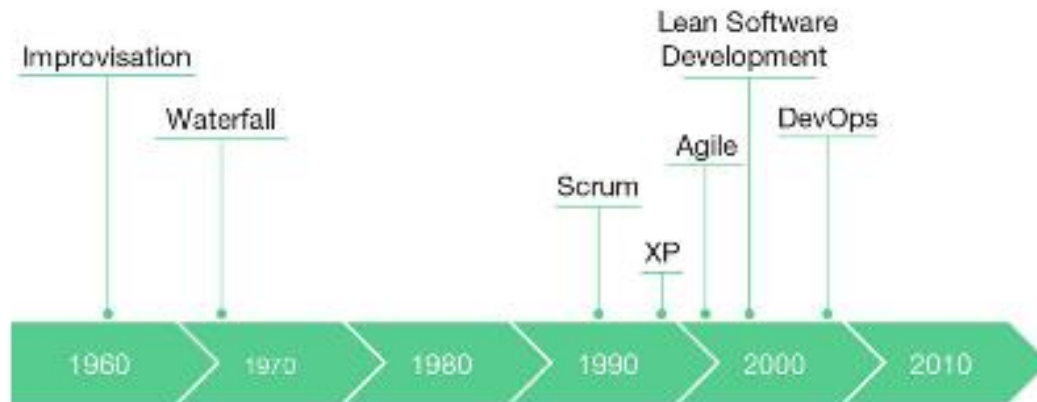


Gestion de projet informatique : SCRUM

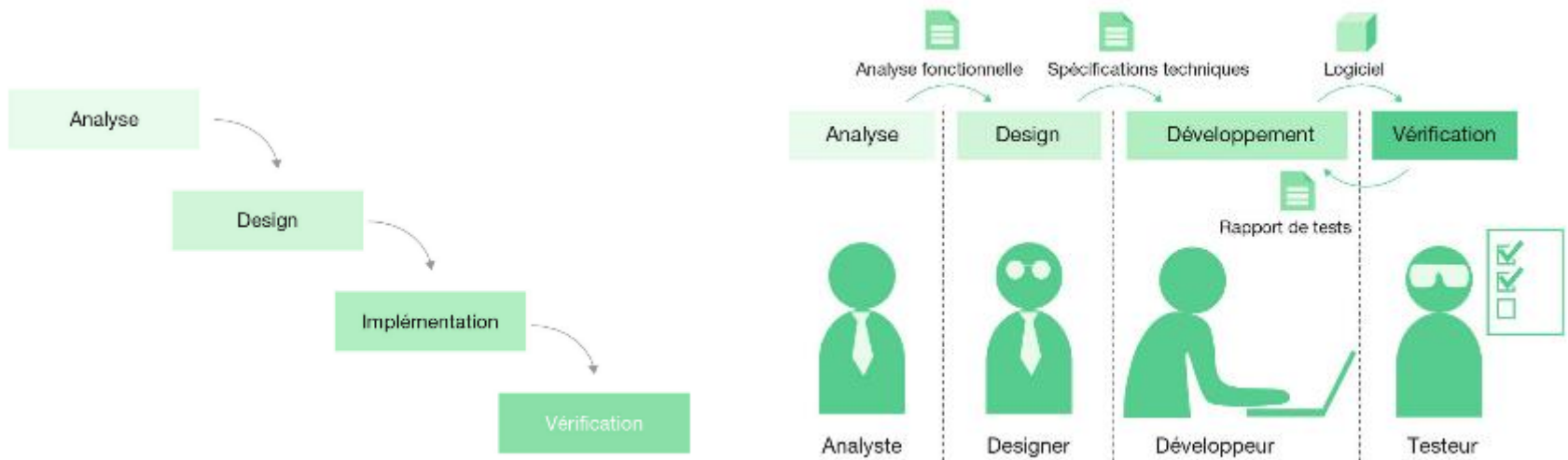
Introduction

De nombreuses méthodes



Waterfall

Méthode classique : en cascade

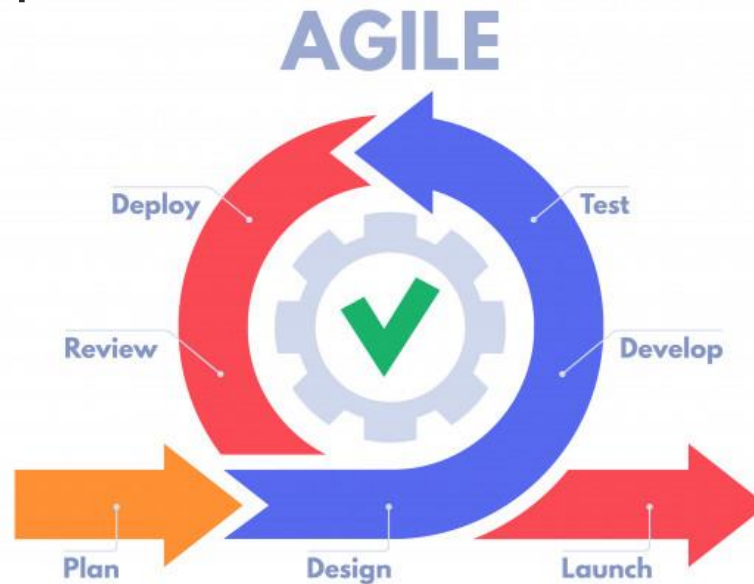


La documentation et les procédures sont très standardisées et sont les moyens de communications privilégiés



Lourdeur administrative

Méthode légère : par itérations successives



La collaboration entre des équipes auto-organisées et pluridisciplinaires avec le client est privilégiée



Faux-agile

Waterfall vs Agile



Waterfall: Scope is a fixed, Cost and Schedules are variables

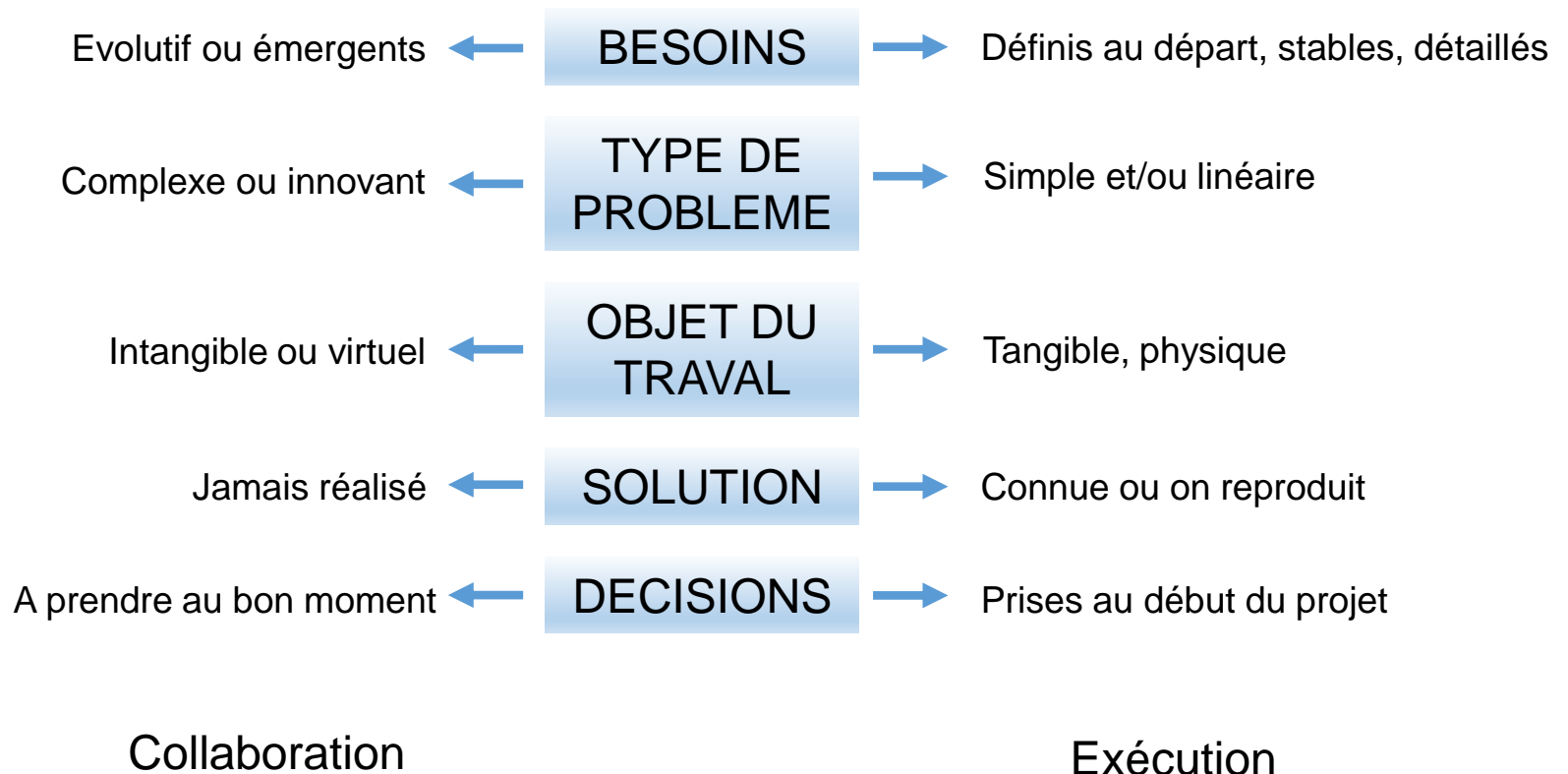


Agile: Cost and Schedules are fixed, Scope is variable

Que choisir ?

Méthode Agile

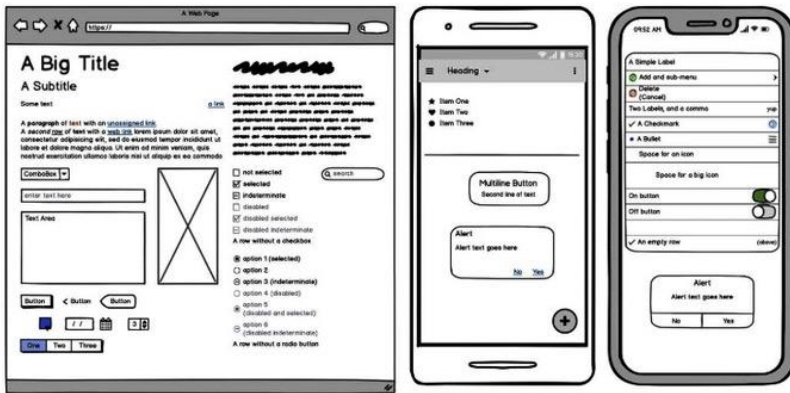
Méthode Waterfall



Exemples

Agile

- Création d'un logiciel



Waterfall

- Construction d'une maison



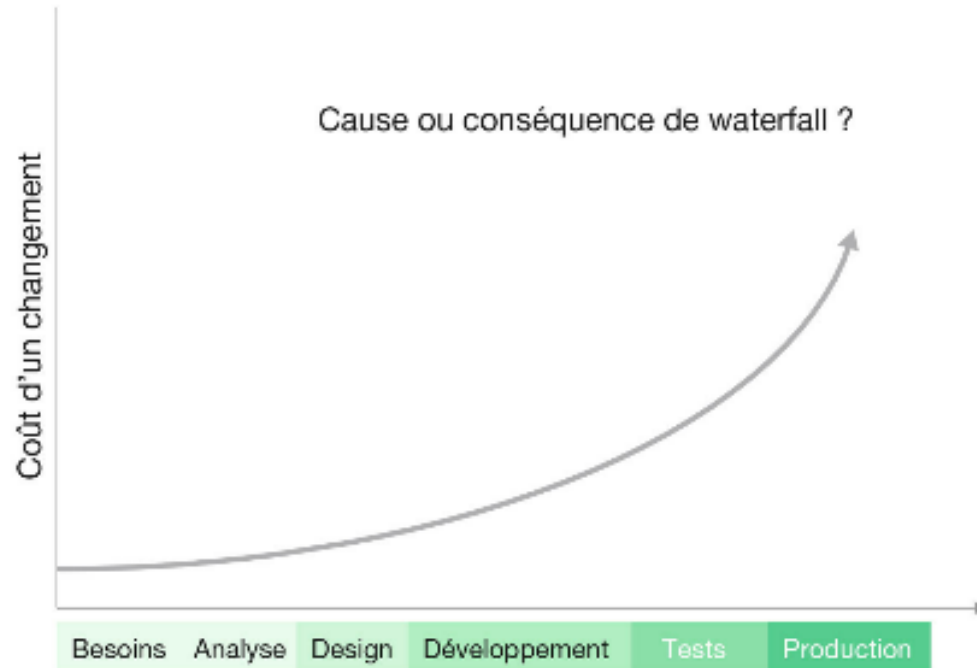
Agile

- Le cahier des charges est très évolutif. Ajout de fonctionnalités en cours
- Créativité lors du codage des fonctionnalités
- Les délais doivent être court pour répondre aux besoins

Waterfall

- Le cahier des charges est connu et ne change pas
- Les étapes de construction sont connues
- Les délais peuvent être long, le besoin reste

Désavantage de Waterfall



Dans les projets informatique les changements sont obligatoires pour obtenir un produit compétitif

Définition des méthodes Agiles

Agile Manifesto

Février 2001

Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.
Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent Beck
Mike Beedle
Arie van Bennekum
Alistair Cockburn
Ward Cunningham
Martin Fowler

James Grenning
Jim Highsmith
Andrew Hunt
Ron Jeffries
Jon Kern
Brian Marick

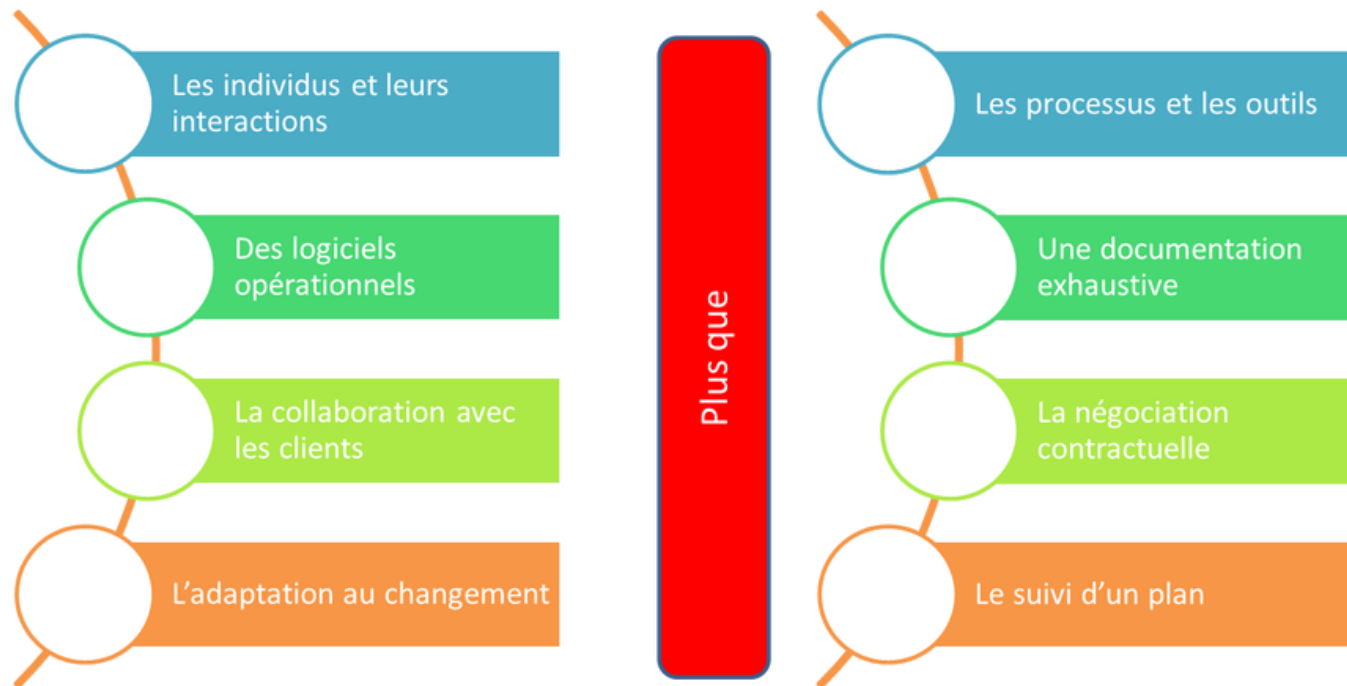
Robert C. Martin
Steve Mellor
Ken Schwaber
Jeff Sutherland
Dave Thomas

© 2001, the above authors

<https://agilemanifesto.org/iso/fr/manifesto.html>

Valeurs !

Manifeste Agile



+ 12 principes !

Les 12 principes (1/2)

- Notre plus haute priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à grande valeur ajoutée.
- Accueillez positivement les changements de besoins, même tard dans le projet. Les processus Agiles exploitent le changement pour donner un avantage compétitif au client.
- Livrez fréquemment un logiciel fonctionnel, dans des cycles de quelques semaines à quelques mois, avec une préférence pour les plus courts.
- Les utilisateurs ou leurs représentants et les développeurs doivent travailler ensemble quotidiennement tout au long du projet.
- Réalisez les projets avec des personnes motivées. Fournissez-leur l'environnement et le soutien dont elles ont besoin et faites-leur confiance pour atteindre les objectifs fixés.
- La méthode la plus simple et la plus efficace pour transmettre de l'information à l'équipe de développement et à l'intérieur de celle-ci est le dialogue en face à face.

Les 12 principes (2/2)

- Un logiciel fonctionnel est la principale mesure de progression d'un projet.
- Les processus agiles encouragent un rythme de développement soutenable. Ensemble, les commanditaires, les développeurs et les utilisateurs devraient être capables de maintenir indéfiniment un rythme constant.
- Une attention continue à l'excellence technique et à un bon design.
- La simplicité – c'est-à-dire l'art de minimiser la quantité de travail inutile – est essentielle.
- Les meilleures architectures, spécifications et conceptions émergent d'équipes auto-organisées.
- À intervalles réguliers, l'équipe réfléchit aux moyens possibles de devenir plus efficace. Puis elle s'adapte et modifie son fonctionnement en conséquence

Une méthode Agile : SRUM

Définition

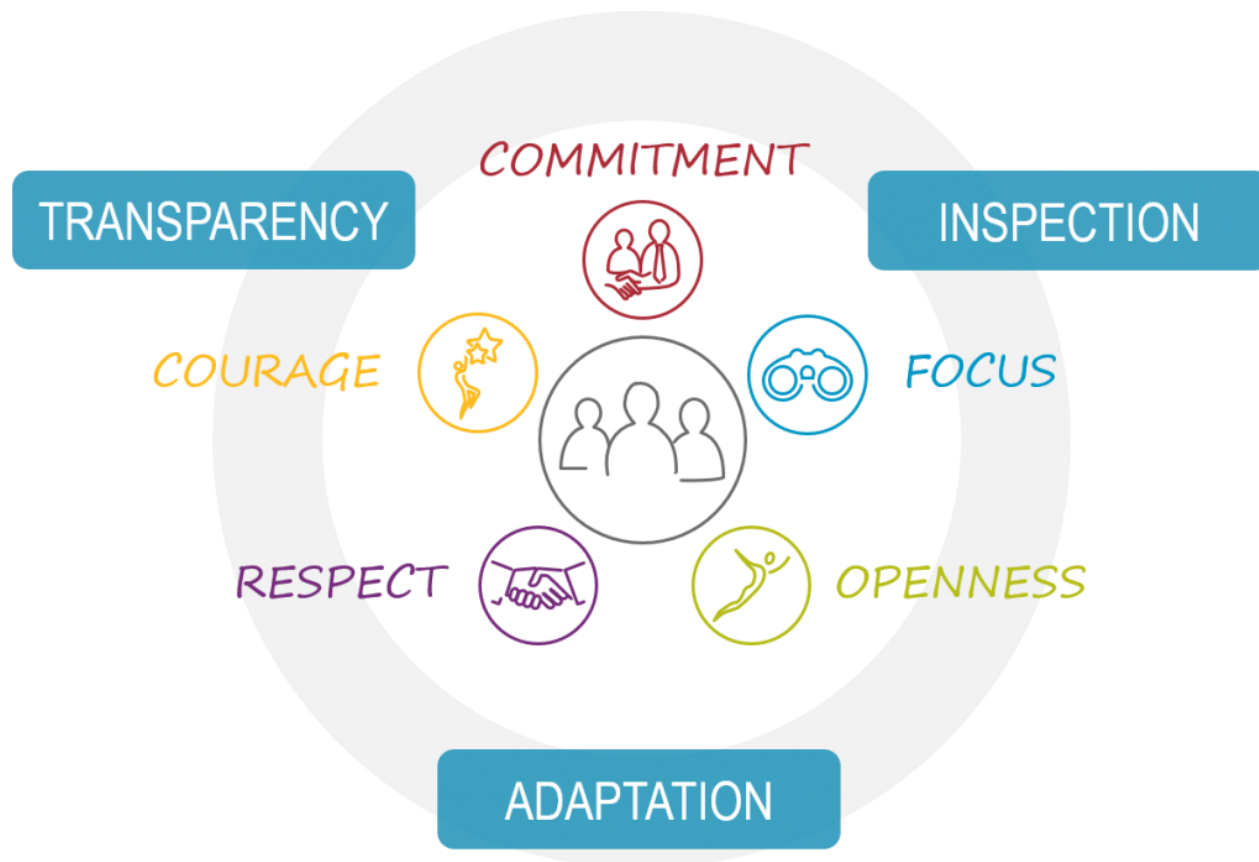


Méthode Agile décrite par
Ken Schwaber et Jeff Sutherland



Framework ! Donc on prend ce que l'on a besoin

Piliers de SCRUM



Vue générale



Scrum Master



L'organisation

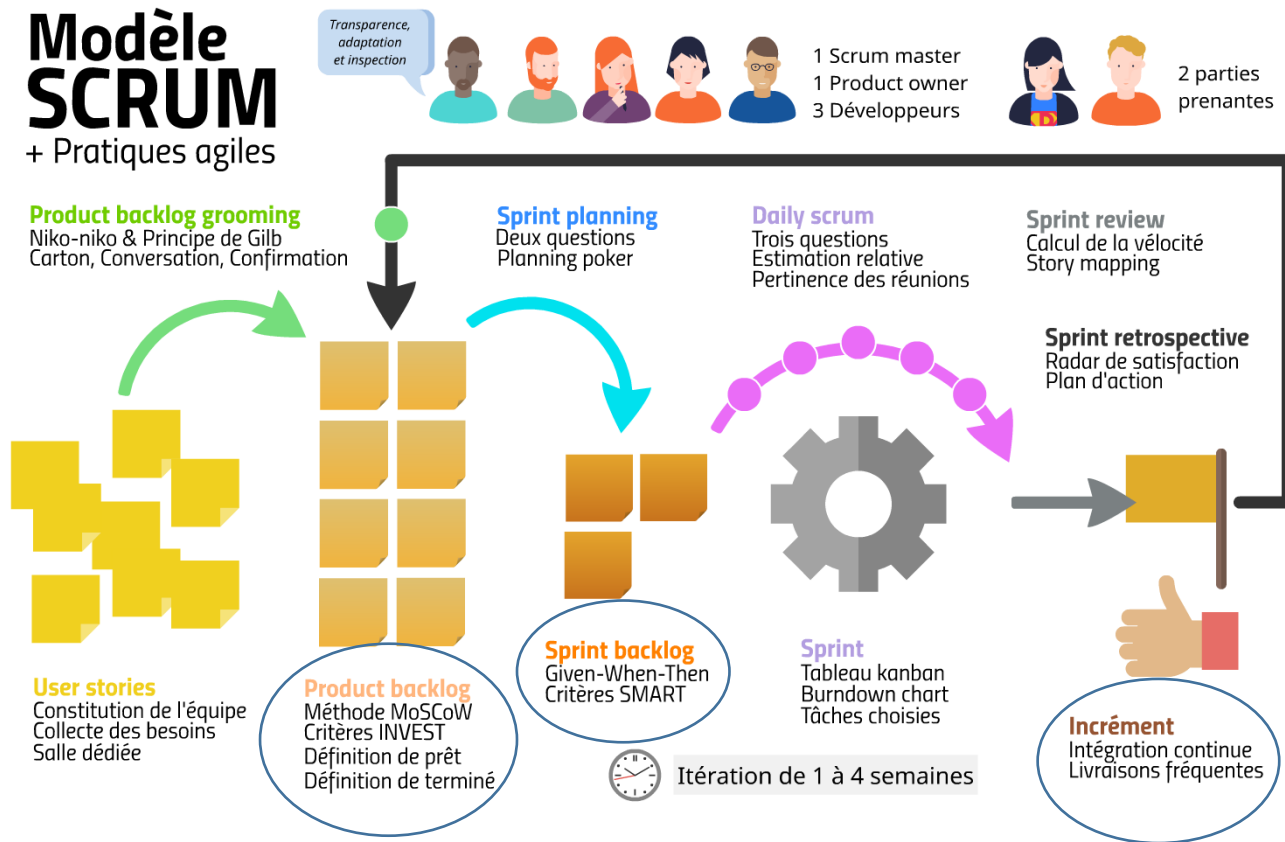


Client
représentant du client

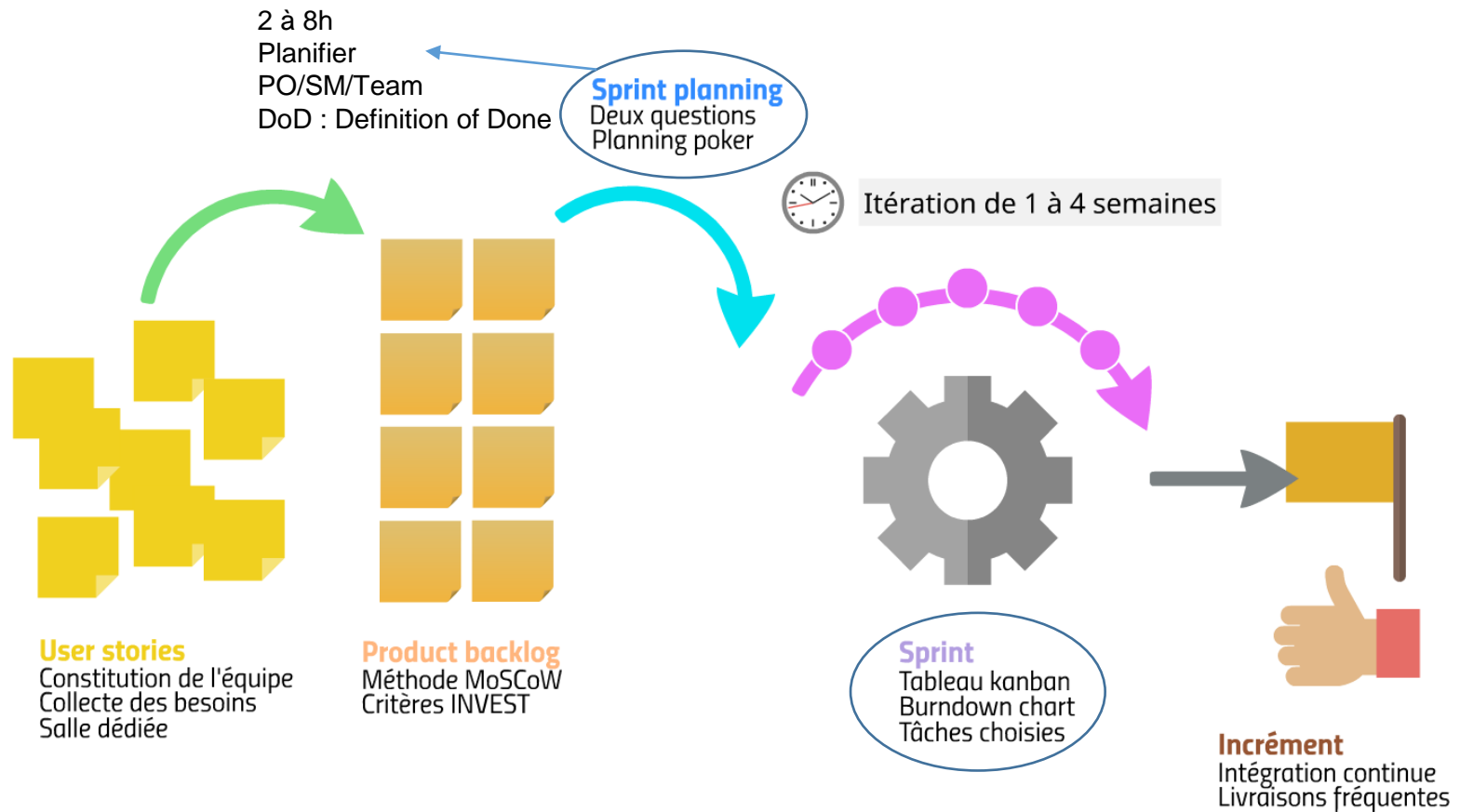
Facilitateur

Auto-organisée
Pluridisciplinaire
3 à 9 personnes

Modèle SCRUM + Pratiques agiles



Cérémoniaux (1)



Cérémoniaux (2)



3 questions :
Qu'est ce qui a été fait hier ?
Quel est le plan de la journée ?
Quels sont les obstacles ?

Cérémoniaux (3)



Présentation devant le PO
Révision du product backlog (si nécessaire)

Cérémoniaux (4)

Sprint retrospective



Max 3h
Feed back
Plan d'amélioration
SM + Team

User story ?

- Ecrire les besoins du client
 - Simple (on utilise les mots du client)
 - Succinct (1 ou 2 phrases)
 - Non technique
 - Avantage
- Ecrit généralement sur des post-it
- Contient un titre, une description et une estimation
- Base de discussion
- Décrit la fonctionnalité à développer mais pas comment

User story : l'écriture

EN TANT QUE {X}
JE VEUX {Y}
AFIN DE {Z}

USER STORY

Y EST UNE FONCTIONNALITÉ

Z EST LE BÉNÉFICE DE LA FONCTIONNALITÉ

X EST LA PERSONNE (OU RÔLE) QUI VA EN BÉNÉFICIER

VOUS IDENTIFIEZ AINSI LA VALEUR APPORTÉE PAR
L'HISTOIRE AU MOMENT OÙ ELLE EST DÉFINIE.

En tant que Client

Je veux pouvoir me connecter avec mon compte Google

Afin de centraliser mes informations de connexion

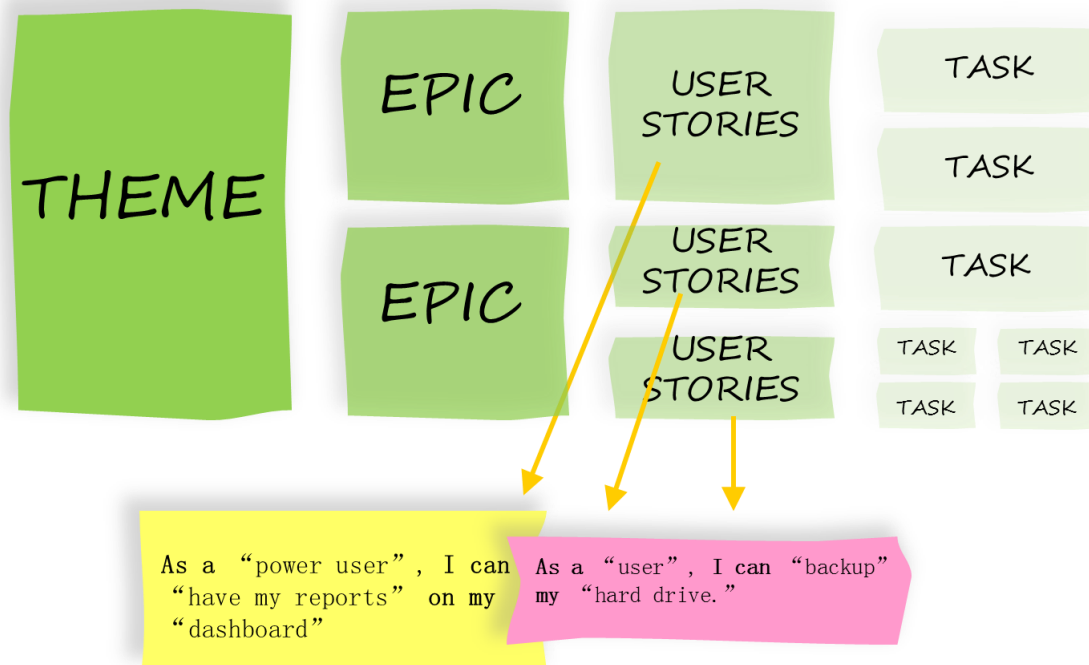
User story : Modèle

- As a <role>, I Want <goal/desire> so that <benefit>
- As a <role>, I Want <goal/desire>
- In order to <receive benefit> as a <rôle>, I want <goal/desire>
- ...

« Si vous vous inquiétez de la façon d'écrire vos user stories, vous passez à côté de l'essentiel. » Michaël D. Hill

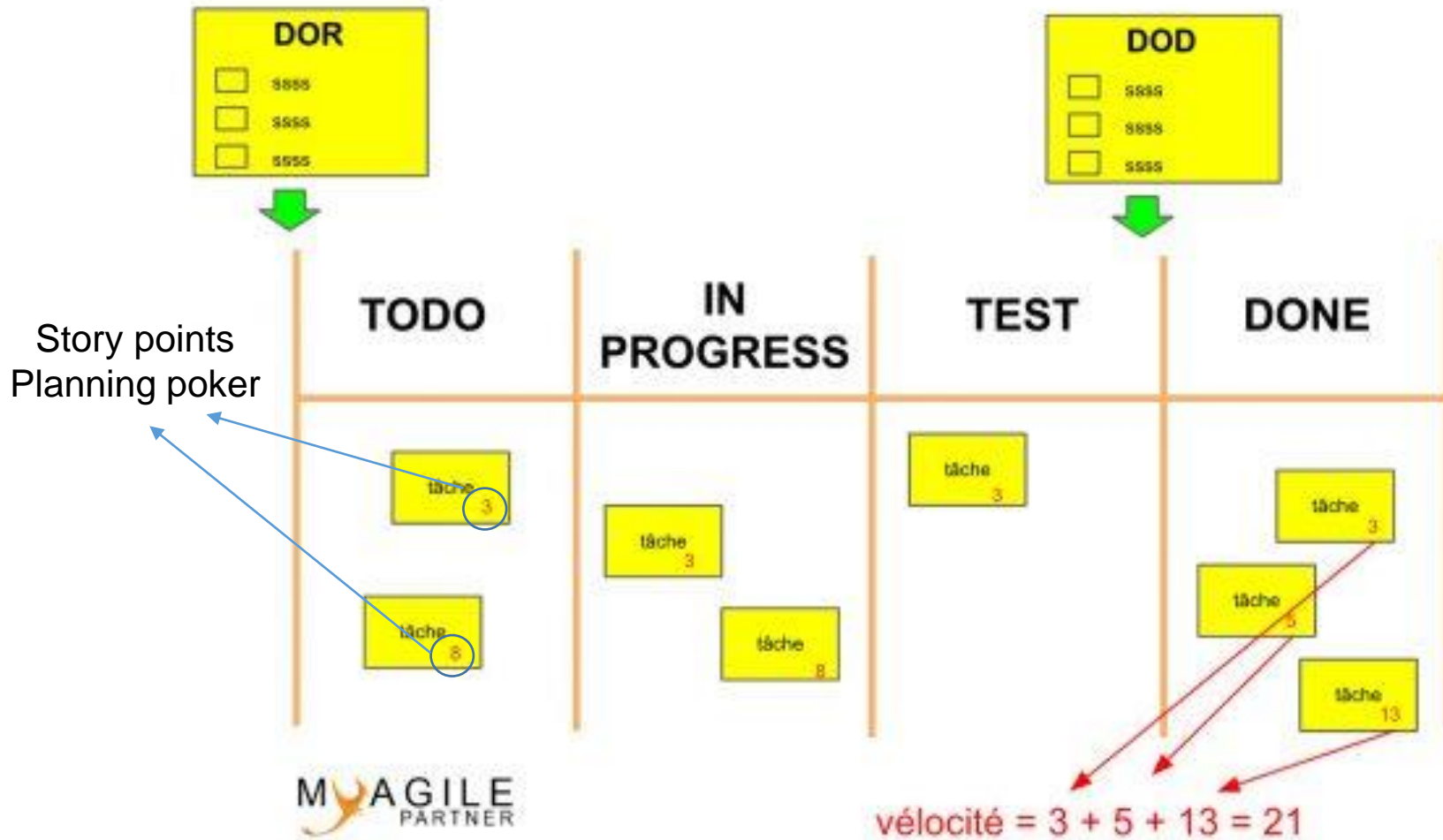
User story : Epic

USER STORIES



www.agile-scrum.be

Scrum board & Vélocité



Aligner l'équipe en définissant correctement les tâches

DOR : Definition Of Ready

- Décrit correctement comment doit être écrit une user story
- Définit le point max d'une tâche avant de la scinder
- INVEST/SMART
- Test
- ...

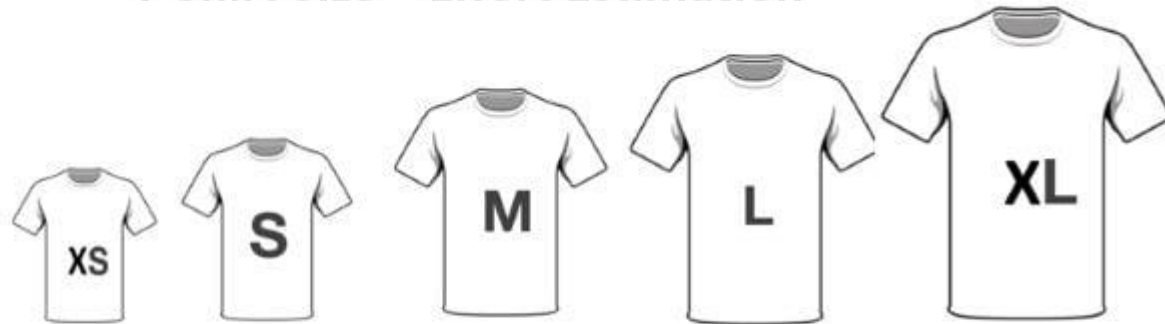
DOD : Definition Of Done

- Décrit correctement quand une tâche est terminée
- Test réalisé
- Documentation rédigée
- ...

- Evaluer la charge de travail d'une tâche
- Valeurs numériques : suite de Fibonacci
1,2,3,5,8,13,20,40...
- Plus facile d'estimer des petites tâches
- Heures vs Story points
- ESTIMATION : ne pas y passer la journée
- Eviter l'influence des autres
- Techniques
 - T-Shirt sizing
 - Jeux de cartes

T-Shirt sizing

T-Shirt Size – Effort Estimation



T-Shirt Size	XS	S	M	L	XL
Estimate	1	2	3	5	8

Source: PMTips.xyz

Planning poker

- Estimation en 3 étapes

1^{er} étape :



2^e étape :



3^e étape :

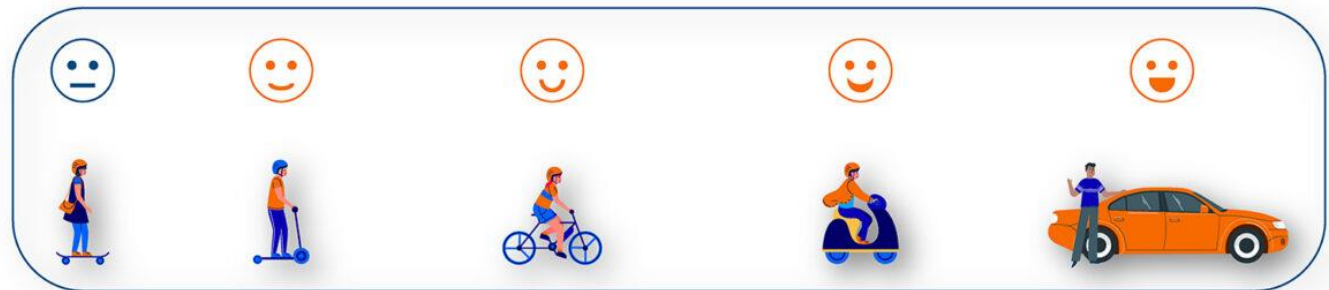


Estimation = 5

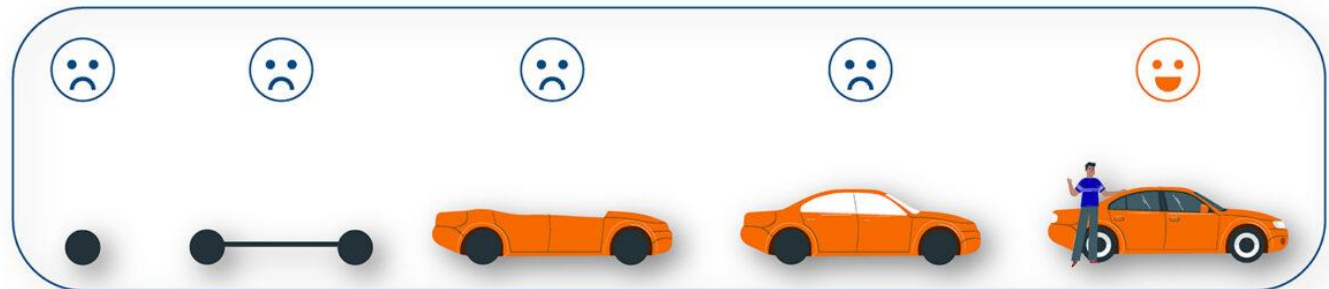
- <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-French.pdf>

MVP - Minimum Viable Product (le produit minimum viable)

MVP

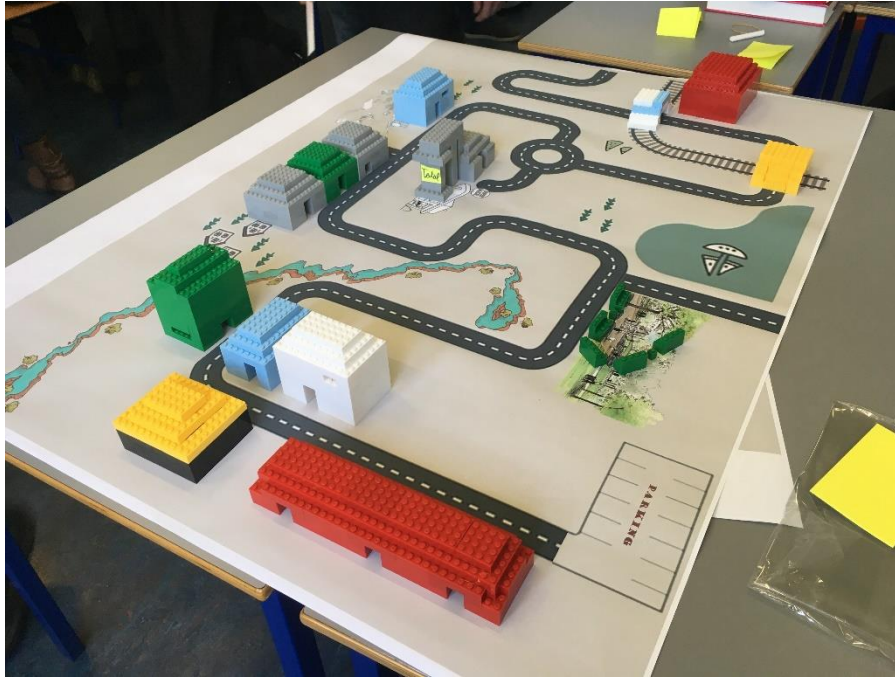


Non MVP



Lego4SCRUM

Notre but :



Product backlog

- Gare
- Ponts pour le train x2
- Maison 4 façades avec étage x2
- Maison 4 façade sans étage x2
- Groupe de 3 maisons mitoyennes x2
- Immeuble de bureaux x1
- Port de plaisance
- Aménagement espace vert
- Portique avec barrière pour le parking
- Station-service
- Magasin
- Ecole
- Des arrêts de bus
- Bar sur la place
- Restaurant sur la place
- Fontaine sur la place
- Un monument à la mémoire d'Alan Turing dans le parc

- Sprint planning 0
 - Sprint 0 de 10 min (daily sprint à 5min)
 - Sprint review 0
 - Retrospective 0
 - Sprint planning 1
 - Sprint 1 de 16 min (daily sprint à 8min)
 - Sprint review 1
 - Retrospective 1
- } 2 ou 3x