

# Comprendre la démarche Agile

---

04/12/2023 – 05/12/2023

**Glodie Tshimini**



# Glodie Tshimini

Consultant Formateur et développeur depuis 2017

**Certifié Professional Scrum Developer (PSD I)**

Certifié 2AICONCEPT Expert Trainer

Certifié 2AICONCEPT IT Expert Trainer At POE

Email : [contact@tshimini.fr](mailto:contact@tshimini.fr)

# **HORAIRES ET PAUSES**

## Lundi

- 9h00 - 12h30
- 13h30 - 17h30

## Mardi

- 09h00 - 12h30
- 13h30 - 16h30

## Les pauses

- Matin : 15 minutes
- Déjeuner : 1 heure
- Après-midi : 15 minutes

# ÉMARGEMENT ET ÉVALUATION FORMATEUR

Chaque matin et chaque après-midi, vous devrez signer les feuilles d'émargement. Les "X" et initiales ne sont pas autorisés, merci de bien vouloir émarger avec une signature.

Le dernier jour de la formation, avant 14h00, vous aurez à saisir une évaluation de la formation. Cette évaluation est obligatoire, il doit être rempli consciencieusement.

# OBJECTIFS PÉDAGOGIQUES

- Décrire ce qui différencie les approches prédictives des approches adaptatives
- Mettre en œuvre les principales approches Agile (Scrum, Kanban, Lean...)
- Présenter un aperçu des éléments / techniques de base
- Décrire leur mise en œuvre dans un projet.

# NIVEAU REQUIS ET PUBLIC CONCERNÉ

## Niveau requis

- Avoir un niveau d'anglais suffisant pour comprendre les termes utilisés.
- Expérience souhaitée en gestion de projet ou participation occasionnelle à un projet dans le cadre de ses missions.
- Aucun prérequis technique n'est demandé.

## Public concerné

- Directeurs
- Chefs de projets
- Acteurs projets ou toute personne devant intervenir dans un projet Agile, en tant que contributeur ou hiérarchique de contributeurs.

# DÉROULEMENT DE LA FORMATION

- 40 % théorie 60% pratique
- *Daily meeting*
- Exercices immédiats d'application du cours
- Les énoncés des exercices, TP et corrections seront accessibles depuis un [dépôt public GitHub](#)

# GitHub de la formation



<https://github.com/glo10/agi-meth-04122023>



# Un formateur à votre écoute



Google Forms

<https://docs.google.com/forms/d/e/1FAIpQLScJZ5khDfQybrCg0yAyd--CKWLurTfS3L5NDf2Sujj1FxArEQ/viewform>



# PRÉSENTATIONS

**Merci de bien vouloir vous présenter individuellement.**

Merci de bien vouloir m'indiquer quelles sont vos attentes suite à cette formation.

# PLAN DU COURS

---



## Chapitres

- I. Approches prédictive
- II. Approches agiles
- III. Les ateliers
- IV. Un projet avec le cadre agile  
SCRUM

## Annexe

- [Manifeste agile](#)
- [Principes agiles](#)

# EXERCICE 1 : expression des besoins d'un client

0-exercices/ex1.md

# I. Approches prédictives

---



# Vision d'un projet

Source image anyideas



Comment le client a exprimé son besoin



Comment le chef de projet l'a compris



Comment l'ingénieur l'a conçu



Comment le programmeur l'a écrit



Comment le responsable des ventes l'a décrit



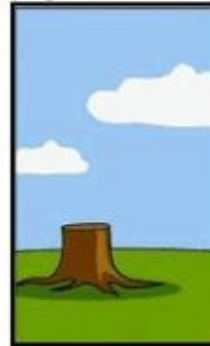
Comment le projet a été documenté



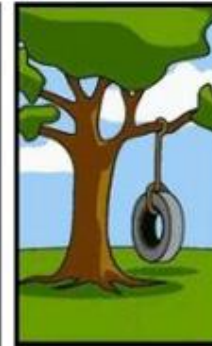
Ce qui a finalement été installé



Comment le client a été facturé



Comment la hotline répond aux demandes



Ce dont le client avait réellement besoin

## Fonctionnement

- Trouve ses origines dans le bâtiment
- Séquentielle
- Besoins spécifiés dans un cahier des charges
- Planification rigoureuse dans un diagramme de Gantt

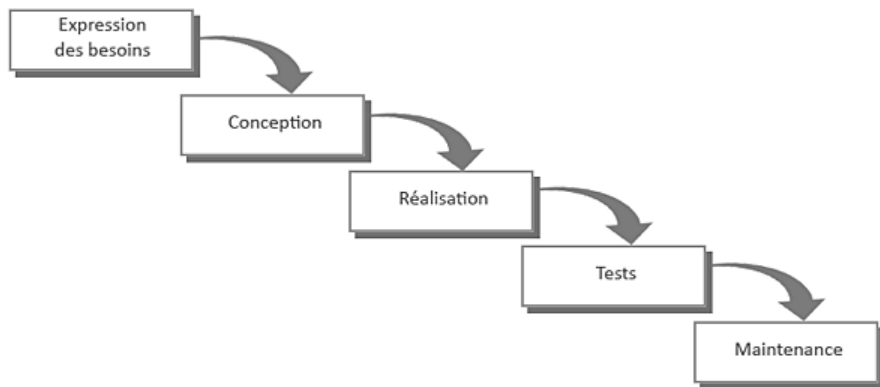
## Faiblesse

- Très peu d'interaction avec le client
- Peu flexible
- Peu d'adaptabilité



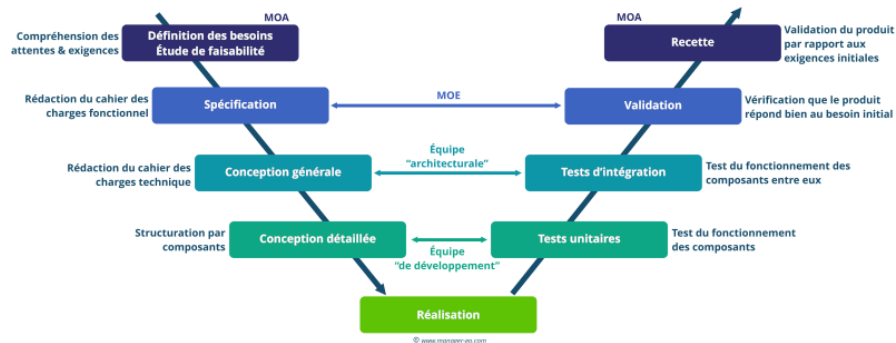
# CYCLE EN CASCADE

[Source image editions-eni.fr](http://source.image.editions-eni.fr)



- Apparition en 1956 Hebert D.Benington
- Une étape doit être finie à 100% avant de passer à l'étape suivante (linéaire)
- Planification rigoureuse et précise
- Besoin prévisible et stable dans le temps
- Repose sur spécifications fonctionnelles, techniques et globalement sur le cahier des charges
- Une mauvaise définition du besoin aura un grand impact sur la durée et le coût du projet
- Moins d'interaction entre les parties prenantes
- Intervention des utilisateurs finaux qu'à la fin

## Source image manageo-go



- Déclinaison de la méthode cascade
- Place plus importante des tests dans le cycle
- Une phrase descendante et ascendante qui peuvent s'effectuer en parallèle
- MOA
  - Maitrise d'ouvrage
  - Chargé de définir les besoins
- MOE
  - Maitrise d'œuvre
  - Chargé de transformer le besoin en solutions digitales (réalisation)

## II. APPROCHES AGILES

---





## EXERCICE 2 : Mind Mapping Agilité

0-exercices/ex2.md

# PRÉSENTATION

---



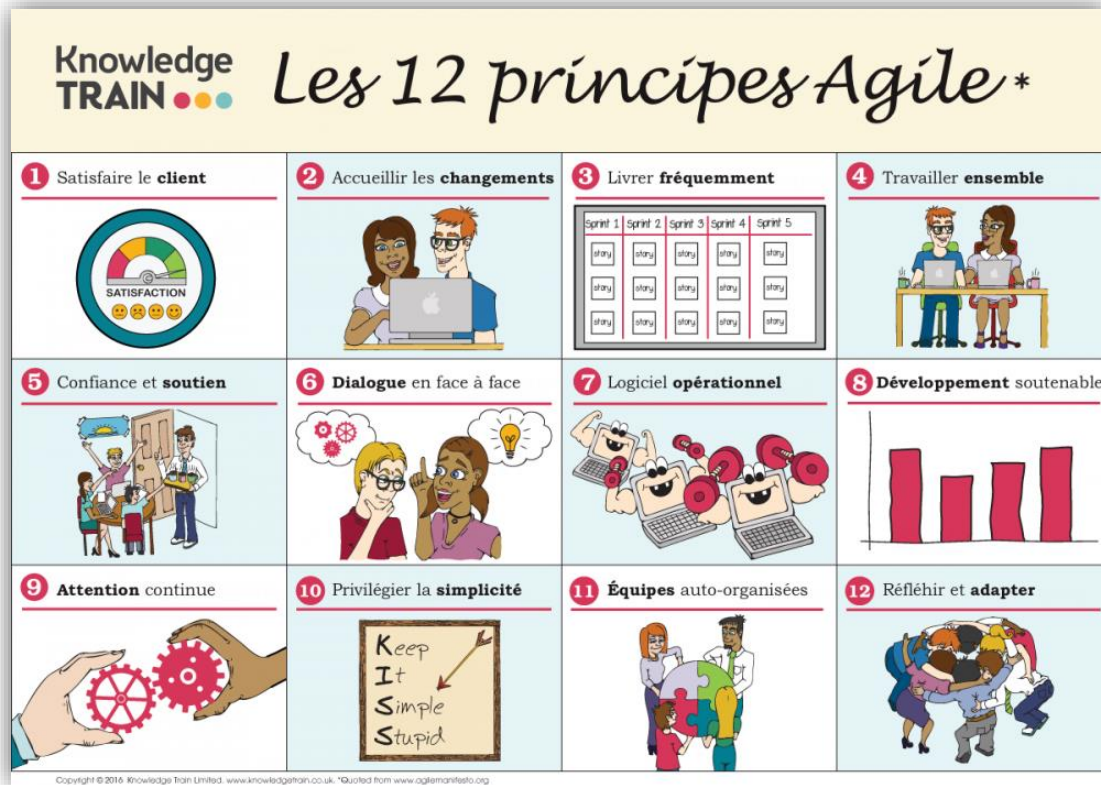
## 4 VALEURS AGILES (MANIFESTE AGILE)

1. Les individus et les interactions, de préférence aux processus et aux outils.
2. Des solutions opérationnelles, de préférence à une documentation exhaustive.
3. La collaboration avec les clients, de préférence aux négociations contractuelles.
4. La réponse au changement, de préférence au respect d'un plan.

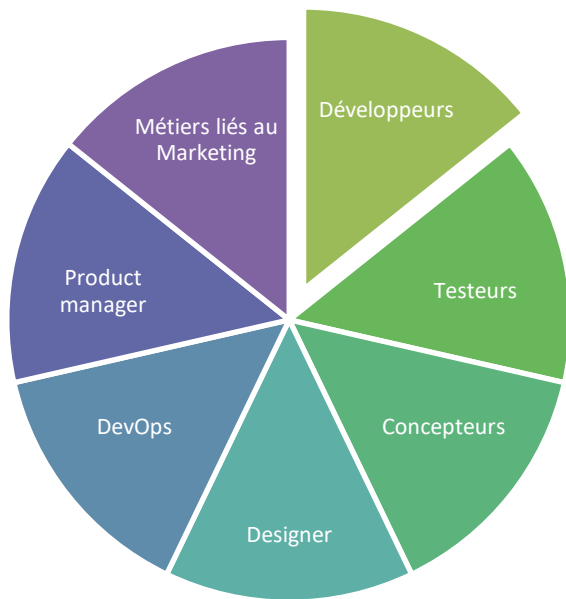
Précisément, même si les éléments à droite ont de la valeur, nous reconnaissons davantage de valeur dans les éléments à gauche.

# LES 12 PRINCIPES AGILES

[source image Wikiagile](#)



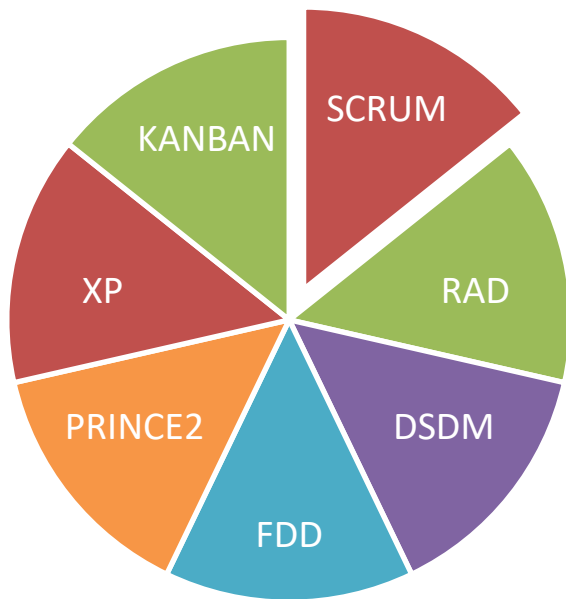
# Quelques membres des équipes agiles



- Les équipes agiles sont pluridisciplinaires
- Les équipes s'organisent, travaillent et prennent des décisions ensemble
- L'accès est mis sur la collaboration des membres de l'équipe plutôt que des individualités (compétences spécifiques)



# Les quelques méthodes agiles



- RAD (Rapid Application Development)
  - Livrer rapidement les fonctionnalités à forte valeur pour un ROI rapide
- DSDM (Dynamic Software Development Method)
  - Collaboration étroite entre les utilisateurs finaux et les devs
- FDD (Feature Driven Development)
  - Réduire les risques avec des cycles de développement très court
- Prince2 (Projects In Controlled Environments)
  - Tient compte des facteurs d'environnement du projet susceptibles d'impacter sa réussite.

**KANBAN**



# CARACTÉRISTIQUES DU FRAMEWORK KANBAN

[Source image Ionis](#)



- Framework (cadre de travail) léger et visuel
- Facile à mettre en place
- Travail en flux continu
- Mise en place par la modélisation du flux de travail
- Stories est le stock de l'ensemble du travail
- Amélioration possible avec le Work In Process (WIP)
  - Nombre maximal des tâches dans une colonne

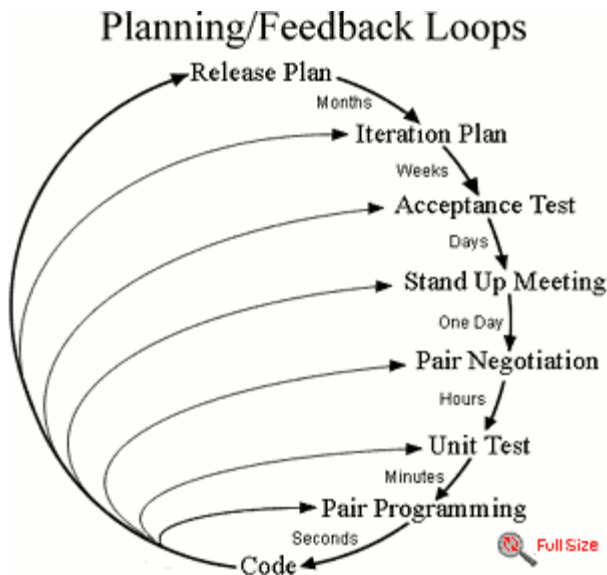
# EXTREME PROGRAMMING

---



# CARACTÉRISTIQUES DU FRAMEWORK XP

[Source image extremeprogramming.org](http://sourceimage.extremeprogramming.org)



## 4 principes

1. Communication : les développeurs communiquent directement avec les utilisateurs finaux
  2. Simplicité: applique le principe *KISS (Keep It Simple and Stupid)* = choisir à chaque fois la solution la plus simple.
  3. Retour d'expérience : impliqué l'utilisateur dans les tests, priorisé les fonctionnalités, améliorer la qualité du code
  4. Livraison : fréquente et rapide
- Les bonnes pratiques de développement sont appliqués et poussées à l'extrême



## **EXERCICE 3 : AVANTAGES ET INCONVÉNIENTS DES MÉTHODES AGILES**

0-exercices/ex3.md

## III. LES ATELIERS

---

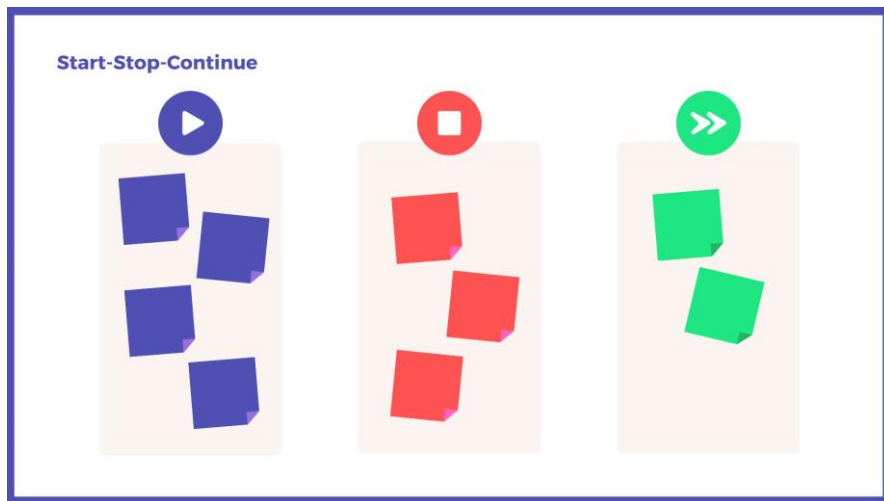


# Les ateliers dans les méthodes agiles

- Les ateliers sont beaucoup utilisés dans les méthodes agiles car ils peuvent avoir une ou plusieurs objectifs parmi la liste non-exhaustive ci-dessous
  - Permettre aux équipes d'être plus créatif et d'innover
  - Permettre de partager ou construire une vision commune
  - Mettre en place de nos nouveaux processus
  - Renforcer la cohésion de l'équipe
  - Permettre une meilleure communication
  - Aider l'équipe à se projeter
  - Aider l'équipe à prioriser
  - Partager des bonnes pratiques
  - Résoudre des conflits
  - Aider l'équipe à s'autogérer
  - Etc.



## Source image infusion



## Mise en œuvre

- À faire entre 2 itérations de réalisation (fin de la précédente et avant d'attaquer la nouvelle)
- 1 Animateur et des participants divisés en sous groupe
- Phase 1 : présentation et préparation avec les post-it
- Phase 2, les groupes notent dans chaque colonne :
  - Les nouvelles choses à faire (Start)
  - Les pratiques à arrêter (Stop)
  - Les bonnes pratiques à garder (Continue)
- Phase 3 ; débriefe et regroupement des éléments par thématique
- Phase 4 : l'animateur et les participants priorisent et concluent l'atelier

# Prune the product tree

[Source image productplan.com](https://productplan.com)

## Product Tree

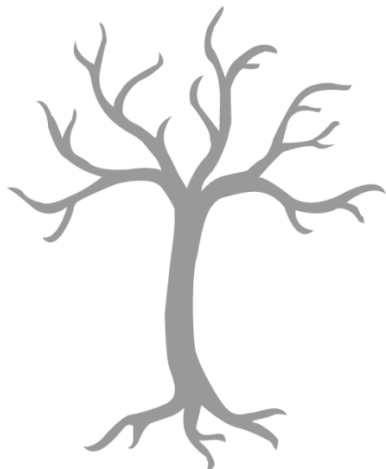
**Trunk:** Core or existing features

**Branches:** Primary product or system functionalities

**Roots:** Technical requirements or infrastructure

**Leaves:** New Ideas

 ProductPlan

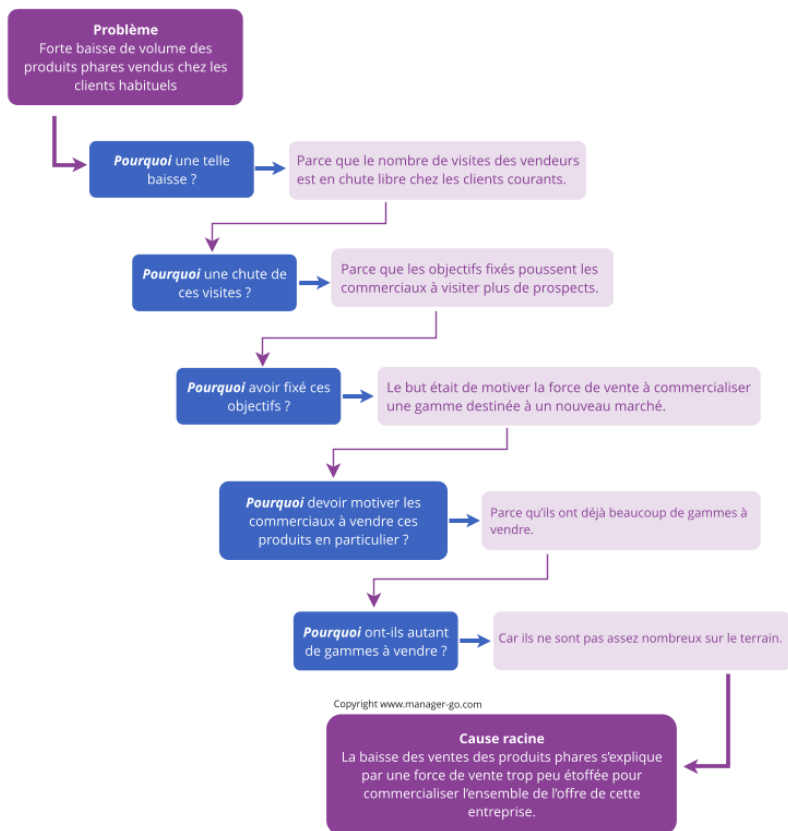


- À utiliser par exemple, dans le cadre du démarrage d'un nouveau projet, la construction d'une vision commune du produit, des améliorations diverses et variées, etc.
- Les participants réfléchissent à des idées ou fonctionnalités qu'ils notent sur un post-it puis ils doivent les regrouper par thème sans se parler
- Enfin, l'animateur et les participants discutent et engagent le débat pour placer les éléments par valeur au niveau
  - Du tronc : indispensables (+++)
  - Branches : importantes (++)
  - Feuilles : confortables (+)

## Perceptions croisées

- Aider une équipe à collaborer pour régler des problèmes, incompréhension et conflits en prenant en compte les difficultés, obstacles, contraintes, etc. des uns et des autres
- Chaque sous-groupe ou participant expose la manière dont elle pense être perçue par les autres par l'intermédiaire d'un porte-parole
- Les autres sous-groupes ou individus écoutent et puis réagissent aux propos pour confirmer, infirmer ou nuancer leurs propos
- L'animateur et participants procèdent à la synthèse en essayant de trouver des solutions ensembles, prendre des décisions ou actions à effectuer.

# Arbre des 5 pourquoi (source image manager-go)



- Pour trouver les racines des problèmes, identifier les besoins profonds d'un client, trouver les origines d'un conflit, Etc.
- L'animateur et les participants définissent un problème à analyser
- L'animateur dessine un arbre avec 5 racines
- À 5 reprises sous forme d'une question *pourquoi...* Les participants analysent la question et essaient de trouver à chaque fois une à plusieurs origines qui sont notées au niveau de la racine
- À chaque tour, l'une des origines trouvées précédemment est formulé sous forme de question *pourquoi...*

# Design thinking

- Méthode de conception et résolution d'un problème centré sur les besoins et attentes du client/utilisateurs finaux
- Atelier à faire avec les participants hétérogènes venant du marketing, management, opérationnel
- Phase 1 empathique : se mettre à la place des utilisateurs finaux et verbaliser leurs besoins, leurs comportements d'utilisation, les cas d'usage
- Phase 2 d'idéation : traiter les informations et formuler une problématique pour améliorer l'expérience utilisateur en trouvant des nouvelles idées
  - Comment pourrait-on... ?
  - Comment satisfaire... ?
  - Par quels moyens... ?
- Phase 3 de maquettage/prototypage
  - Rendre ses idées concrètes par les maquettes ou prototypes à faire tester aux utilisateurs finaux



## EXERCICE 4 : Boîtes à idées

0-exercices/ex4.md

## IV. UN PROJET AVEC LE CADRE SCRUM

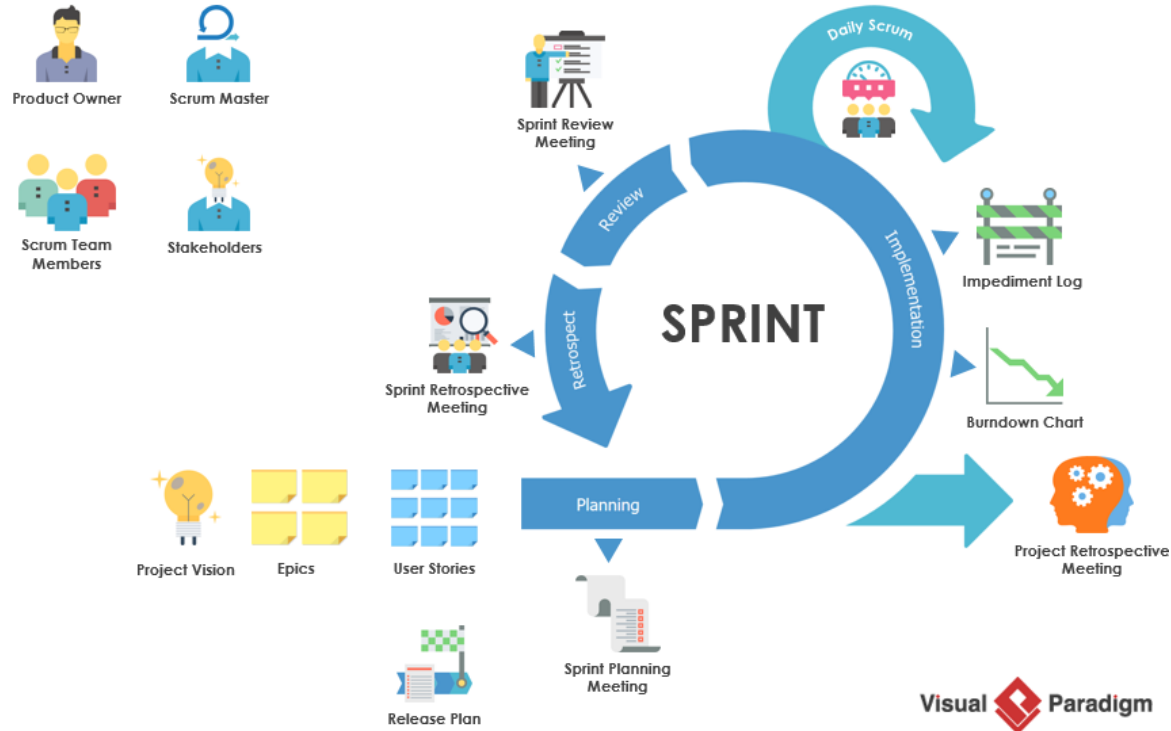
---



# CYCLE SCRUM

source image visual paradigm

## The Agile – Scrum Framework





# HISTORIQUE

- Développé dans les années 90
- Créateurs
  - [Ken Schwaber](#)
  - [Jeff Sutherland](#)
- *SCRUM GUIDE*
  - Document officiel qui partage la vision de *SCRUM* par ses créateurs
  - Les versions
    - Première en 2010
    - Précédente version 2017
    - Version actuelle 2020

# FRAMEWORK SCRUM

*SCRUM* est

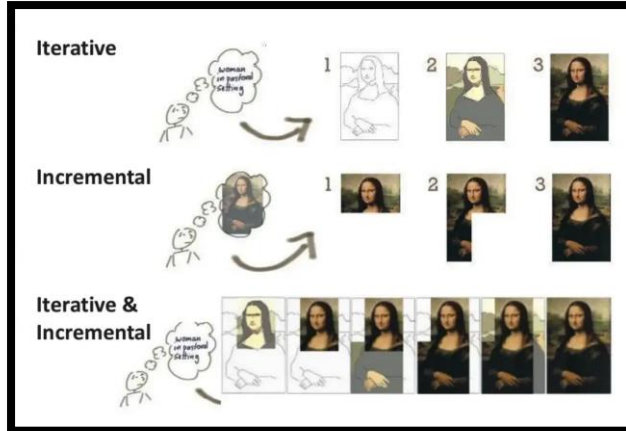
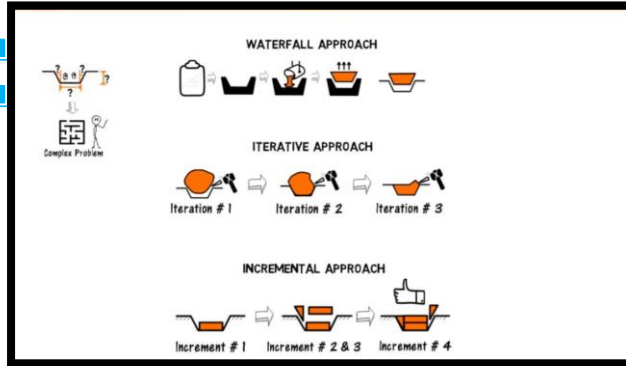
- Un *Framework léger*
- Un ensemble des **règles** pour **guider les relations et les interactions** entre les individus participant au projet
- **Volontairement incomplet** pour laisser la liberté à **l'intelligence collective** (capacité à résoudre des problématiques en travaillant ensemble avec des profils divers)
- Un *Framework Agile*
- Axé sur la **valeur du produit** et sa **maximisation**
- **Adapté aux solutions innovantes et complexes**
- Une aide pour les équipes à générer rapidement de la valeur aux produits et services à développer
- Utilisé pour des projets dans divers domaines (n'est plus exclusivement réservé à l'IT)

# THÉORIE : EMPIRISME ET LEAN

- La théorie de *SCRUM* se base sur l'**Empirisme** et la pensée *LEAN*
- Empirisme
  - Courant philosophique qui dit que toutes connaissances viennent de l'expérience
  - Appliqué dans le cadre SCRUM, cela signifie que les « *décisions à prendre doivent s'appuyer sur l'observation de faits* »
- Pensée du *LEAN*
  - Réduction du gaspillage, concentration sur l'essentiel et discipline
    - Meilleure performance
    - Meilleure rentabilité
    - Meilleure qualité

# APPROCHE ITÉRATIVE ET INCRÉMENTALE

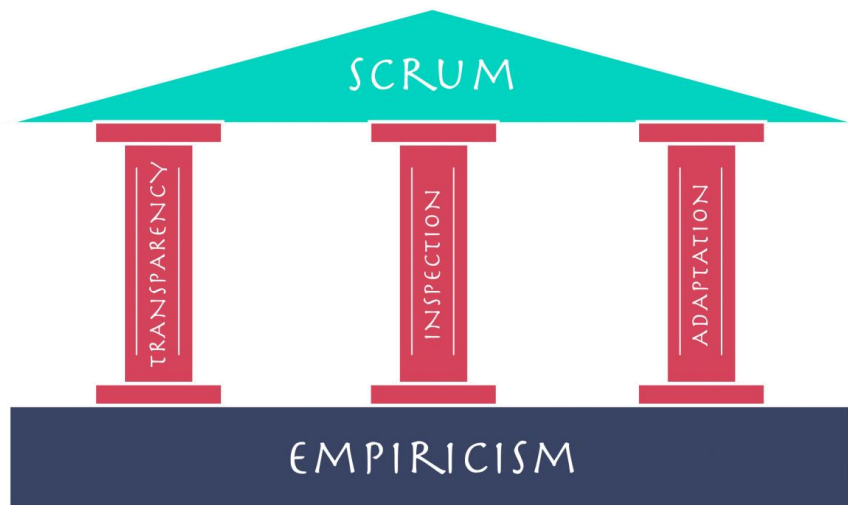
Sou  
Sou



- « *Approche à la fois itérative et incrémentale pour optimiser la prédictibilité et le contrôle de risque.* »
- Les équipes SCRUM ont toutes les **compétences et expertises** pour travailler sur le projet
- Les membres de l'équipe **partagent** leurs compétences et expertises au sein de l'équipe

# LES 3 PILLIERS DE SCRUM

[Source image wikiagile](#)



- **Transparence**
  - Tout ce qui concerne le produit doit être **visible par toutes les personnes impliquées**
  - Peu de transparence augmente le risque et diminue la valeur du produit
  - La transparence permet l'inspection
- **Inspection**
  - Inspection fréquente du travail en cours pour **détecter rapidement les écarts**
  - L'inspection permet l'adaptation
- **Adaptation**
  - **S'ajuster** dès que les écarts deviennent inacceptables pour atteindre l'objectif fixé

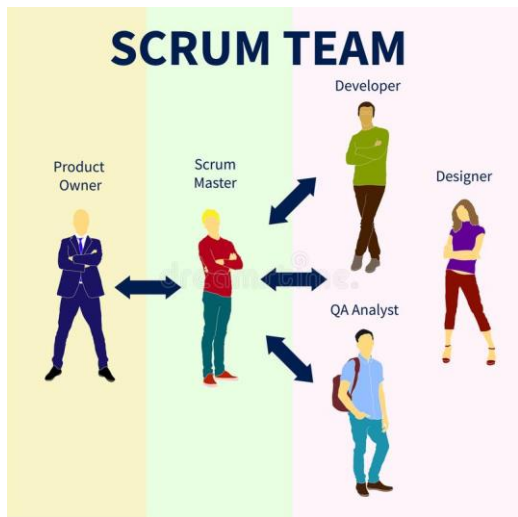
# L'ÉQUIPE SCRUM

---



# L'ÉQUIPE SCRUM

Source image dreamstime



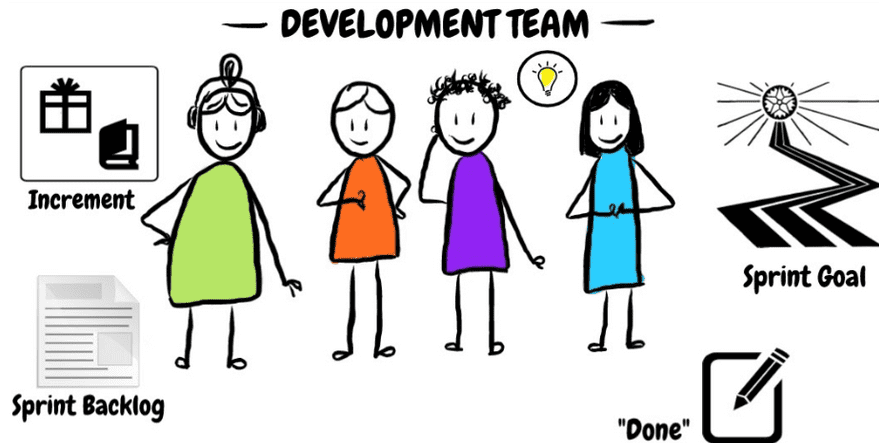
- 1 Scrum Master (SM)
  - 1 Product Owner (PO)
  - Developers (3 à 9 individus)
- 
- Pas de hiérarchie
  - Pas de sous-équipe
  - Pas d'autres rôles que les 3 ci-dessus
- 
- Dans le cas de plusieurs équipes SCRUM travaillant sur le même produit ou service
    - 1 PO
    - 1 SM par équipe

# COMPÉTENCES ET EXPERTISE DE L'ÉQUIPE

- Les équipes SCRUM sont
  - **Cross-functional** (multifonctionnel ou transversal)
    - Les membres ont toutes les compétences requises pour produire de la valeur à chaque itération de la réalisation du produit
  - **Self-managing** (auto-gestion)
    - Les membres décident de qui fait quoi, quand et comment
- Les équipes sont **soutenues par leur direction** dans leurs prises de décision
- **Tout le monde est responsable de la création de valeur.**
- Les équipes plus petites ont tendance à mieux s'organiser et à être plus productif (c'est une généralité et non une vérité absolue)



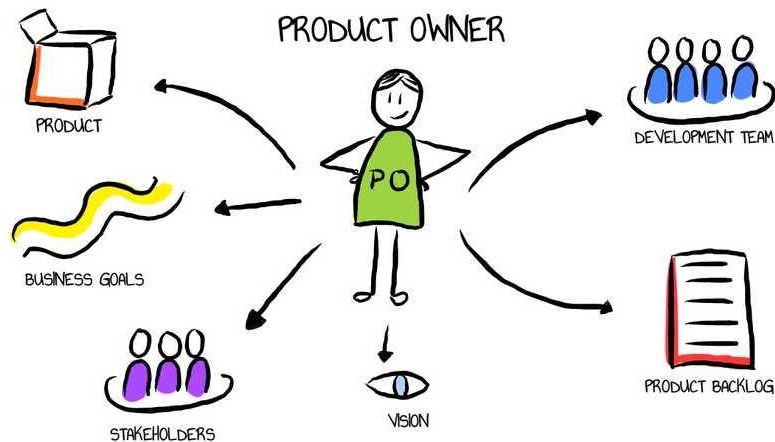
[Source image Letsscrumit](#)



- Le rôle *developers* regroupent toutes les personnes responsables de la création d'au moins un incrément de valeur à chaque sprint. (équipe de réalisation)

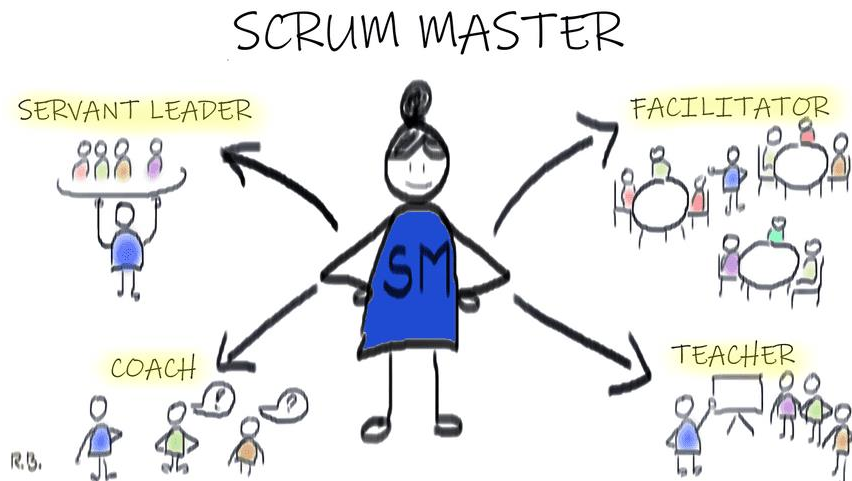
# PRODUCT OWNER

Source image Letsscrumit



- Une seule personne
- Représente les autres parties prenantes (client, utilisateurs finaux, etc.) au sein de l'équipe
- Responsabilités
  - Maximiser la valeur du produit
  - Gérer efficacement le *Product Backlog (PB)*
    - Développer et communiquer le *Product Goal*
    - Créer, communiquer et prioriser les *Product Backlog Items*
    - Rendre le PB transparent, visible et compréhensible

Source image Letsscrumit



- S'assurer de la **bonne application de SCRUM**
  - Tous les événements ont lieu avec le respect du temps (*timebox*) et que les membres concernés y participent
- Aide l'équipe et l'organisation à comprendre SCRUM
  - Efficacité de l'équipe
  - Aider à l'amélioration des pratiques
- Au service de l'équipe
  - En tant que **facilitateur**
  - Celui qui **lève les obstacles**
  - **Coache** l'équipe dans l'autogestion et partage des compétences
  - Aide l'équipe à se concentrer sur la maximisation de la valeur

## Collaboration avec le Product Owner

- Aide à trouver des techniques plus efficaces pour communiquer le *Product Goal*
- Aide à gérer efficacement le *Product Backlog*
- Aide à comprendre et à décrire clairement les *Product Backlog Items*
- Facilite la collaboration avec les parties prenantes du projet

## Collaboration avec l'organisation (entreprise)

- Guide, entraîne et coach l'organisation à adopter *SCRUM*
- Planifier et conseil l'organisation à mettre en place *SCRUM*
- Supprime les barrières entre les parties prenantes et l'équipe *SCRUM*

## EXERCICE 5

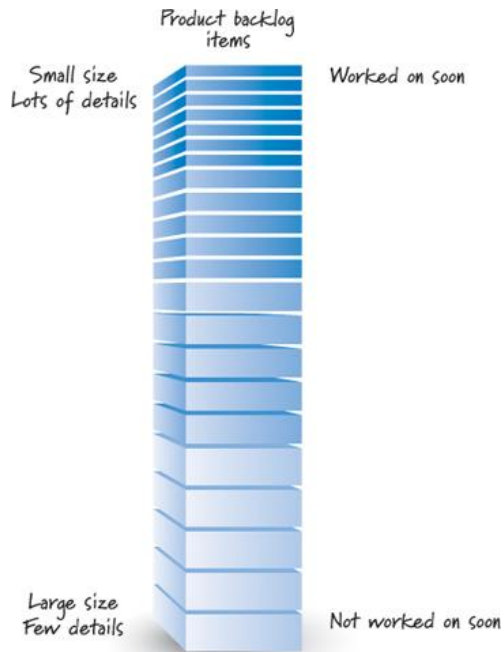
0-exercices/ex5.md

# PRODUCT BACKLOG

---



## Source image Informat



## Liste émergente et ordonnée

- Ensemble de tous les éléments, exigences, améliorations, corrections visant à **construire** ou à **améliorer un produit**
- *Product Backlog* contient
  - Besoins fonctionnels
  - Besoins non-fonctionnels (NFRs)
  - Bugs
  - Etc.
- Les éléments constituant le *Product Backlog* peuvent être déclinés en
  - *Epic* (macro fonctionnalité)
  - *User Story* (fonctionnalité)
  - Tâche (plus petite unité de travail)
  - Etc.

# PRODUCT BACKLOG ITEMS

Source image Cybermedian



Les éléments du Product Backlog doivent idéalement être

- Suffisamment détaillés et transparents
- Émergents
- Priorisés

Classés généralement dans cet ordre

- Ceux qui ont le plus de valeurs et/ou les plus affinés sont en haut de la pile
- Ceux qui ont les moins de valeurs et non détaillé en bas de la pile
- Estimé uniquement par les *Developers*
- Techniques d'estimation
  - *Extreme quotation*
  - *Planning Poker*





# **EXERCICE 6 : ATELIER BRAINSTORMING PRÉPARATION CRÉATION PRODUCT BACKLOG**

0-exercices/ex6.md

# MÉTHODES POUR CONSTRUIRE UN PRODUCT BACKLOG

---



- Une modélisation des utilisateurs finaux à partir des données issues des recherches, entretiens ou du marketing.
  - Un *persona* est un personnage **fictif** qui a les caractéristiques d'une personne réelle dont le but est de représenter un segment cible de sa clientèle.
- 
- On se sert des *personas* pour :
    - Connaître les **attentes, objectifs, motivations, contraintes, comportements, habitudes de consommation**, etc. d'un groupe d'utilisateur ;
    - Émettre, valider ou infirmer des hypothèses sur un groupe d'utilisateur.
    - Créer des prototypes testables

## COMMENT RÉALISER DES PERSONAS ?

- Mener des entretiens sur un panel d'utilisateurs pour identifier des caractéristiques communes pouvant faire la modélisation d'un persona
- Durant ses enquêtes, observer et identifier les objectifs et motivations des utilisateurs dans le contexte du produit ou application.
- Idéalement
  - Ne pas dépasser 10 *personas*
  - Prioriser les *personas*
    - 1 primaire (principale)
    - Les autres secondaires

# EXEMPLES

[sources images mailjet.com](#)



## GÉRALDINE DESLOGES

### PROFIL

Femme  
41 ans  
Professeur de français  
Habite dans le centre de Lyon  
Mariée  
2 enfants (1 fille de 13 ans  
et 1 fils de 8 ans)

### HISTOIRE

Passionnée de littérature depuis le lycée, Géraldine a toujours su qu'elle voulait devenir professeur de français. Elle n'a pas quitté la ville de Lyon dans laquelle elle a grandi. Elle s'est mariée à 24 ans et a eu 2 enfants avec son mari. Ils habitent dans un grand appartement dans le centre ville.

### PERSONNALITÉ

Géraldine est une personne passionnée, aimante et déterminée. Elle aime le fait de pouvoir passer du temps avec ses enfants. Lorsqu'elle a du temps libre, elle adore lire et créer des bijoux qu'elle offre aux personnes de son entourage. Elle aimerait habiter dans une maison avec jardin plutôt que dans un appartement afin de pouvoir entretenir un potager. Sensible à l'écologie, elle fait attention à adopter des comportements et à réaliser des achats respectueux de l'environnement. Elle est coquette mais n'aime pas passer trop de temps à se préparer.

### OBJECTIFS

- Souhaite utiliser des cosmétiques naturels et biologiques
- Privilégie les marques responsables

### FREINS

- Ne souhaite pas dépenser trop d'argent pour des produits cosmétiques
- N'aime pas passer trop de temps à chercher des produits cosmétiques

### SOURCES D'INFORMATIONS

- Préfère se renseigner dans les magazines féminins qu'elle lit plutôt que sur internet
- Préfère le bouche-à-oreille car fait confiance à l'expérience de ses amies



## PAUL MELLER

Homme  
29 ans  
Célibataire  
Pas d'enfant  
Habite à Paris  
Fondateur de start-up

### HISTOIRE

Originaire de Nantes, Paul a passé un bac ES, puis a suivi une école de commerce. Ayant obtenu un master 2 dans l'entrepreneuriat, Paul a commencé à travailler pour une startup en tant que Rédacteur Web. Après 2 ans au sein de la société, Paul a souhaité lancer une start-up spécialisée dans la rédaction de contenus autour des voyages et de la nature.

### PERSONNALITÉ

Passionné de photographie et amoureux de la nature, Paul a déjà réalisé de nombreux voyages où il y a pu réaliser de nombreux clichés de paysages. Il poste ses meilleures photos sur son profil Instagram qu'il accompagne d'un texte pour préciser le contexte dans lequel il les a prises et pour sensibiliser sur l'impact de l'homme sur la nature. Perfectionniste et ambitieux, il est très pris par sa vie professionnelle à laquelle il accorde la plus grande partie de son temps.

### OBJECTIFS

- Base ses décisions sur des faits chiffrés et éprouvés
- Souhait développer son business le plus rapidement possible

### FREINS

- Fortes contraintes budgétaires
- N'a que très peu de temps à accorder pour les réunions

### SOURCES D'INFOS

- Groupes dédiés aux start-ups sur les réseaux sociaux
- Médias en ligne à destination des entrepreneurs



# EXERCICE 7 : CRÉATION DES PERSONAS

0-exercices/ex7.md

# USER STORY

# USER STORY (RÉCIT UTILISATEUR)

[Source image Justinmind](#)

## User Story

**As a** <role>  
**I want** <goal>  
**so that** <benefit>

**Acceptance criteria:**  
(Conditions of Satisfaction)

...  
...

**As an** Account Manager  
**I want** a sales report of my account  
to be sent to my inbox daily  
**So that** I can monitor the sales  
progress of my customer portfolio

**Acceptance criteria:**

1. The report is sent daily to my inbox
2. The report contains the following sales details: ...
3. The report is in csv format.

- Les entretiens avec les utilisateurs finaux et le client permettront d'avoir une vision plus précise des besoins
- Découpage le plus petit possible, réalisable en **1 Sprint**
- Un *User story* permet de découper le besoin en répondant à 4 questions
  - **Qui ?**
    - Les utilisateurs
    - Formulation : *en tant que ...*
  - **Quoi ?**
    - Besoin profond
    - Formulation : *je veux, je souhaite, etc...*
  - **Pourquoi ?**
    - La valeur
    - Formulation : *afin de, dans le but de ...*
  - Quels sont les critères d'acceptation
    - Liste d'éléments pour vérifier que le besoin est satisfait



# **Démo découpage en user stories avec un persona**

---

# DÉCOUPAGE D'UNE USER STORY AVEC LA GRILLE INVEST

## Grille *INVEST*

- ***Independant***: techniquement indépendante des autres
- ***Negociable***: choix de la solution (autrement dit les devs ont la possibilité de choisir une solution)
- ***Valuable***: apporte de la valeur à l'utilisateur
- ***Estimable***: estimable à l'aide d'un outil ou méthode d'estimation
- ***Small***: suffisamment petit pour être réalisé lors d'un *Sprint*
- ***Testable***: valider les critères d'acceptation et tester avec les tests automatisés



## **EXERCICE 8 : CONSTRUCTION DU PRODUCT BACKLOG AVEC LES USER STORIES**

0-exercices/ex8.md

# ESTIMATION DES USER STORIES

---



# ESTIMATION AVEC LE PLANNING POKER

Source image Zentao

Agile Estimation Technique

Planning Poker



- Estimer la complexité d'un *US*, consiste à mesurer (curse)
  - *La difficulté*
  - *L'incertitude*
  - *Risque*
  - *Périmètre*
  - *Effort*
- **Il ne s'agit pas de déterminer le temps de réalisation de l'*US***
- L'estimation doit être faite uniquement les *Developers*
- *PO* et *SM* peuvent estimer uniquement s'ils participent activement à la réalisation d'un élément du *Sprint Backlog*
- Pour les valeurs, utilisation la suite de *Fibonacci* 1, 2, 3, 5, 8, 13, 21 etc.
- *Storypoint* = complexité attribuée à un élément du *Product Backlog*



## EXERCICE 9 : ESTIMATIONS AVEC LE PLANNING POKER

0-exercices/ex9.md

# LES ÉVÉNEMENTS SCRUM

---



# SPRINT PLANNING

- Événement d'ouverture du Sprint
- L'équipe peut inviter des personnes extérieures pour avoir des conseils et leurs expertises
- Le Sprint Planning doit répondre à 3 questions :
  1. *Pourquoi ce Sprint est-il important ?*
    1. Le *Product Owner* expose la valeur qu'elle apporte au produit
    2. L'équipe finalise l'objectif (*Sprint Goal*) du *Sprint* avant la fin de la réunion
  2. *Que peut-on faire durant ce Sprint ?*
    1. Les *Developers* sélectionnent les éléments du *Product Backlog* sur lesquels ils vont travailler pour atteindre l'objectif
    2. Ils affinent et s'engagent à les réaliser durant le *Sprint*. Pour cela, ils s'appuient sur les performances passées, leur projection sur les capacités à venir et la *Definition of Done*.
  3. *Comment le travail choisi sera -t-il réalisé ?*
    1. Pour chaque élément du *Product Backlog* sélectionné, les *Developers* définissent un plan d'action qui répond à la *Definition of Done*
    2. Les *Developers* sont les seuls à décider de la manière de travailler sur un *Sprint*.
- *Sprint Backlog = Sprint Goal + Product Backlog items sélectionnés + Plan de livraison*



- Événement **conteneur**
  - Inclus tous les autres événements
- Développer un incrément opérationnel (livrable utilisable)
  - Une brique supplémentaire qui vient s'ajouter à l'existant
  - Un incrément qui nous rapproche davantage du produit final
- Les sprints plus courts limitent les risques liés aux coûts et à l'effort, car le *feedback* est plus rapide et donc l'adaptation est faite au plus tôt.
- **Attention, le Sprint 0 n'existe pas dans la théorie de SCRUM**, c'est une mauvaise pratique qu'on retrouve dans les entreprises qui consistent à ne pas produire de la valeur au premier sprint, à la place à mettre en place l'architecture, la base de données, l'organisation de l'équipe, etc.
- Le *Sprint* peut être annulé uniquement par le *Product Owner* si l'objectif du *Sprint* devient obsolète.
- On ne présente pas et par extension, on ne livre pas un incrément non fini.  
Les éléments non finis du *Sprint Backlog* retournent dans le *Product Backlog*

# DAILY SCRUM

- Objectifs
  - **Inspecter** les travaux en cours vers l'objectif de *Sprint* et **adapter** le *Sprint Backlog* si nécessaire
  - Améliorer la communication entre les *Developers*
  - Identifier les obstacles et les lever
  - Éliminer la nécessité d'avoir recours à d'autres réunions (réduire ne veut pas dire supprimer entièrement cette possibilité)
- Se tient **tous les jours ouvrés** à la **même heure** et **au même endroit** pour réduire la complexité
  - La durée maximum de l'événement est de 15 min
- Participation
  - Obligatoires pour les *Developers*
  - Le *Scrum Master* et *Product Owner* peuvent participer uniquement dans le cas où ils travailleraient activement sur un élément du *Sprint Backlog*. Ils participent alors dans ce cas avec le rôle de *Developers*



## EXERCICE 10 : SPRINT PLANNING

0-exercices/ex10.md

# SPRINT REVIEW

[Source image letsscrumit](#)



- Session de travail à part entière (ne se limite pas à une présentation du travail réalisé durant le *Sprint*)
- Objectifs
  - **Inspecter** le travail réalisé durant le *Sprint*
  - Présenter ce travail aux parties prenantes du projet et la progression vers l'objectif du produit
  - **Récolter du *feedback***
  - **Adapter** la direction à prendre pour les prochains *Sprints*

# SPRINT RETROSPECTIVE

- Objectifs
  - Réflexion sur les **pistes pour améliorer la qualité et l'efficacité de l'équipe** pour les prochains *Sprints*
  - **Inspecter** le déroulement du *Sprint* en cours au niveau
    - Des individus
    - Interactions
    - Processus
    - Outils
    - *Definition Of Done (DoD)*
  - Identifier les éléments qui ont fait dévier l'équipe de son plan initial et leurs origines
  - Identifier les obstacles, problèmes et comment ils ont été résolus ou non
- Les éléments d'amélioration les plus impactant peuvent être **ajoutés** au *Product BackLog*
- L'équipe peut décider durant cet événement de changer le *DoD*
- Événement qui conclut le *Sprint*

# TIME-BOXING DE TOUS LES ÉVÉNEMENTS SCRUM

- *Sprint*
    - 1 mois au maximum (entre 1 et 4 semaines)
  - *Sprint Planning*
    - Pour un *Sprint* d'un mois, c'est 8 heures maximum
    - Pour un *Sprint* de moins d'un mois, c'est moins de 8 heures
  - *Daily Scrum*
    - 15 minutes maximum
- 
- *Sprint Review*
    - Pour un *Sprint* d'un mois, c'est 4 heures maximum
    - Pour un *Sprint* de moins d'un mois, c'est moins de 4 heures
  - *Sprint Retrospective*
    - Pour un *Sprint* d'un mois, c'est 3 heures maximum
    - Pour un *Sprint* de moins d'un mois, c'est moins de 3 heures



# EXERCICE 11 : REVIEW ET RETROSPECTIVE

0-exercices/ex11.md

# ANNEXE

---





- Manifeste Agile
- Principes agiles

**Merci de votre attention et participation  
Glodie Tshimini**

---

