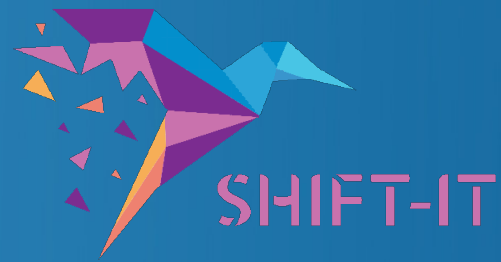
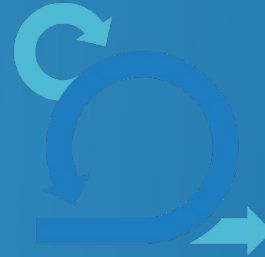


Rémy BROUSSET  
remy.brousset@aquilenet.fr

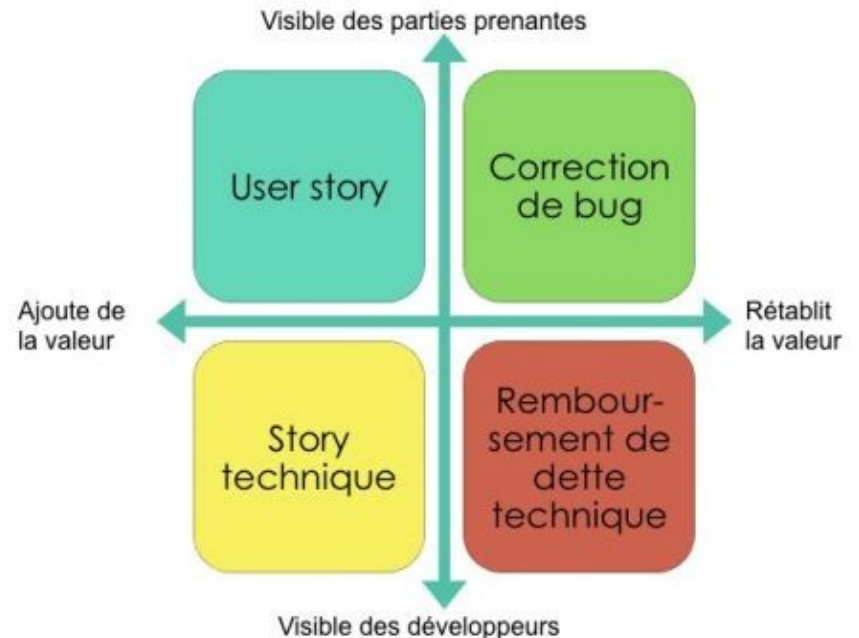


# Introduction à SCRUM - Artefacts



# User Story

- Unité de base de Scrum
- Courte phrase décrivant l'ajout de valeur pour le client
- Rédigée par le Product Owner
- Généralement fonctionnelle et nominale
- Plus rarement (à éviter) :
  - Bug (ou Defect Story)
  - Tache Technique
  - Refactoring



# Les 3 « C »

- **Card** : la phrase est courte :

- En tant que <rôle>
- Je veux <atteindre un but>
- En <effectuant une action>
- (ou : En tant que «rôle», j'«effectue une action» pour «atteindre un but»)
- Le but est obligatoire, mais on élude parfois l'action

*En tant qu'utilisateur,  
je peux payer  
en utilisant mon compte Paypal*

- **Conversation** : les détails d'implémentation sont négociés/discutés (oralement) par l'équipe et le PO (Rien n'interdit de garder une trace écrite des décisions).
- **Confirmation** : l'histoire est confirmée par les tests (rédigés si possible sur la carte)



# 1<sup>er</sup> « C » : « Card »

- Exemples :
  - « En tant qu'utilisateur je veux m'authentifier avec un login/mdp »
  - « En tant qu'administrateur, je veux bannir un user ayant tenu des propos déplacés »
  - « En tant qu'utilisateur, je veux changer de mot de passe »
  - « En tant qu'utilisateur, je veux re-cr  er un mdp en cas d'oubli»



## 3<sup>ème</sup> C : Confirmation

- Exemples :
  - « En tant qu'utilisateur je veux m'authentifier avec un login/mdp »
    - Je saisis : login:alain, mdp:azerty123, résultat attendu : je vois mon dashboard
  - « En tant qu'administrateur, je veux bannir un user ayant tenu des propos déplacés »
    - J'affiche un profil utilisateur, je clique sur «bannir»,résultat attendu : le user est banni
    - Je suis simple user, j'affiche un profil utilisateur, je ne vois pas le bouton «bannir»



## 3<sup>ème</sup> C : Confirmation

- Exemples :
  - « En tant qu'utilisateur, je veux changer de mot de passe »
    - Sur mon dashboard, je peux cliquer sur « changer mon mdp » (saisie ancien + 2 fois le nouveau) ; je reçois une confirmation par mail
  - « En tant qu'utilisateur, je peux re-cr  er un mdp en cas d'oubli »
    - Sur l'  cran de connexion, il y a un lien « Mot de passe oubli   »
    - Je re  ois un lien dans ma boite mail, je peux re-cr  er un mdp en cliquant dessus.

# Propriétés d'une story

I	independent	Des autres stories
N	negotiable	Courte (cf 1 <sup>er</sup> & 2 <sup>ème</sup> « C »)
V	valuable	Source de valeur (utile)
E	estimable	Par l'équipe (Points d'Effort)
S	small	Réalisable en une itération
T	testable	Cf. 3 <sup>ème</sup> « C »



# Product Backlog

- Ensemble des besoins métier, exprimés sous la forme d'**une liste de User Stories** (et/ou Epics), **estimés** et **priorisés**.
- Fait au début du projet...
- ... mais évolue en permanence !
- Responsabilité du Product Owner
- Souvent produit à l'issue de séances de Product Vision, Personas, User Journeys, Story Mapping, et Refinement...
  - <https://www.infoq.com/fr/presentations/art-maniement-exigences-agiles>





# Estimation

- Chaque US doit être estimée en difficulté, et non pas **en temps/durée!**
- On l'exprime en **Points d'Effort (dits «PE»)**, qui représentent la **difficulté ET l'incertitude de faisabilité** (forte corrélation).
- La difficulté est relative à une (ou quelques) US de référence qui représentent le niveau le plus **simple** (et pour laquelle on est donc **entièrement sûr** qu'elle est faisable)
- Les valeurs de points suivent (généralement) la suite de Fibonacci : 1 (story de référence), puis : 2, 3, 5, et parfois 8 (8 représente une très grande complexité ET une très grande incertitude).

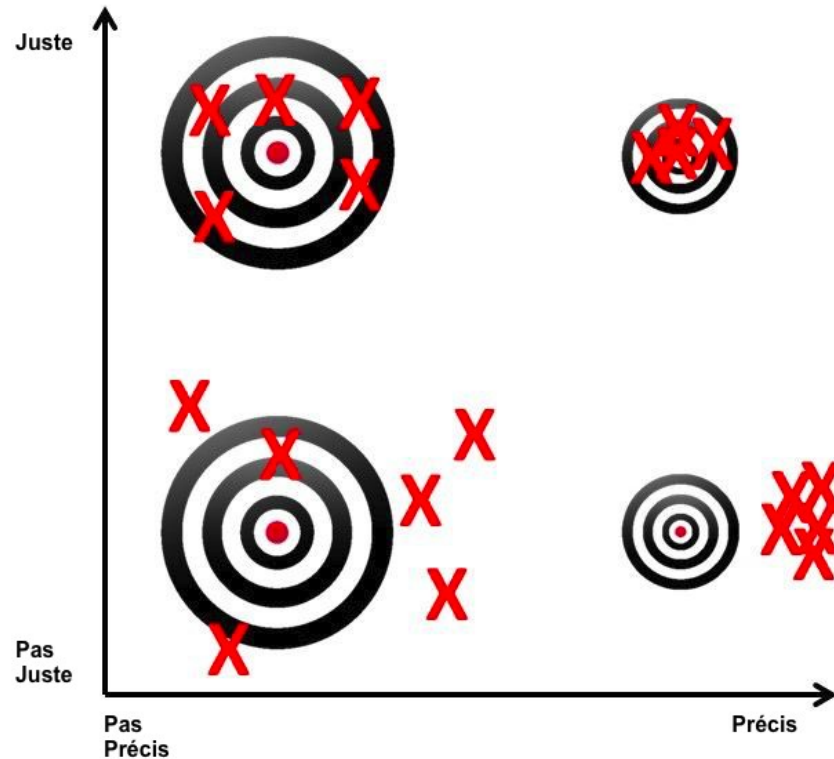


# Estimation

- On réalise généralement l'estimation des US lors d'une séance appelée **Planning Poker** (pas officiellement SCRUM, mais très courant).
- La plupart des équipes réservent la difficulté **8 (et plus)** aux stories trop complexes/incertaines pour être développées ; On les place alors **en attente** pour affinage ultérieur.
- L'estimation « en temps » pourra se faire **seulement ultérieurement**, en connaissant la **vélocité** de l'équipe (nombre de PE réalisés par dev & par jour; il s'agit d'une mesure **à posteriori**).

# Estimation

« *Mieux vaut être vaguement juste que précisément faux !* »





# Priorisation

- Le Product Owner doit pondérer chaque User-Story d'une priorité afin de décider le contenu des sprints.
- Critères possibles :
  - Gain financier
  - Coût de développement : **l'estimation de la difficulté technique est donc généralement faite au préalable !**
  - Satisfaction utilisateurs
- Différentes méthodes de priorisation
  - Informel (MoSCoW)
  - KANO
  - RICE, ...



# Priorisation empirique

- Le PO estime «au doigt mouillé»
- On utilise classiquement la notation **MoSCoW** (du plus urgent au moins urgent) pour catégoriser une US :
  - **MUST**
  - **SHOULD**
  - **COULD**
  - **WOULD** (ou **Won't Have**)



# Méthodes rationnelles

- Le PO peut prioriser les US avec des méthodes plus élaborées.
- Entre autres :
  - La méthode **KANO** classe les US en plusieurs catégories (essentielle, linéaire, excitantes) en fonction de leur désirabilité.
  - La méthode **RICE** pondère par un calcul l'importance d'une US pour l'utilisateur avec la difficulté à la réaliser.

# Exemple de Product Backlog

ID	Rôle En tant que...	Action Je peux...	But Pour...	Estimation (Points)	Priorité (MoSCoW)	Sprint (ID)	Statut
1	Responsable	publier une actualité	informer les clients	1	MUST	1	Fait
2	Utilisateur	rechercher des vols	voyager au meilleur coût	8	MUST	1	En cours
3	Utilisateur	utiliser une carte VISA	payer	3	SHOULD	1	A faire
4	Journaliste	publier des articles	informer les lecteurs	5	COULD	2	A faire
5	Éditeur	relire les posts	éviter les erreurs d'orthographe	2	COULD	2	A faire
6	Juriste	valider les posts	éviter les erreurs juridiques	1	WOULD	2	A faire



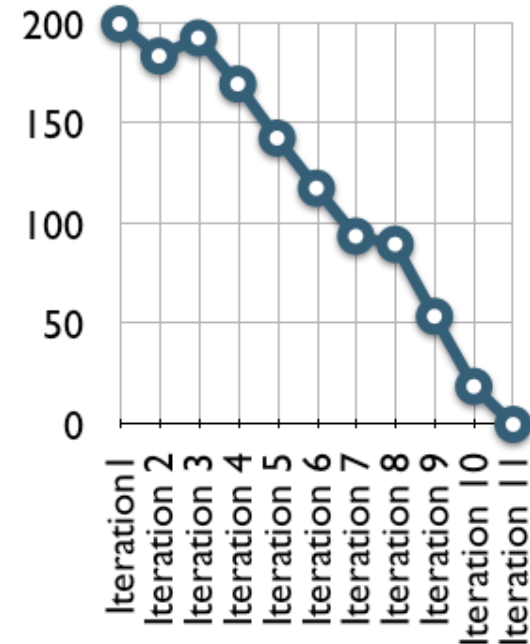
# Release Plan

- Quand toutes les stories sont estimées & priorisées
- Quand la vélocité initiale est déterminée (sprint 0 effectué)
- Les itérations/releases sont planifiées (date de début & de fin).
  - Une estimation (macro) de la taille de l'équipe (et/ou des congés) est nécessaire.
- Chaque US du Product Backlog est assignée à un sprint.
  - On en déduit un nombre de sprints nécessaires (estimation initiale)



# Release Plan

- Le **Release Burndown-Chart** est alors créé :  
graphique de visualisation de l'avancement global du projet
  - *abscisses : nombre de sprints*
  - *ordonnées : somme des PE du Product Backlog*





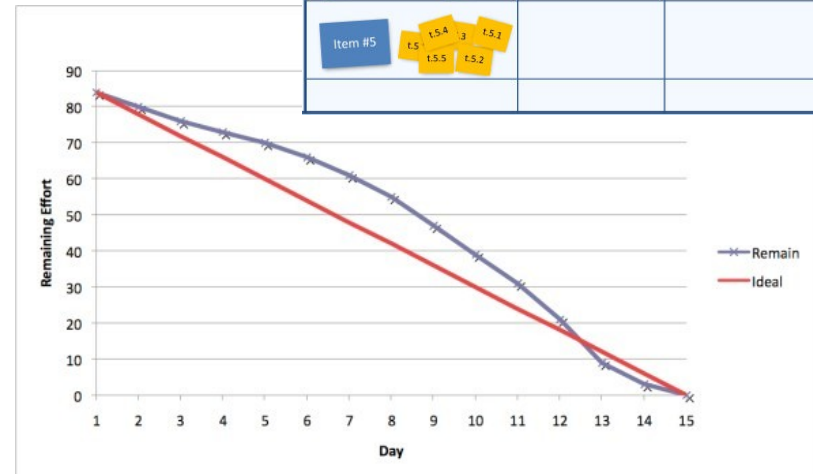
# Sprint Backlog

ID	Rôle En tant que...	Action Je peux...	But Pour...	Estimation (Points)	Priorité (MoSCoW)	Statut
1	Responsable	publier une actualité	informer les clients	1	MUST	Fait
2	Utilisateur	rechercher des vols	voyager au meilleur coût	8	MUST	En cours
3	Utilisateur	utiliser une carte VISA	payer	3	SHOULD	A faire

# Sprint Burndown Chart

- Positionnement des US sur la grille/radiateur
- Création du **Sprint Burndown Chart** :  
graphique indiquant l'évolution du sprint.
  - *abscisses : nombre de jours du sprint*
  - *ordonnées : somme des PE du Sprint Backlog*

To Do	Doing	Done
		Item #1 t.1.6 t.1.1 t.1.3 t.1.4 t.1.5 t.1.2
Item #2 t.2.7	t.2.6 t.2.5	t.2.1 t.2.3 t.2.4 t.2.2
Item #3 t.3.4 t.3.5 t.3.3	t.3.1	
Item #4 t.4.4 t.4.2 t.4.5 t.4.3		
Item #5 t.5.4 t.5.1 t.5.5 t.5.2		



A decorative graphic in the top-left corner consisting of a cluster of overlapping squares in various shades of blue, some with small white dots.

MERCI ! ...