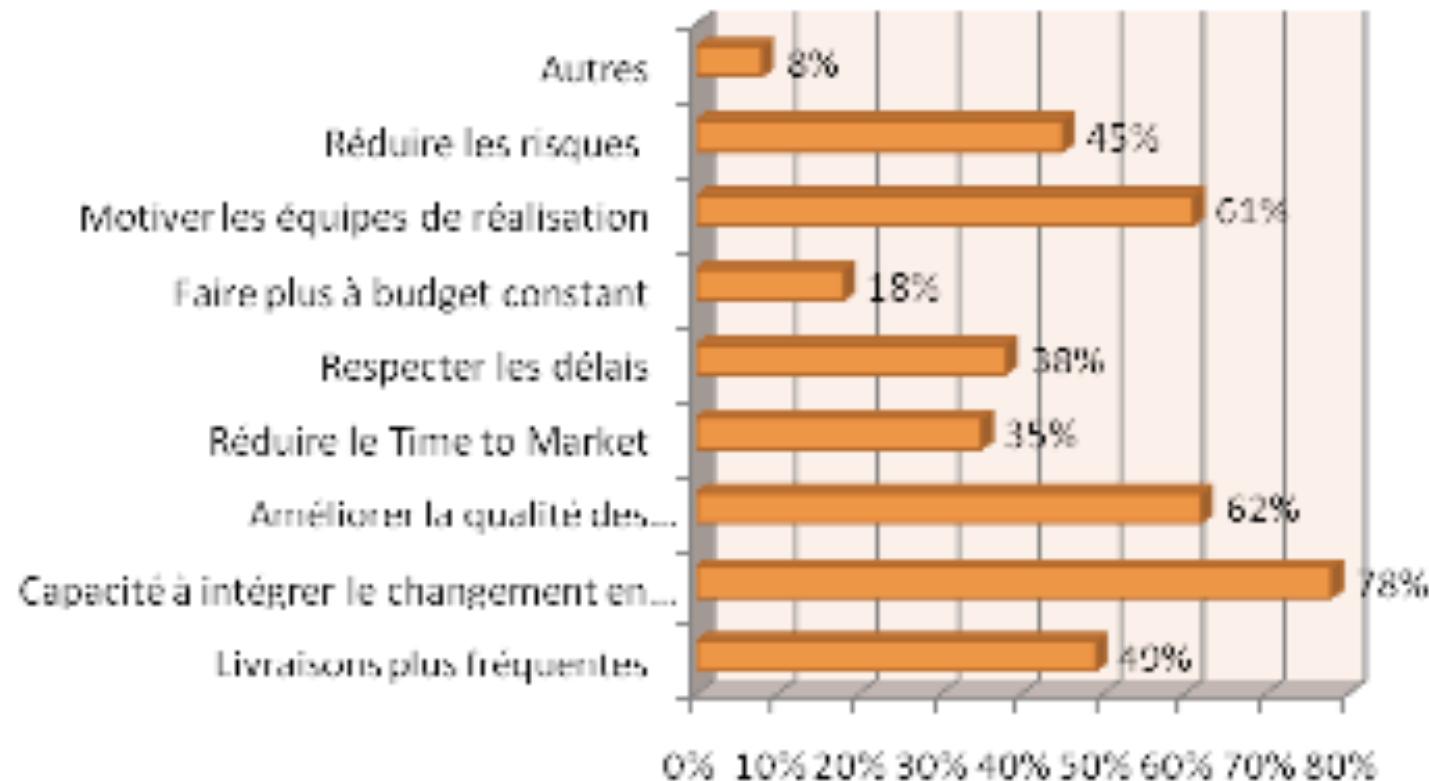


# SCRUM : le détail

# Les fondateurs

- Jeff Sutherland
  - Débute Scrum en 1993
  - Environ 600 personnes chez IDX
- Ken Schwaber
  - Initie les définitions de Scrum avec Jeff en 1996
- Mike Beedle
  - Schémas SCRUM

# Motivations



# Lancement d'un projet Scrum

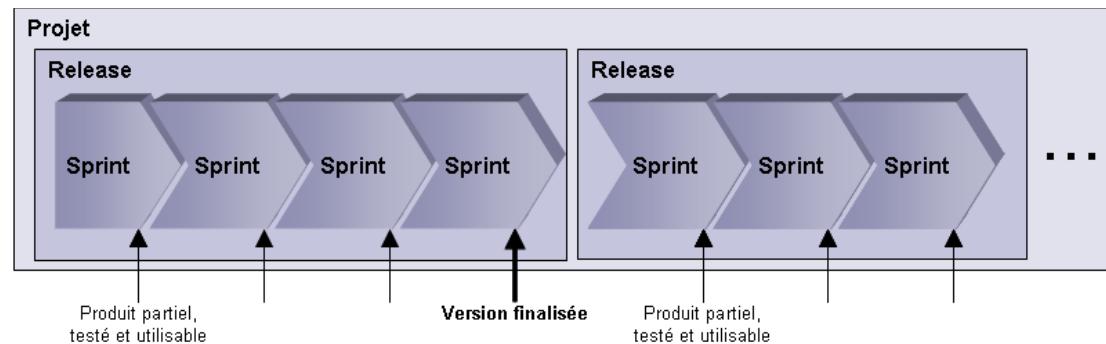
- Créer un product backlog avec une équipe et un product owner
- Faire une revue du backlog pour les risques et les questions complexes
- Définir avec le management la roadmap (Priorité, Dates, etc.)
- Assigner les rôles dans l'équipe
- Lancer le projet
- Valider les objectifs des sprints
- Daily meeting : résoudre les obstacles
- Introspection et modification / adaptation

# Le product backlog

- Liste des fonctionnalités désirées
  - En général il est constitué de
  - User stories (fonctionnel)
  - Technical stories (amélioration technique,etc.)
- Cette liste est priorisée par le product owner (en respectant la valeur ajoutée)

# Scrum – Planifier un projet

Backlog produit - Site marchand XY					
ID_Item	Titre	Importance	Estimation	Démonstration de la fonctionnalité	Commentaires
SPRINT 1					
1	Besoin 1	130	12	xxxxxxxx	xxxxxxxx
2	Besoin 2	120	9	xxxxxxxx	xxxxxxxx
3	Besoin 3	115	20	xxxxxxxx	xxxxxxxx
SPRINT 2					
4	Besoin 4	110	8	xxxxxxxx	xxxxxxxx
5	Besoin 5	100	20	xxxxxxxx	xxxxxxxx
6	Besoin 6	95	12	xxxxxxxx	xxxxxxxx
SPRINT 3					
7	Besoin 7	80	10	xxxxxxxx	xxxxxxxx
8	Besoin 8	70	8	xxxxxxxx	xxxxxxxx
9	Besoin 9	60	10	xxxxxxxx	xxxxxxxx
10	Besoin 10	40	14	xxxxxxxx	xxxxxxxx
SPRINT 4					
11	Besoin 11	35	4	xxxxxxxx	xxxxxxxx
12	Besoin 12	25	6	xxxxxxxx	xxxxxxxx
13	Besoin 13	10	7	xxxxxxxx	xxxxxxxx
14	Besoin 14	10	11	xxxxxxxx	xxxxxxxx
15	Besoin 15	10	3	xxxxxxxx	xxxxxxxx



Source : <http://fr.wikipedia.org>

- Constitution du **backlog produit** par le product owner.
- Répartition en **sprints** et en **releases**.

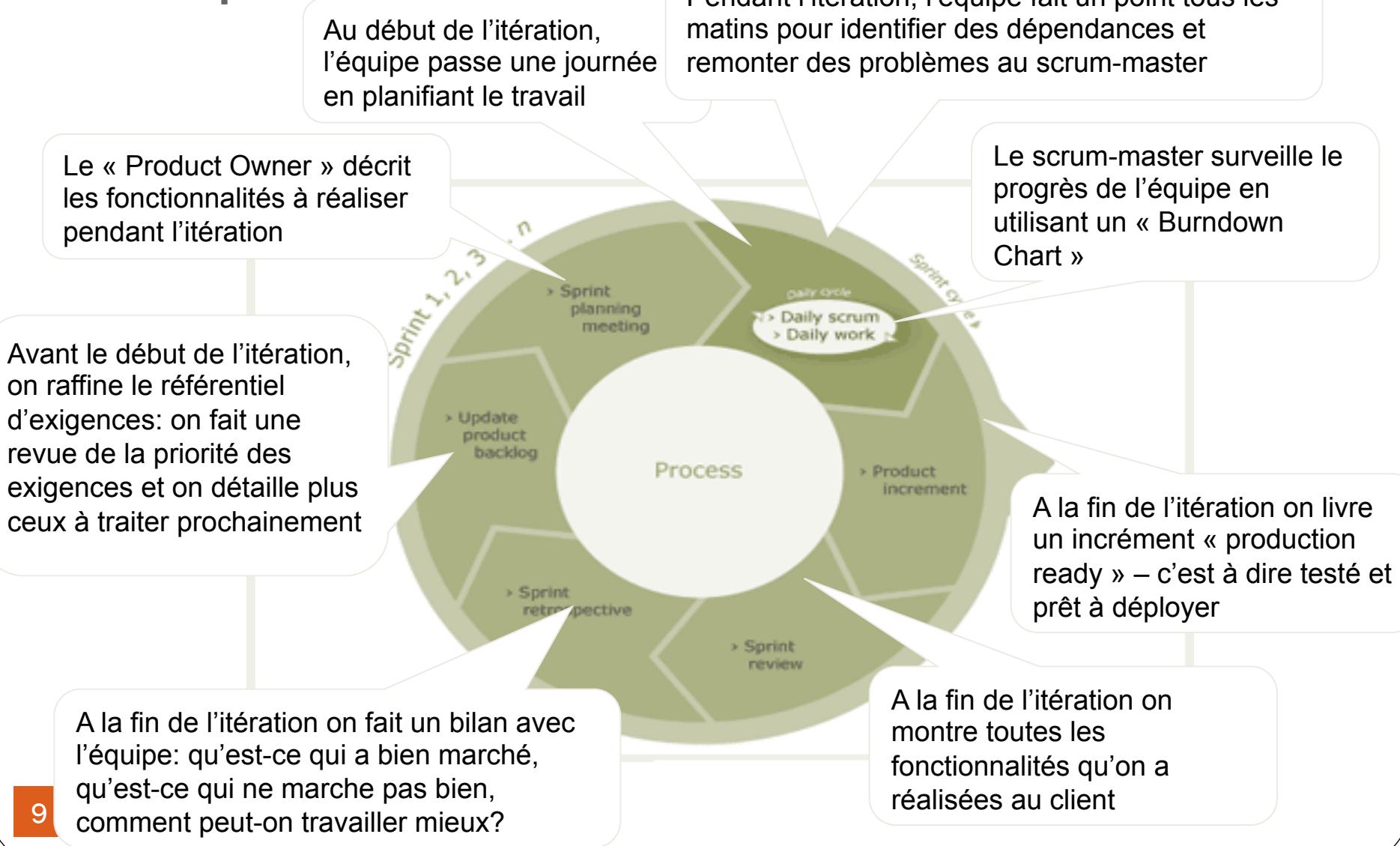
# SCRUM : User stories dans un backlog

- **ID - un identifiant unique, juste un nombre auto-incrémenté.**
- **Nom - Un nom, une description courte de l'histoire.**
  - Par exemple « Voir l'historique de ses transactions ».
  - Normalement 2 à 10 mots.
- **Importance** - Par exemple 10. Ou 150. Élevée = plus important.
- **Estimation initiale :** = combien de jours \* combien de ressources = Points histoires

# Exemple de Backlog

BACKLOG DE PRODUIT (exemple)					
ID	Nom	Imp	Est	Démo.	Notes
1	Dépôt	30	5	Authentification, ouvrir la page de dépôt, déposer 10€, aller sur la page du solde et vérifier que ça a bien augmenté de 10€.	Nécessite un diagramme de séquences UML. Ne pas se soucier du cryptage pour l'instant.
2	Voir l'historique de ses	10	8	Authentification, cliquez sur « transactions ».	Utiliser la pagination pour éviter

# Un sprint : déroulement



# Planning Sprint

- Faites en sorte que le backlog de produit soit en ordre *avant cette* réunion de planification du sprint.
- Il faut définir :
  - Un but pour le sprint.
  - Une liste des membres d'équipe (et de leur niveau d'engagement, si ce n'est pas 100%).
  - Un backlog de sprint (= une liste des histoires incluses dans le sprint).
  - Une date bien définie pour la démonstration.
  - Une heure et un lieu bien définis pour la mêlée quotidienne

# Exemple de réunion de planning sprint

- **13:00 – 17:00 (10 minutes de pause toutes les heures)**
  - **13:00 – 13:30.** Le directeur de produit décrit le **but du sprint** et résume le backlog de produit. Le lieu, la date et l'heure de la démonstration sont fixés.
  - **13:30 – 15:00.** L'équipe fait les estimations en temps, et décompose les éléments selon les besoins. Le directeur de produit met à jour les niveaux d'importance selon les besoins. Les éléments sont clarifiés. « Comment démontrer » est explicité pour chacun des éléments les plus importants.
  - **15:00 – 16:00.** L'équipe sélectionne les histoires à inclure dans le sprint. Elle fait les calculs de vitesse pour se confronter à la réalité
  - **16:00 – 17:00.** On choisit le lieu et l'heure pour la mêlée quotidienne. On poursuit la décomposition des histoires en tâches

# But du sprint

- Pourquoi faisons-nous ce sprint ?
- Exemple :
  - Rendre le système suffisamment bon pour être déployé chez un groupe de beta-testeurs
  - Ajouter un support basique pour le back office

## Le choix des histoires à inclure dans le sprint

- 1. Comment l'équipe décide-t-elle quelles histoires inclure dans le sprint ?
- 2. Comment le directeur de produit peut-il influencer leur décision ?
- Estimation en fonction de la vélocité :
  - 1. Décider la *vélocité estimée*
  - 2. Calculer combien d'histoires vous pouvez ajouter sans dépasser la vélocité estimée
- La vélocité mesure la «quantité de travail fini», où chaque élément est pondéré selon son estimation initiale.

# Exemple de vitesse estimée

Début du sprint

8
5
5
3
5

Vélocité estimée = 26

Fin du sprint

Terminé !	8
Terminé !	5
Terminé !	5
Presque terminé	3
Pas commencé	5

Vélocité réelle = 18

Vélocité estimée , météo de la veille : on utilise le sprint précédent

VELOCITE ESTIMEE CE SPRINT :

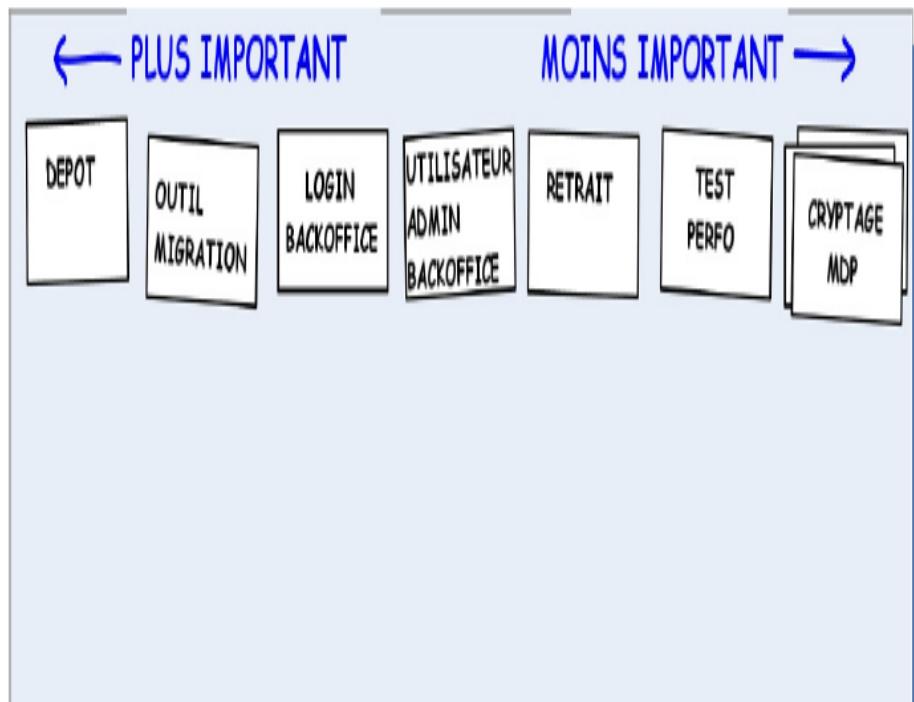
$$(JOURS-HOMME DISPONIBLES) \times (\text{FACTEUR FOCALISATION}) = (\text{VELOCITE ESTIMEE})$$

$$(\text{FACTEUR FOCALISATION}) = \frac{(\text{VELOCITE REELLE})}{(\text{JOURS-HOMME DISPONIBLES})}$$

# Vélocité estimée : exemple

- Supposons que 3 ressources sont disponibles 50 jours/hommes
- Supposons que le dernier sprint on a fait 18 points d'histoire et donc le facteur de focalisation est :  $18/45 = 40\%$
- Supposons qu'il y a une nouvelle ressource pour le nouveau sprint et que l'équipe ainsi construite est disponible pour 50 J/H, la vélocité estimée est alors  $40*50=20$  points
- Recommandation pour une équipe nouvelle en SCRUM = **70%**

# Utiliser des fiches



Elément Backlog #55

## Dépôt

Notes

Besoin d'un diagramme de séquence UML.  
Nul besoin de se soucier du chiffrement pour l'instant.

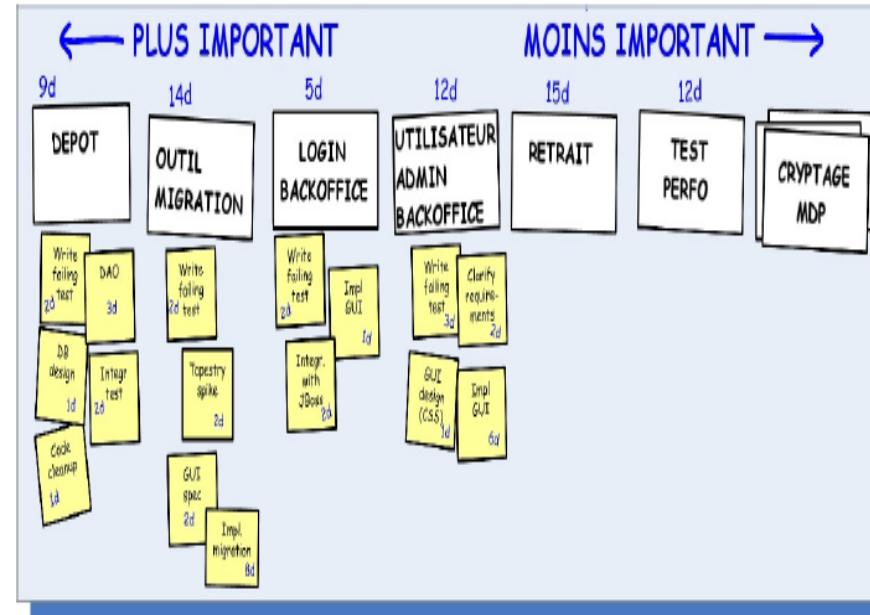
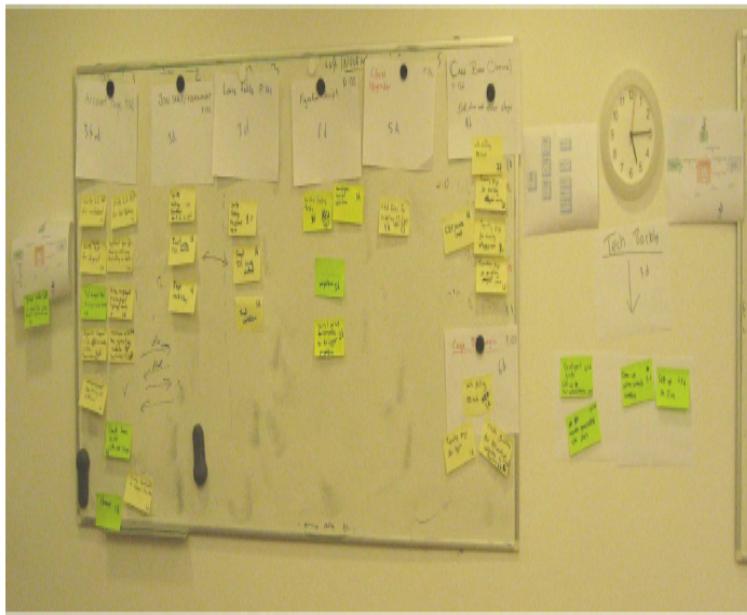
Comment démontrer

S'authentifier, ouvrir la page des dépôts, déposer 10 € et vérifier sur la page du solde qu'il a augmenté de 10 €.

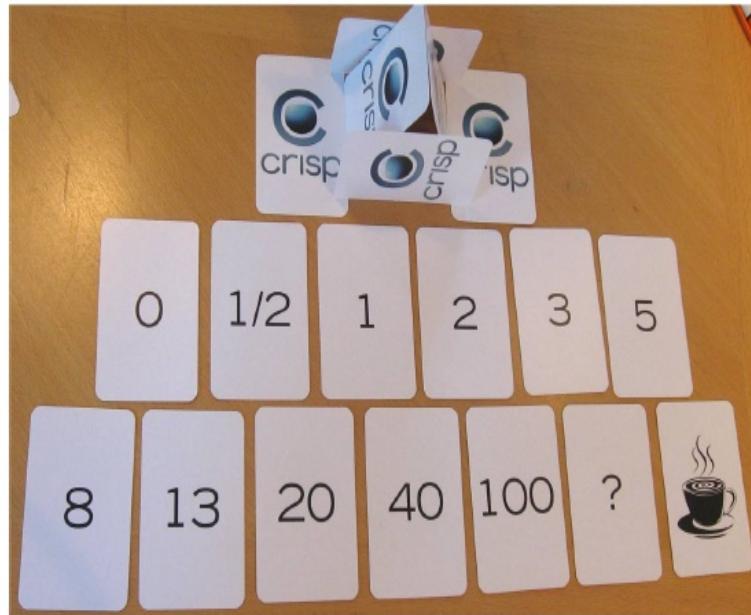
Importance  
30

Estimation

# Sprint planning



# Estimée avec le Pocker planning



- Chaque membre de l'équipe reçoit un jeu de 13 cartes
- Chaque membre de l'équipe sélectionne une carte qui représente son estimation de temps (en points d'histoire) et la place sur la table face cachée.
- Quand tous les membres de l'équipe ont terminé, toutes les cartes sur la table sont révélées simultanément
  - 0 = « cette histoire est déjà faite ».
  - ? = « Je n'ai aucune idée. Vraiment aucune. »
  - Tasse de café = « Je suis trop fatigué pour penser. Faisons une courte pause. »

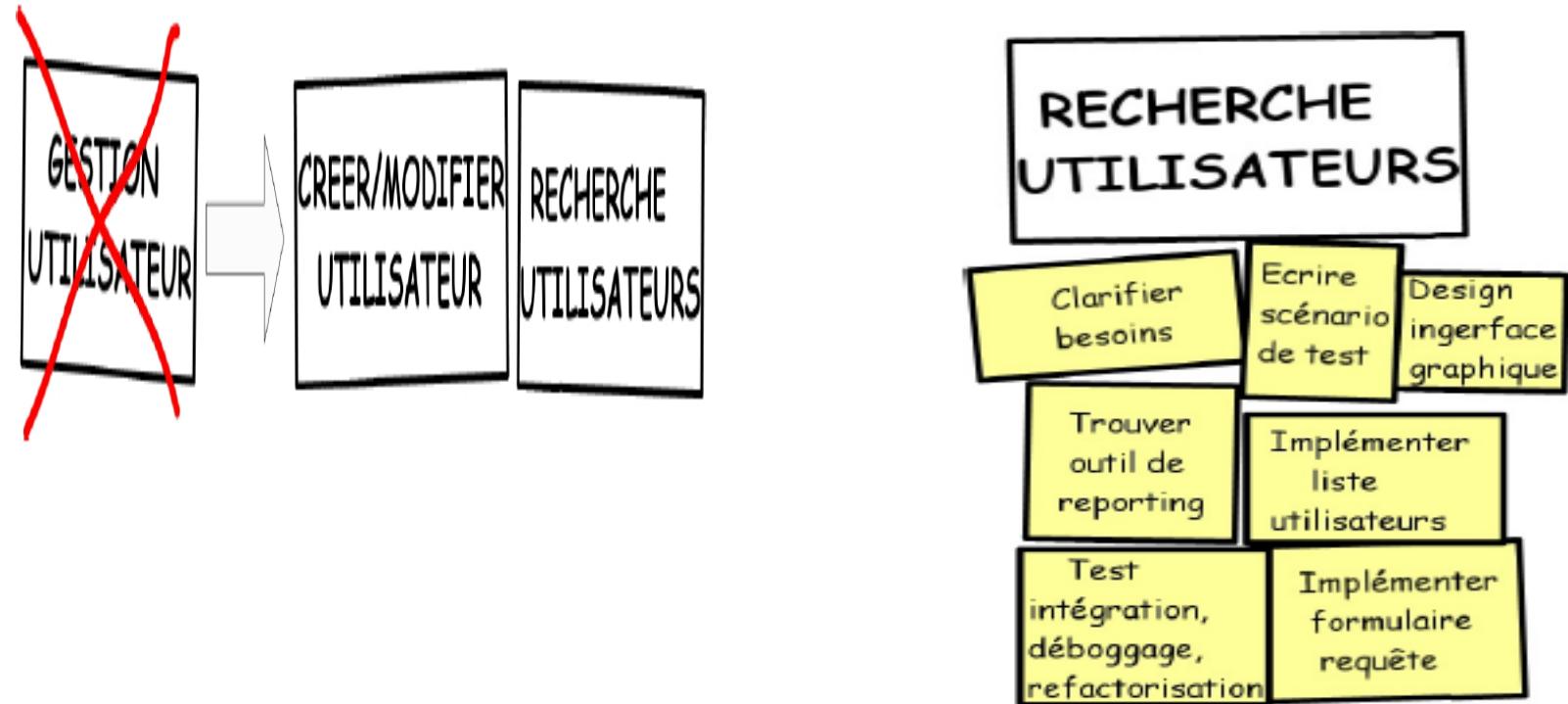
# Quelques chiffres comme exemple

- Les histoires doivent ni être trop petite ni trop grosse
- Moyenne d'une histoire : 2 à 8 jours/hommes
- Vélocité entre 40 et 60
- 10 histoires par sprint

# Histoire # Tâche

- Les histoires sont des choses livrables qui intéressent le directeur de produit.
- Les tâches sont des choses non livrables, ou des choses auxquelles le directeur de produit ne s'intéresse pas
- Exemple :

# Exemple d'histoire et tâches



# Comment communiquer sur les sprints

## Equipe Jackass, sprint 15

### **Objectif du sprint**

- Release prête pour la beta !

### **Backlog du sprint** (estimations entre parenthèses)

- Dépôt (3)
- Outil de migration (8)
- Login backoffice (5)
- Administrateur backoffice (5)

Vélocité estimée : 21

### **Planning**

- Période de sprint : 06/11/2006 au 24/11/2006
- Scrum quotidien : 9h30 - 9h45, salle de l'équipe
- Démo du sprint : 24/11/2006 à 13h, cafétéria

## Objet : sprint 15 de Jackass a commencé

Salut à tous ! L'équipe Jackass a commencé le sprint 15. Notre but est de démontrer une release prête pour la beta le 24 novembre.

Les détails sur la page d'info du sprint :

<http://wiki.mycompany.com/jackass/sprint15>

# Comment communiquer sur les sprints

## Tableau de bord

### Sprints en cours

- Equipe X sprint 15
- Equipe Y sprint 12
- Equipe Z sprint 1

Objet : démo sprint Jackass demain 13h à la cafétéria

Salut à tous ! Vous êtes les bienvenus pour assister à la démo de sprint à 13h à la cafétéria demain (vendredi). Nous démontrerons une release prête pour la beta.

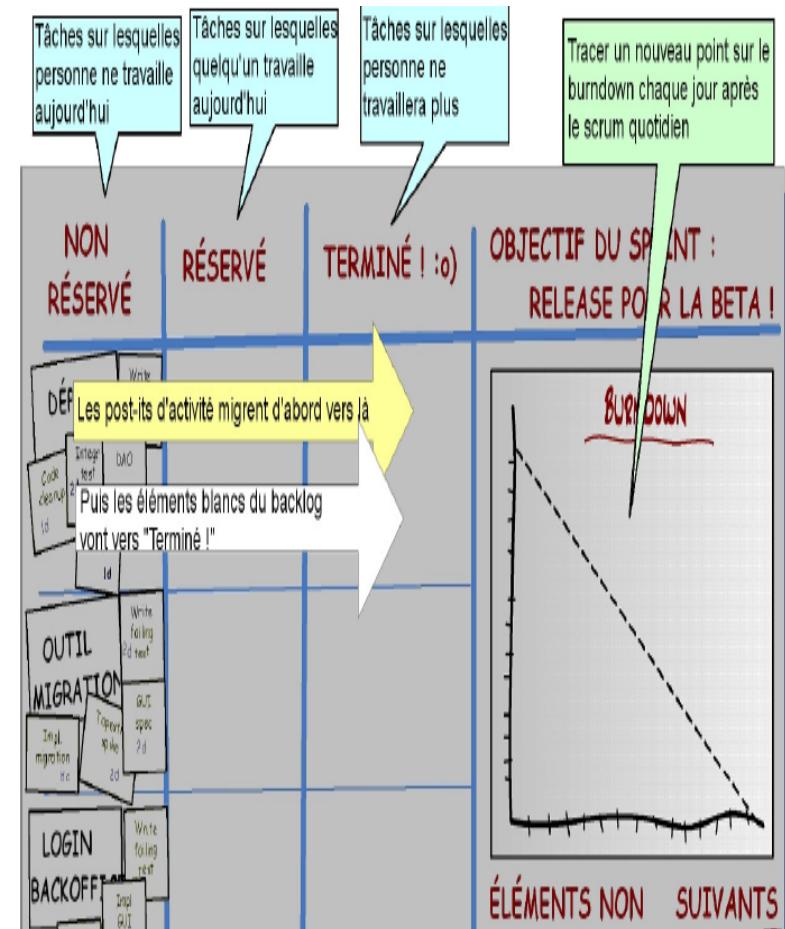
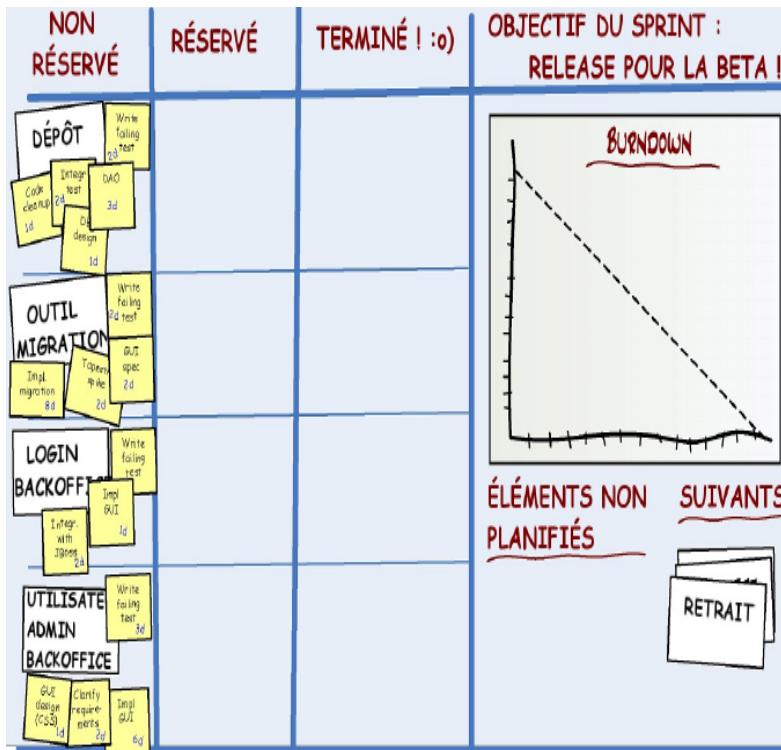
Les détails sur la page d'info du sprint :

<http://wiki.mycompany.com/jackass/sprint15>

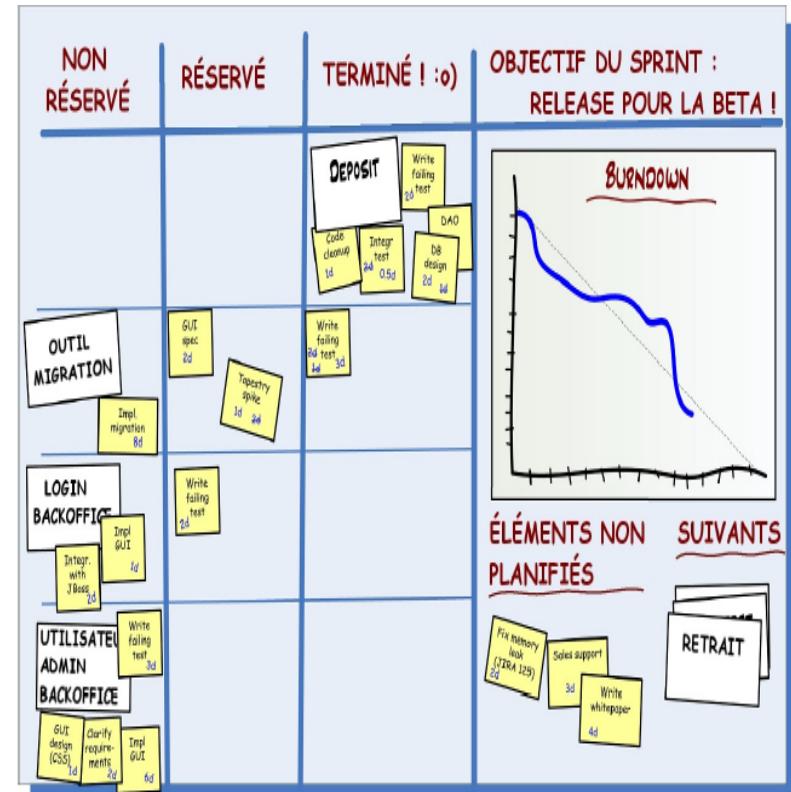
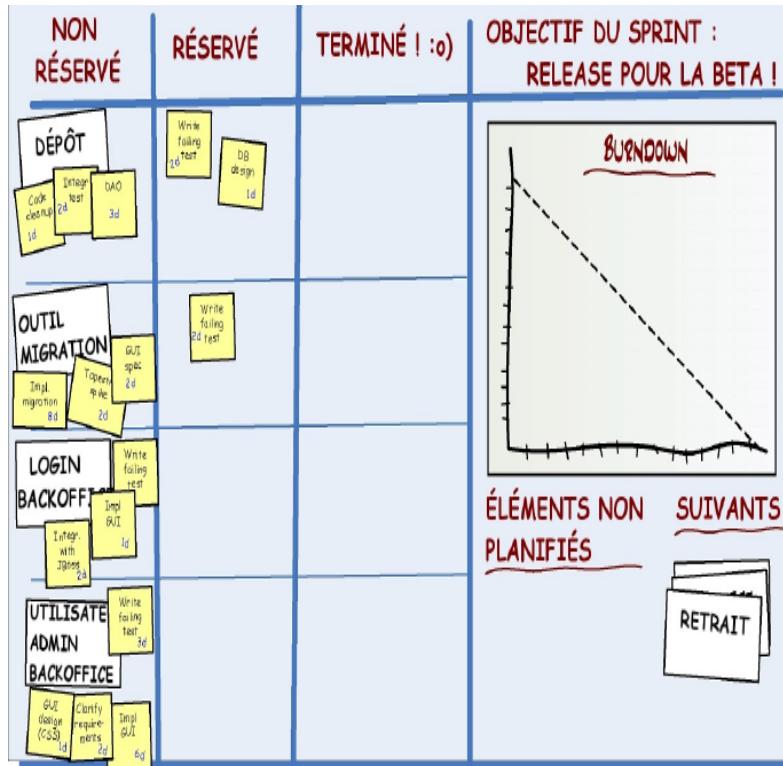
# Backlogs de sprint

- On peut utiliser soit :
- Excel
- Tableau physique de tâches sur le mur( 2x2 mètres)
- Outils de gestion de bugs comme Jira

# Tableau de tâches : fonctionnement



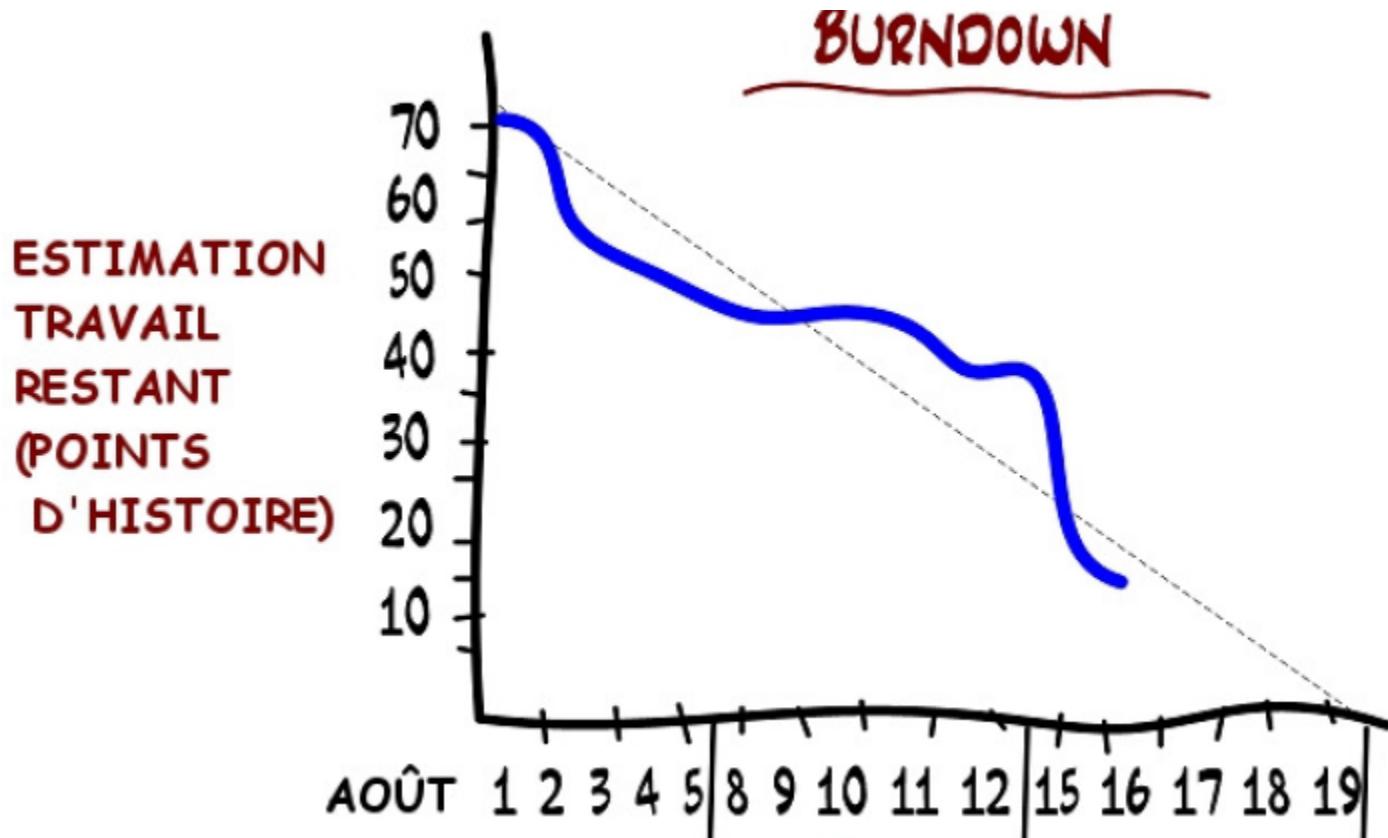
# Tableau de tâches : exemple



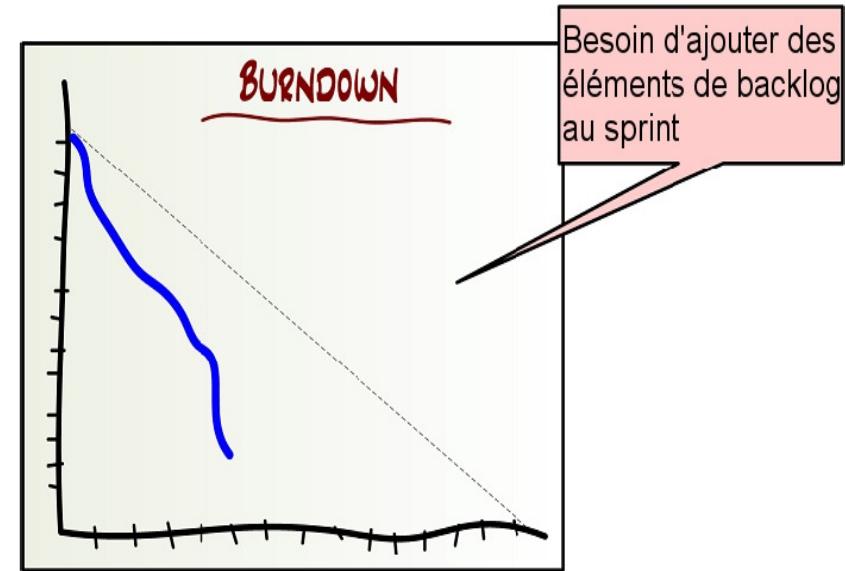
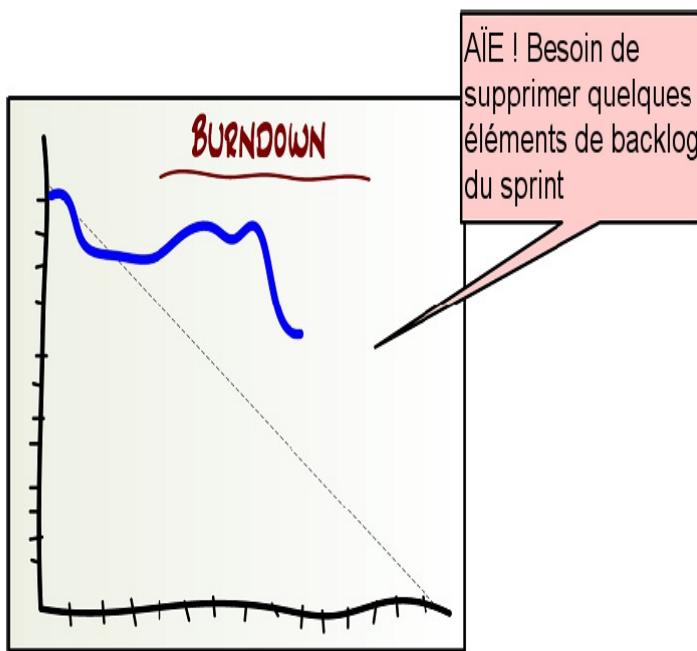
# Exemple de vrai planning sprint



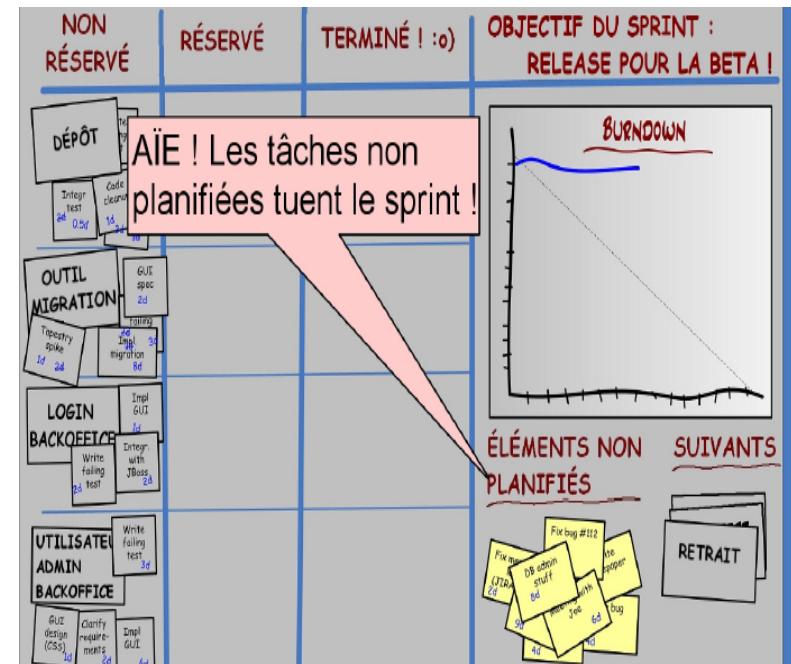
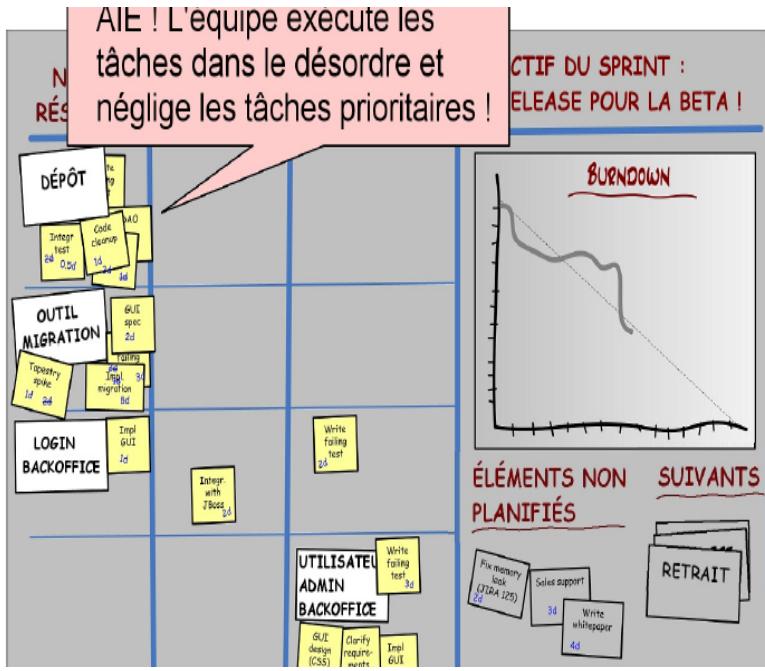
# Comment le burndown chart fonctionne



# Comment le burndown chart fonctionne

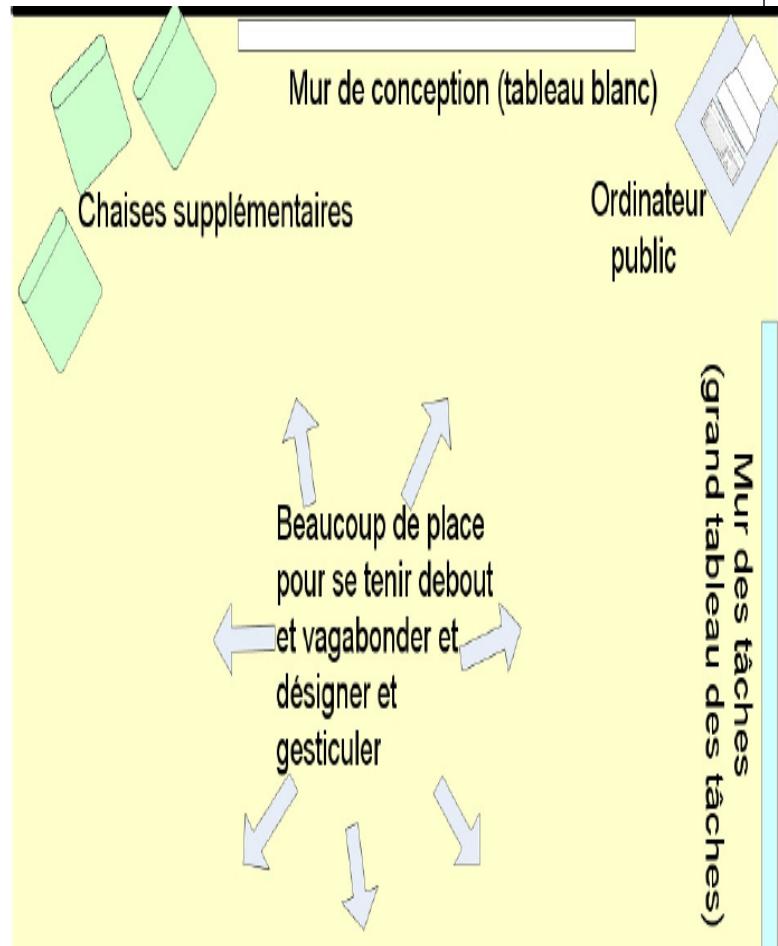


# Comment le burndown chart fonctionne



# Estimations et coin conception

- 1 jour-homme effectif = 6 heures homme effectives.
- La plus petite valeur est de 0,5 jour



# Mise à Jour du Taskboard

- Pendant ou avant le daily scrum :
- Responsable : Scrum Master ou l'équipe

# Restrospective de sprint

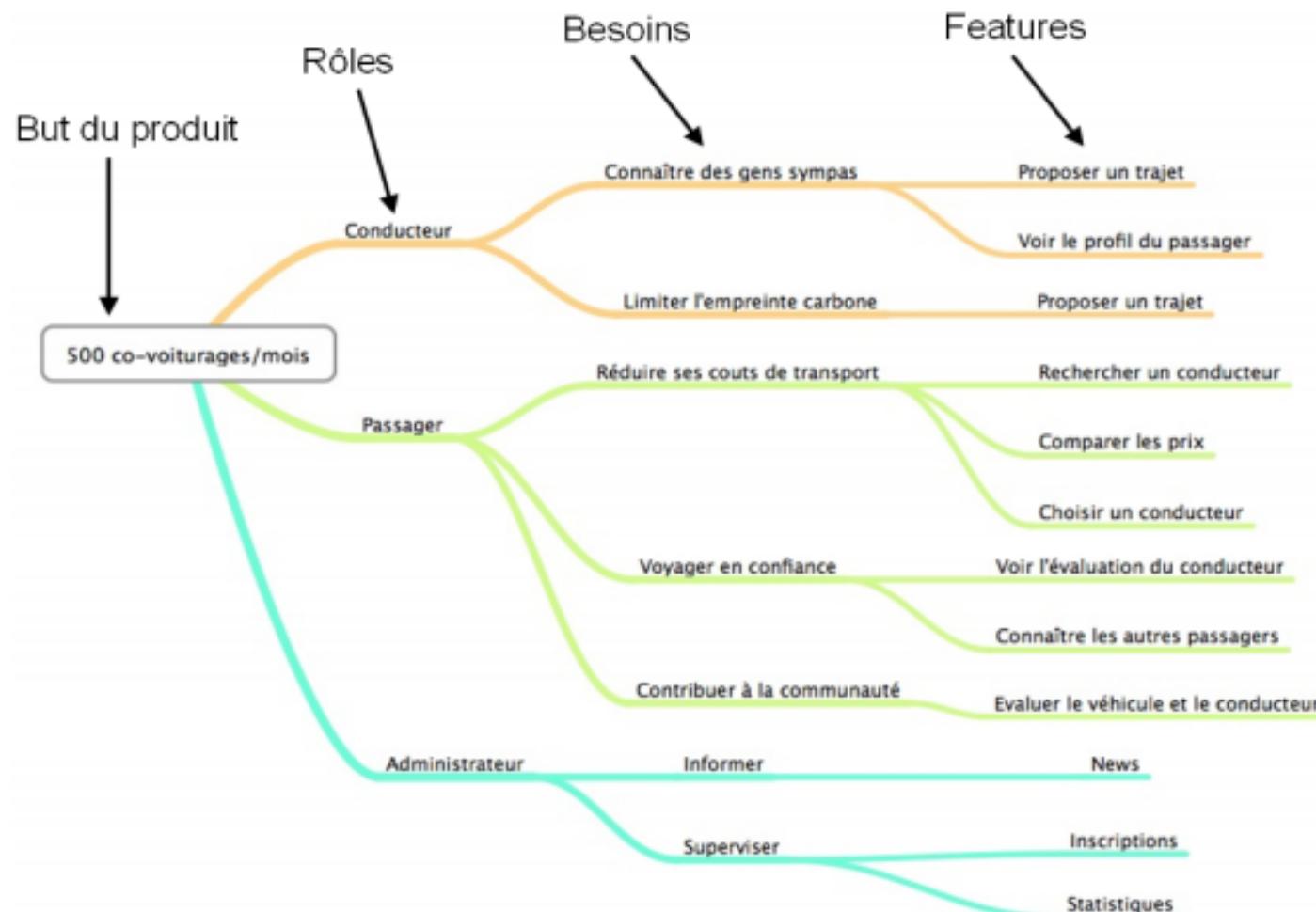


# Restrospective de sprint

- Trois colonnes :
  - **Bien** : *Si nous pouvions refaire le même sprint, nous ferions ces choses exactement pareil.*
  - **Peut mieux faire** : *Si nous pouvions refaire le même sprint, nous ferions ces choses différemment.*
  - **Améliorations** : *Idées concrètes pour s'améliorer dans le futur*

# Compléments sur SCRUM

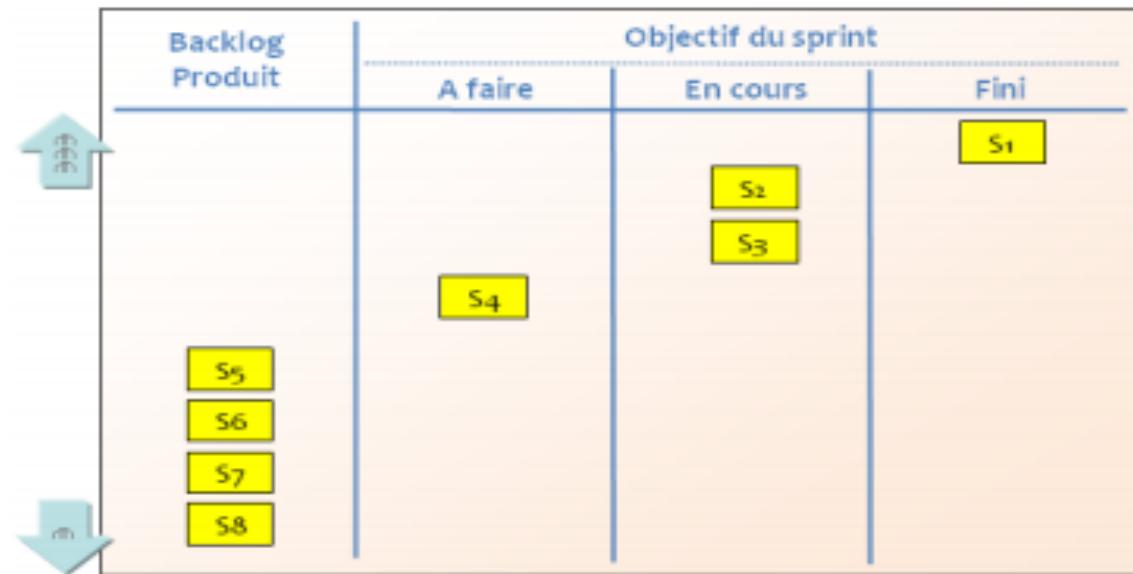
# Vision du produit : 4 niveaux



Vision du produit « Site de co-voiturage »

# ScrumBoard

1. Les items de votre Backlog à terminer pour le dimanche après-midi
2. Les items à faire lors du sprint à venir → A faire
3. Les items en cours → En cours
4. Les items finis → Fini



# Vie d'un Post-iT

L'initialisation d'un Post-It (un item du Backlog Produit).

Son point de départ c'est donc la 1<sup>ère</sup> colonne de gauche. Pour le moment, il contient ces informations :

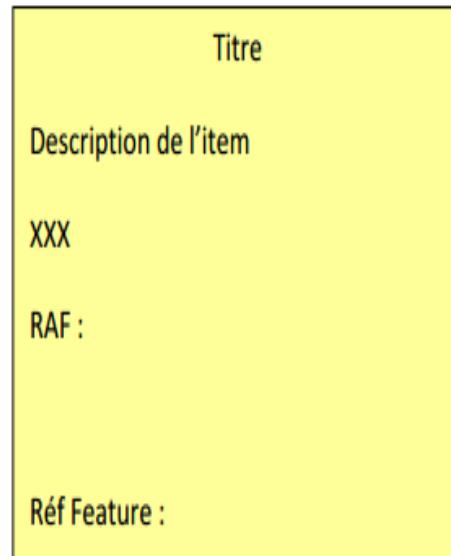


# Vie d'un Post-iT

- Titre: Nom de la fonctionnalité (on parle de User Story).
- Description : on utilise généralement le plan type à 3 rubriques : en tant que <rôle>, je peux <intention fonctionnelle> afin de <valeur ajoutée> Exemple : en tant que Conducteur, je propose une place dans ma voiture à un passager afin d'être accompagné dans mon prochain voyage.
- Réf Feature : Le nom de la feature à laquelle l'élément est associé (back office des utilisateurs,...),

# Vie d'un Post-iT

Il passe enfin dans la colonne A faire quand quelqu'un commence à travailler dessus.



- On ajoute Les initiales du ou des développeurs en charge de la tâche,
- RAF : Le Reste à faire jour par jour permettant de mettre à jour le BurnDown. Le dernier RAF doit être à 0 et permet de passer la tâche dans la colonne "Done".

# La matrice Rôle-Fonctionnalité-Bénéfice

- En tant que <rôle>
- Je veux que <fonctionnalité>
- Afin de <bénéfice>
- Par exemple:
  - En tant que client de la banque
  - Je veux retirer de l'argent à un guichet automatique
  - Afin de ne me soucier ni des horaires d'ouverture ni de l'affluence en agence

# Carton, Conversation, Confirmation

- proposée par Ron Jeffries
  - le *carton* ou Post-It, qui permet de donner une forme tangible et durable à ce qui n'est sans cela qu'une abstraction, à savoir:
  - une *conversation* qui peut avoir lieu en plusieurs temps au fil d'un projet, entre toutes les personnes concernées par un aspect fonctionnel d'un logiciel: clients, utilisateurs, développeurs, testeurs;
  - la *confirmation*, enfin, la plus formalisée possible, que l'objectif dont il a été question au cours de cette conversation est atteint.

# INVEST

- I – Independent : Indépendante des autres
- N – Negotiable : Négociable initialement, plutôt qu'un engagement ferme
- V – Valuable : Verticale, ou ayant de la valeur en soit
- E – Estimable : Evaluée en termes de complexité relative
- S – Small : Suffisamment petite
- T – Testable : Testable en principe, ce qu'on vérifie en écrivant un test

# INVEST : Négociable

- Pour répondre au critère **N**, ne formuler dans un premier temps que l'essentiel, à savoir l'objectif fonctionnel recherché; on évitera par exemple de spécifier dans une User Story des éléments techniques, par exemple "En tant qu'acheteur, lorsque j'écris dans le champ texte puis que je clique sur le bouton Recherche, la liste à gauche du champ de recherche est renseignée avec les articles correspondants". Ces détails d'implémentation feront l'objet d'une discussion permettant d'identifier la meilleure solution; initialement, une formulation du type "L'acheteur peut chercher des articles par mot-clé" est suffisante pour l'estimation et la planification.

# INVEST : Vértical

- Pour répondre au critère **V**, représenter un incrément réellement utile pour l'utilisateur final ou du point de vue du client. Par exemple, "réaliser le schéma de la base de données pour la facturation" n'est pas un incrément ayant de la valeur en soi, mais une tâche technique. A contrario, "émettre une facture pour les achats d'articles en France" en laissant pour plus tard une seconde Story dont l'énoncé serait "émettre une facture pour des achats livrés depuis l'étranger" représente un meilleur découpage: chaque incrément permet de réaliser une partie distincte du chiffre d'affaire

# SMART Tasks

- Specific = défini et explicite
- Measurable = quantifiable, observable
- Achievable = peut exister, peut être fait
- Relevant = approprié à cette user story
- Time bound = quand cela doit-il intervenir

# Given - When - Then

- Format recommandé pour le test fonctionnel d'une User Story:
  - (Given) (Etant donné) un contexte,
  - (When) (Lorsque) l'utilisateur effectue certaines actions,
  - (Then) (Alors) on doit pouvoir constater telles conséquences

# Given - When – Then : exemple

- Par exemple:
  - Etant donné un solde positif de mon compte, et aucun retrait cette semaine,
  - Lorsque je retire un montant inférieur à la limite de retrait,
  - Alors mon retrait doit se dérouler sans erreur ou avertissement
  - Des outils tels que JBehave, RSpec ou Cucumber encouragent l'utilisation de cette formule.