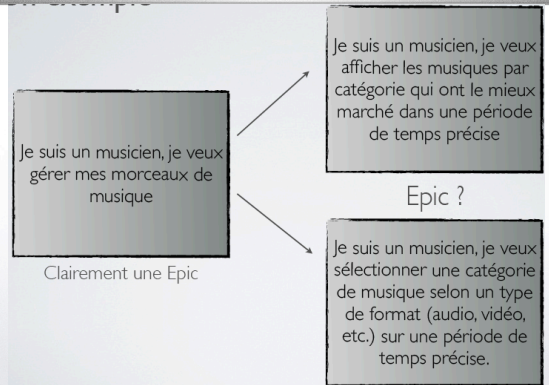
Yannick Quenec'hdu
yquenechdu@gmail.com

Utiliser des rôles



93

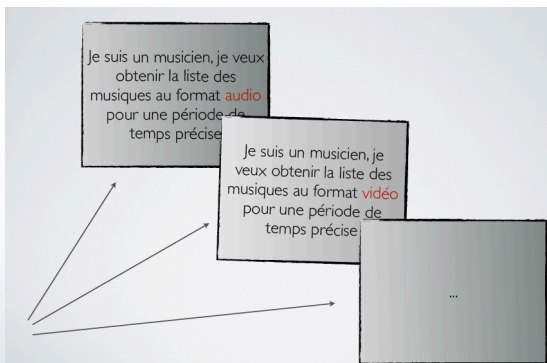
EPIC



Yannick Quenec'hdu
yquenechdu@gmail.com

94

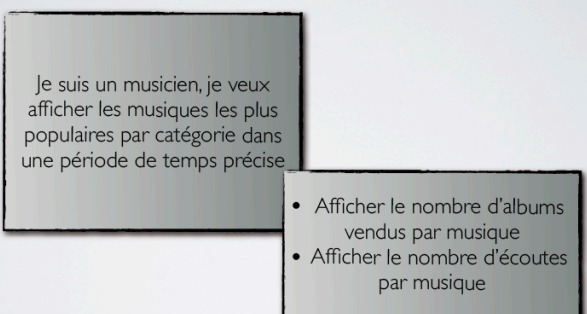
EPIC vers User stories



Yannick Quenec'hdu
yquenechdu@gmail.com

95

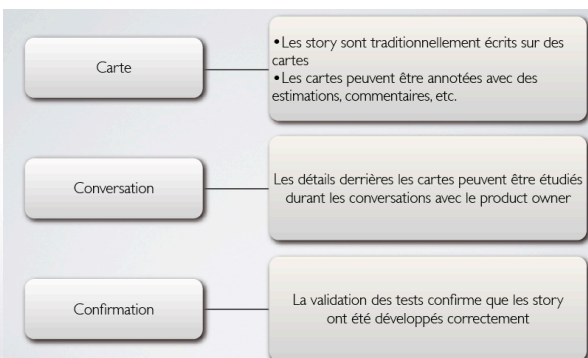
TERMINER, EN AJOUTANT LES CONDITIONS DE SATISFACTION



Yannick Quenec'hdu
yquenechdu@gmail.com

96

Bonnes User stories & les 3 C



Yannick Quenec'hdu
yquenechdu@gmail.com

97

Une bonne «story»

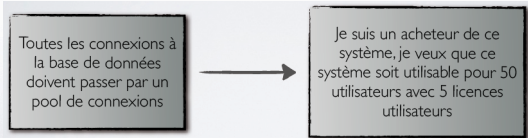
- INVEST →
- I - Indépendant
 - N - Négociable
 - V - Valeur
 - E - Estimable
 - S - Size (Taille)
 - T - Testable

Yannick Quenec'hdu
yquenechdu@gmail.com

98

Une bonne «story» : Valeur

Elle doit être réécrite pour montrer le bénéfice pour les utilisateurs et les clients



99

Yannick Quenec'hdu
yquenechdu@gmail.com

Une bonne «story» : Taille



- Des petites user stories pour un futur proche
- Macro (Epic) pour les prochaines
- Les user stories sont progressivement affinées dans le temps, plus elles s'approchent de la fin,
- Deux types de grandes user stories
 - Les user stories complexes : intrinsèquement grande et sans possibilités de les réduire
 - Les user stories combinées : Plusieurs user stories combinées en une seule

100

Yannick Quenec'hdu
yquenechdu@gmail.com

Une bonne «story» : Testable

- Un test démontre que la user stories correspond aux besoins du client



101

Yannick Quenec'hdu
yquenechdu@gmail.com

Une bonne «story» : Testable

SMART

- Spécifique - défini et explicite
- Mesurable - quantifiable et mesurable
- Atteignable - qui peut être réalisé et validé
- Relevante - pertinente pour la story
- Temporaire - limité dans le temps

Facilite la rédaction des critères d'acceptation

102

Yannick Quenec'hdu
yquenechdu@gmail.com

User Stories versus Use cases

- Une "User story" : un **but**, pas une **séquence d'actions**.
- Une "User story" souvent seulement **un des scénarios** du "Use case".
- Une "User story" émerge plus **rapidement** => **moins** d'analyse
- Les "User stories" plus **faciles à lire** (Mouais...)
- Les "User stories" sont rédigées (en principe) sur des cartes: leur durée de vie est limitée et elles n'ont pas pour vocation à être conservées (contrairement aux "Use cases"). Or, ce besoin de traçabilité peut être réel.
- Les "User stories" reposent sur un **mode oral, collaboratif**, de proximité : elles sont discutées (Customer / Developer) et tout n'est pas rédigé (contrairement aux "Use cases") => Pbme dans le cas de gros projets, équipe importante, offshore...
- Une "User story" doit être implémentée et testée en une itération. Un "Use case" peut être traité sur plusieurs itérations (scénario nominal sur une, scénarios alternatifs sur une autre) en fonction des risques à lever.
- ...

103

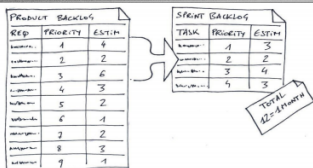
«Product Backlog»

SPRINT BACKLOG (TASKS)					
Product Backlog	Not Started	In Progress	In Tests	Rejected	Accepted
■	■ ■	■	■		
■	■ ■	■ ■			
■	■	■ ■		■	
■	■ ■				■ ■
■	■				

<http://blog.valkir.net/2011/02/24/core-ingredients-of-scrum/>

104

Artefacts : le sprint Backlog



- Recueil des différentes tâches, extraites du Product Backlog, que l'équipe s'engage à réaliser lors du Sprint.
 - Le travail n'est jamais assigné par un autre
- L'estimation du reste à faire est ajustée chaque jour
- Si une tâche n'est pas claire, ou trop volumineuse, la décomposer en tâches plus petites.

105

Monitor the task board

Story	To Do	Tests Ready	In Process	To Verify	Hours
A user can ... 5	Code the ... 8	X	Code the ... SC 6	Code the ... LC 2	31
	Code the ... 5		Code the ... DC 4		
	Test the ... 6				
A user can ... 2	Code the ... 8				13
	Code the ... 5				
A user can ... 3	Code the ... 3	X	Code the ... MC 4		13
	Code the ... 6				

106

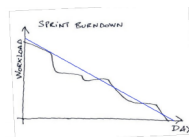
Sprint Backlog (autre version)



<http://blogs.microsoft.co.il/blogs/kolbis/archive/2009/06/24/what-gives-scrum-at-nana1.0.aspx>

107

Artefacts : le Burndown Chart



- Graphique permettant de voir le reste à faire sur un Sprint
 - En abscisse : le nombre de jours du Sprint
 - En ordonnée : la quantité de travail à réaliser
- La ligne droite (en bleu) représente la "Vélocité" idéale de l'équipe.
- La ligne courbe (en noir) représente la "Vélocité" véritable de l'équipe.
- Après chaque Daily Scrum Meeting, en fonction des travaux de la veille de chacun, le Burndown Chart est mis à jour

108

Un pbme est détecté au 4^e jour; la charge est réévaluée

Accélération

104 points scrum

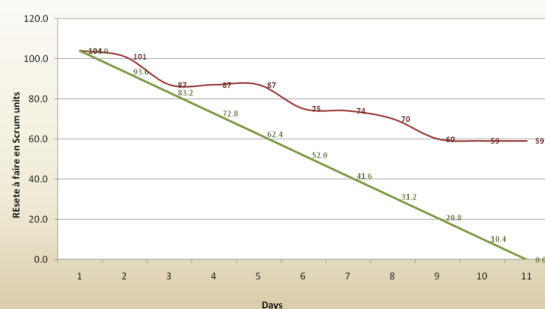


Durée du sprint fixée à 10 jours

<http://www.frenchsug.org/display/FRSUG/Burndown+charts>

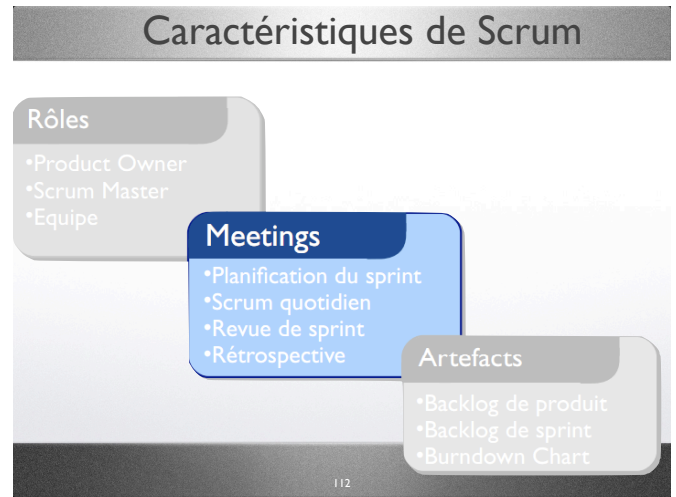
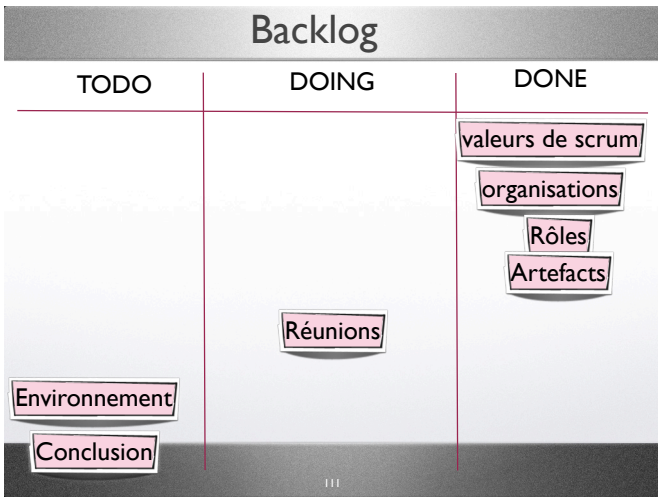
109

Sprint burndown chart - sans avancement



<http://www.frenchsug.org/display/FRSUG/Burndown+charts>

110



Meetings : Planification du sprint

L'équipe choisit, à partir du backlog de produits, les éléments qu'elle s'engage à finir.

- Le backlog de sprint est créé.
 - Les tâches sont identifiées et estimées (1-16 heures)
 - Collectivement, pas seulement par le ScrumMaster
- La conception de haut niveau est abordée

En tant que *touriste potentiel dans la région*, je *veux voir les photos des hôtels*

- Coder la couche de persistance (8 heures)
- Coder l'IHM (4)
- Ecrire les test fixtures (4)
- Coder la classe foo (6)
- Maj les test de performance (4)

114

Sprint Planning 2 : une approche

- **Sprint Planning 2 est ici limité à 2 hours (6 histoires) ...**
- **Agenda for Sprint Planning 2**
 - 14.00 - 14.05 Formation des paires et distribution des histoires
 - 14.05 - 14.35 **Concept** - Réflexion sur les aspects techniques -> Production de documents à présenter à l'équipe
 - 14.35 - 15.05 **Présentation** des paires limitée à 5 Minutes par histoire (Q/R comprises)
 - 15.05 - 15.35 **Tâches** - Chaque paire découpe les histoires en ensemble de tâches d'au plus un jour.
 - 15.35 - 16.00 Présentation des tâches en 4 Minutes maximum (25 Minutes / 6 Stories).

<http://www.scrum-breakfast.com/2011/02/how-we-do-sprint-planning-2.html>

115

Sprint Planning 2 : une approche

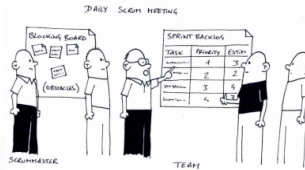
- Au moins 2 personnes ont travaillé chaque tâche
- La réunion est structurée.
- Les solutions sont présentées et discutées avec l'ensemble de l'équipe qui peut alors s'entraider.

<http://www.scrum-breakfast.com/2011/02/how-we-do-sprint-planning-2.html>

116

Meetings : Scrum quotidien

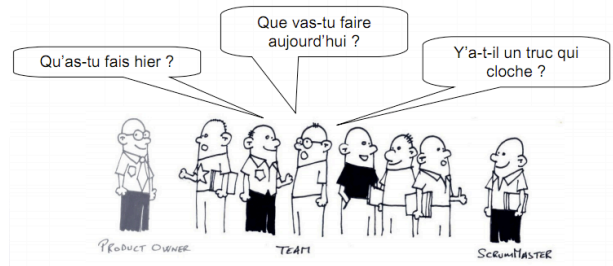
- Tous les jours
- 15 minutes (time boxed)
- Debout



- Pas fait pour résoudre les problèmes
 - Tout le monde est invité
 - Seuls les membres de l'équipe peuvent parler
- Permet d'éviter l'organisation d'autres réunions

117

Meetings : Scrum quotidien

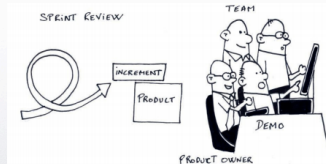


Il ne s'agit pas de compte-rendus au Scrum Master.
=> Ce sont des engagements devant les pairs

118

Meetings : Revue de Sprint

- L'équipe présente ce qu'elle a fait pendant le Sprint.
- L'équipe effectue une démo des nouvelles fonctionnalités incluses dans le livrable de ce Sprint.
- La revue de Sprint est "Informel".
 - Le temps de préparation doit être minimisé.
 - Pas de slides.
- Toute l'équipe participe.
- Tout le monde est invité.



119

Mais que signifie TERMINE ? DONE?

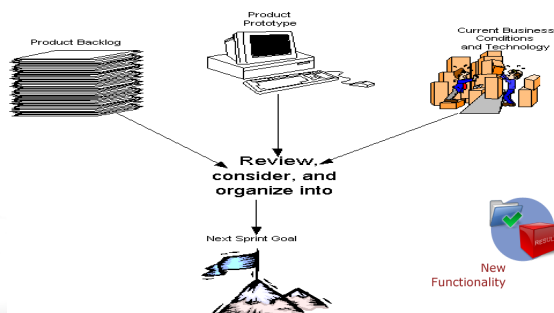
- Il est interdit de livrer un item inachevé, même avec l'intention de « le terminer plus tard ».
 - ➔ Maintenir la confiance avec le client de ne pas « cacher » le travail non « terminé ».
- Functionality has been code reviewed, functionality has been integrated and built, acceptance tests have been run, and documentation has been created.
- Code adheres to standards, is clean, has been refactored, has been unit tested, has been checked in, has been built, and has had a suite of unit tests applied to it

120



Danube Technologies

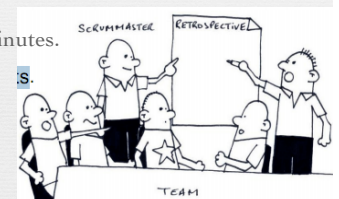
Sprint Review Meeting



121

Meetings : Rétrospective

- A la fin de chaque sprint
- Permet de réfléchir régulièrement à ce qui marche et ce qui ne marche pas.
- Dure en général de 15 à 30 minutes.
- Fait à la fin de chaque Sprint.
- Toute l'équipe participe.
 - Scrum Master
 - Product Owner
 - Equipe
 - Eventuellement d'autres intervenants

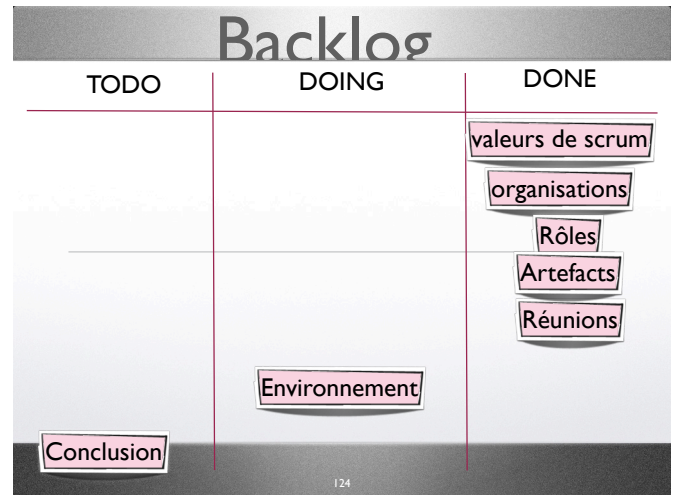


122

Evaluation Consequences

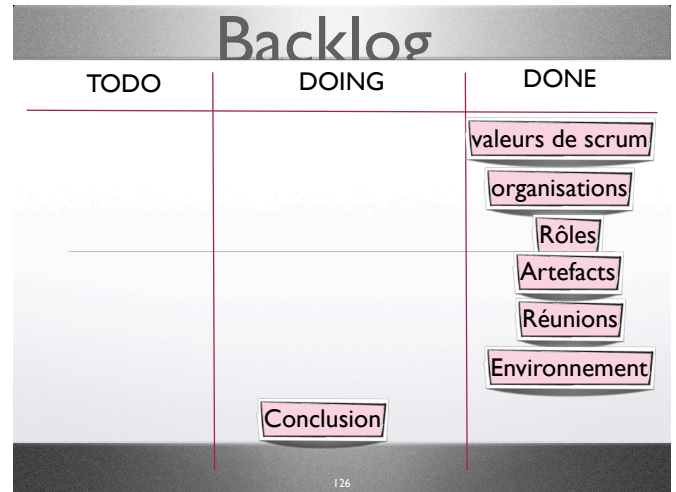
1. Restoring unfinished functionality to the Product Backlog and prioritizing it.
2. Removing functionality from the Product Backlog that the team unexpectedly completed.
3. Working with the ScrumMaster to reformulate the team.
4. Reprioritizing the Product Backlog to take advantage of opportunities that the demonstrated functionality presents.
5. Ask for a release Sprint to implement the demonstrated functionality, alone or with increments from previous Sprints.
6. Choosing not to proceed further with the project and not authorizing another Sprint.
7. Requesting that the project progress be sped up by authorizing additional teams to work on the Product Backlog.

123



Environnement de collaboration

Organisation en « war room »



Conclusion

- Méthode de gestion de projet – développement logiciel
- A compléter avec des techniques d'ingénierie logicielle (XP est un support intéressant)
- Conditions propices nécessaires
- Expérimentations prometteuses
- Principal bénéfice : des équipes motivées

127

Perspectives

- Défauts à palier
 - Absence de dépendance entre les tâches
 - Polyvalence des programmeurs
 - Productivité équivalente supposée
 ⇒ Grande **maturité** nécessaire
- Contrats à adapter
- Stratégie d'introduction de Scrum en entreprise

128

Agilité et Systèmes d'information

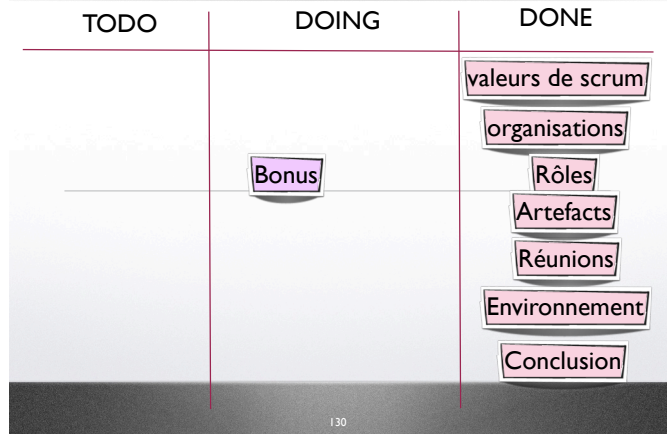


19/9/11 : Agilité des SI, la révolution de l'adaptabilité permanente

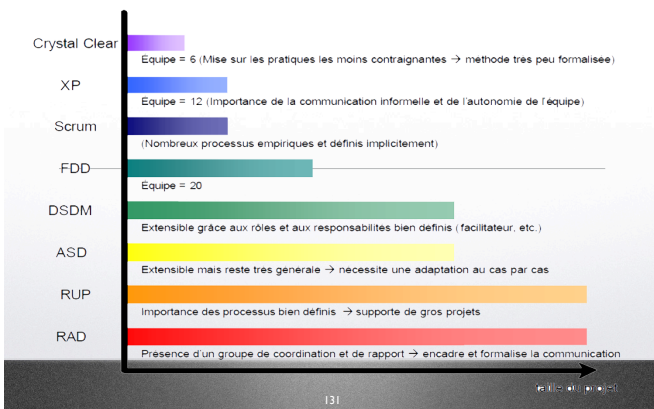
- > **Face aux évolutions permanentes de leur business, les entreprises misent sur l'agilité.**
- > L'adoption des méthodes agiles itératives reste plus que jamais d'actualité. En s'inspirant de Scrum, XP ou Unified Process, on accélère la délivrance de chaque projet en accord avec les priorités des métiers. Parallèlement, une démarche de Lean Management implique plus fortement les développeurs vis-à-vis de la qualité au meilleur coût.

Article à lire (beaucoup d'autres informations)

Backlog



Comparaison des méthodes agiles



Bibliographie

- UNIVERSITE PARIS XII - ISIAG , MASTER 2, METHODOLOGIE ET CONDUITE DE PROJETS, CHAPITRE 3
- Génie Logiciel Orienté Objets, Philippe Collet, Master 1 Informatique, 2007-2008
- Processus de conception de SI MI MIAGE - SIMA - 2005-2006 Yannick Prié UFR Informatique - Université Claude Bernard Lyon 1
- Méthodes de conduite de projet, Tester, optimiser, structurer ses applications Jean David Olekhovitch, jd@olek.fr - www.olek.fr
- Les cours IBM sur le RUP
- Conduite de projet, Méthode d'analyse et de conception, Processus unifié, G. Picard, SMA/G2I/ENS Mines Saint-Etienne gauthier.picard@emse.fr, Octobre 2009
- Processus Unifié : www2.lifl.fr/~clerbout/.../Cours4-ProcessusUnifie.pdf
- eXtreme Programming & Scrum Practices, Embrace Change, Naresh Jain
- La Gestion de projets (informatiques) et la question de la collaboration; G. Beauvallet, Octobre 2007, telecom paris.

Bibliographie

- Exigences et spécifications du logiciel -- © RR - LOG3410 – XPproc, Université de Montreal, 2011
- Dave Elliman, XP, Extreme programming
- Céline ROUDET, Module ITR3 – Génie Logiciel
- Raphaël Marvie, Lean Software Development
- eXtreme Programming (XP) CSE30, University of Sunderland Harry R. Erwin, PhD
- eXtreme Programming & Scrum Practices, Embrace Change, Naresh Jain
- Méthode AGILE Les meilleures pratiques Compréhension et mise en oeuvre, Jean-Pierre Vickoff (Auteur), AgileAlliance

