

Artcoded

# La méthode Agile

Description et fonctionnement de la méthode Agile et ses dérivés

Soultan Hatsijev  
03/05/2022

## Table des matières

Introduction.....	2
Fonctionnement .....	2
<i>Backlog</i> .....	2
<i>Story Points</i> .....	2
Sprint.....	3
Les différents types de méthode Agile .....	5
Scrum .....	5
Kanban .....	5
Les différences .....	6
GitHub .....	8
Références.....	10

## Introduction

Agile est une méthode de gestion de projet qui contrairement à Waterfall (en cascade) qui est l'autre méthode la plus répandue ne demande pas aux collaborateurs de connaître les étapes de développement entièrement en avance.

En effet, la méthode Agile permet aux équipes de développer un logiciel par tâche définies à l'avance, cette méthode est aussi plus flexible étant donné qu'elle permet de rajouter en cours de route de nouvelles étapes tel qu'ajouter une nouvelle fonctionnalité dans une tâche déjà terminée.

Les résultats sont évalués continuellement, cette méthode permet de répondre rapidement aux changements.

## Fonctionnement

### *Backlog*

Lors de la prise d'un projet, tout le monde se réunit (développeurs, chef de projet, business analyste, etc.) lors de ce qu'on appelle un « *Backlog<sup>1</sup> Refinement Initial<sup>2</sup>* » pour discuter du découpage du produit en différentes fonctionnalités nécessaires au logiciel appelées « *User Stories<sup>3</sup>* » qui seront toutes contenues dans le « *Backlog* », cette méthode est principalement utilisée avec Scrum décrits plus tard.

### *Story Points*

Lorsque chacun des récits aura été défini, l'équipe se rassemble pour attribuer des « *Story Points<sup>4</sup>* » qui représenteront la difficulté estimée de la tâche pour chacun des développeurs lors du « *Backlog Refinement Initial* » ou « *Product Backlog Refinement* », cette dernière réunion est similaire à la précédente à la différence qu'elle est effectuée « en cours de route », donc pendant le projet, pour réévaluer les tâches.

Ces points ne sont pas représentatif du temps mais bien de la difficulté ! Les chiffres utilisés ne sont pas les mêmes partout cependant les plus communs sont ceux de la suite de Fibonacci ainsi « 1 » représente une tâche jugée facile et « 100 » une tâche jugée très compliquée.

Lors de l'évaluation des tâches, chacun des membres seront interrogés sur leur choix, « Pourquoi avoir attribué X à cette tâche », les membres devront donc se justifier et faire cela est important lors de l'élaboration ainsi les membres peuvent repenser leur choix ou le chef de projet pourrait évaluer le niveau des membres, il pourrait aussi décider de simplifier le travail en le découpant encore une fois, etc.

Lorsque les « *SP<sup>5</sup>* » d'une tâche sont élevés, on appelle ça une « *Epic<sup>6</sup>* » cela représente souvent des grosses tâches comme développer l'interface graphique de l'application mobile. Cette dernière pourrait être découpée en sous-tâches tel que

- Développer la page d'accueil
- Développer la page produit
- Mettre en place une page de chargement

---

<sup>1</sup> Carnet de produit en français

<sup>2</sup> Raffinement du *backlog* initial en français mais représente la mise en forme et la constitution du *backlog*

<sup>3</sup> Récit utilisateur en français

<sup>4</sup> Points de récit en français

<sup>5</sup> Story Points

<sup>6</sup> Epique en français

- S'occuper des effets sonores
- ...

Une fois l'avis des membres de l'équipe récoltés, le « *Scrum master*<sup>7</sup> » s'assurera des modifications nécessaires au bon fonctionnement du système notamment en assignant des duos de membre appelé également *Pair Programming*<sup>8</sup>, le duo est composé d'un membre qui juge une tâche difficile et un autre membre qui la juge facile, ceci permet non seulement de renforcer la cohésion d'équipe mais aussi au développeur le moins expérimenté d'apprendre du plus expérimenté.

Le « *Scrum master* » se chargera aussi d'attribuer un « SP » médian par exemple si une partie de l'équipe déclare une tâche avec un « SP » de « 5 » et une autre partie de l'équipe avec « 13 », le « *Scrum master* » s'occupera d'attribuer un « 8 ».

Toute cette dernière partie est réalisée lors d'un « Planning Poker » où un « *User Story* » est choisi et les membres évaluent la difficulté.

## Sprint

Les Sprint dans Agile représente la liste de toutes les tâches à accomplir dans un certain délai, celui-ci n'est pas fixe à une semaine ou autre, il est défini par le *Scrum Master*.

Durant le Sprint, il survient des événements explicités par une terminologie propre à Agile utilisée pour faciliter la compréhension et la flexibilité des utilisateurs pouvant passer d'une entreprise à une autre, les termes étant toujours les mêmes. Dans la majeure partie des cas, les événements ne sont rien d'autre que des réunions pour parler de l'avancement du projet ou pour discuter de changement ou pour une autre raison.

## Sprint planning

Une fois que le découpage est terminé et l'évaluation de la difficulté aussi, il est temps de se lancer dans le développement, pas vraiment !

Chaque début de Sprint, une réunion appelée « *Scrum meeting*<sup>9</sup> » ou « *Sprint Planning*<sup>10</sup> » est organisée avec le « *Scrum master* » pour définir les tâches et évaluer les « SP » total nécessaires au développement des fonctionnalités

Les « US<sup>11</sup> » sont choisis pour atteindre un nombre de « SP » maximum pour le sprint, si à la fin du sprint l'équipe réalise plus que prévu, le sprint suivant demandera un plus grand nombre de « SP », évidemment un bon « *Scrum master* » ne reprendra pas le nombre effectué mais une valeur entre ce dernier et la valeur normalement attendue.

## Sprint review / demo

A la fin du Sprint, l'équipe organise un Sprint Review où tout le monde est invité des clients au sponsor. Durant cet événement, l'équipe présente les fonctionnalités développées.

---

<sup>7</sup> Le leader de Scrum, il s'occupe d'organiser les réunions, supprimer les obstacles et garder le Backlog constamment à jour

<sup>8</sup> Programmation en pair en français

<sup>9</sup> Réunion Scrum en français

<sup>10</sup> Planification du sprint en français

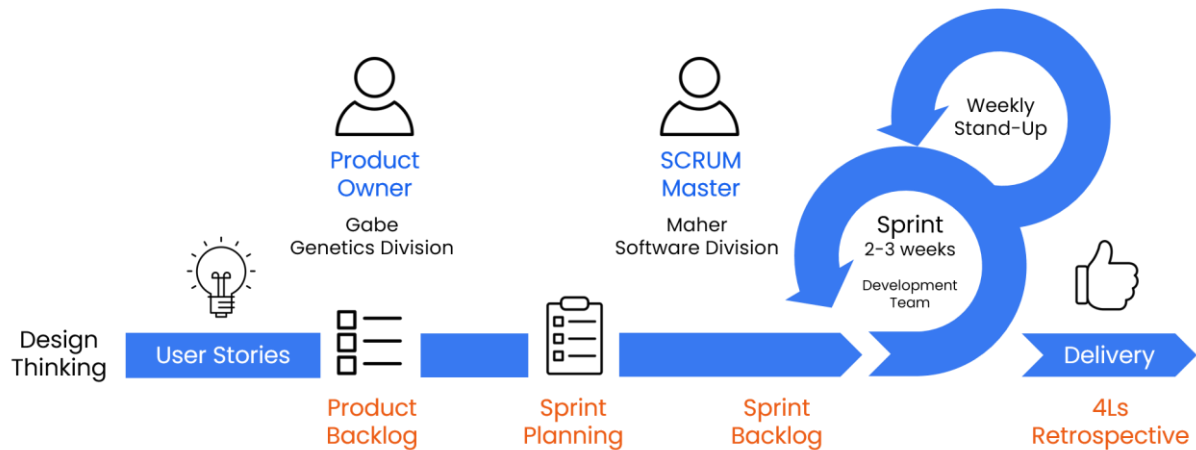
<sup>11</sup> *User Stories*

## Sprint retrospective

Une fois la *review* terminée, l'équipe se rassemble une fois de plus, cette fois-ci sans les invités pour évaluer le bon déroulement de l'opération.

C'est lors de cette évènement que l'équipe mettra en avant les points positifs et négatifs, ce qui a le mieux marché et le moins bien marché. Ces points pourraient être tel que « La collaboration avec l'équipe était très bonne » ou « L'accès à l'intranet n'est toujours pas disponible ».

De cette manière, l'équipe prend des engagements pour le sprint suivant par exemple « Envoyer un mail demandant l'accès à l'intranet » et lors de la rétrospection suivante, l'équipe vérifie si les problèmes précédents ont été réglés et si de nouveaux problèmes sont apparus.



[Cette photo](#) par Auteur inconnu est soumise à la licence [CC BY](#)

Figure 1 Durée de vie d'un projet Agile (Scrum)

## Standup

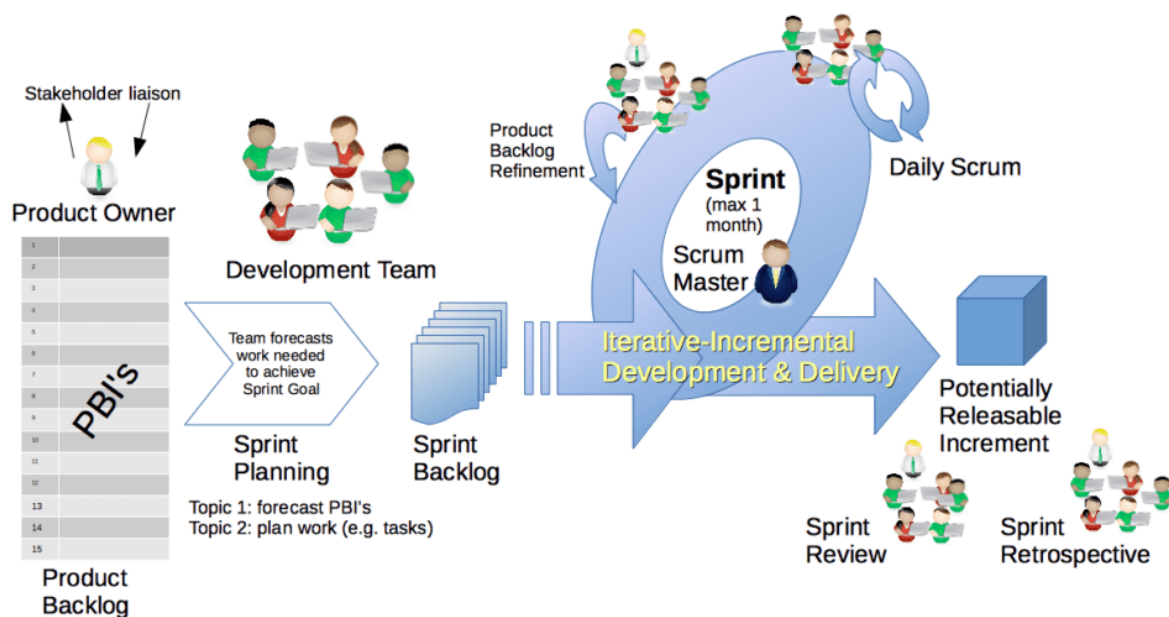
Enfin, il existe encore une réunion, assez courte appelée « *Standup* », elle est organisée soit de manière journalière ou hebdomadaire (le plus répandu) pour évaluer les tâches et avoir un suivi constant avec les développeurs.

## Les différents types de méthode Agile

### Scrum

Scrum pourrait être défini comme étant un Framework d'Agile, il encourage les membres à apprendre par l'expérience, ceci permet d'identifier ses points faibles et ses points forts tout au long ou à la fin d'un projet.

Les méthodes décrites dans le fonctionnement d'Agile sont celles de Scrum.



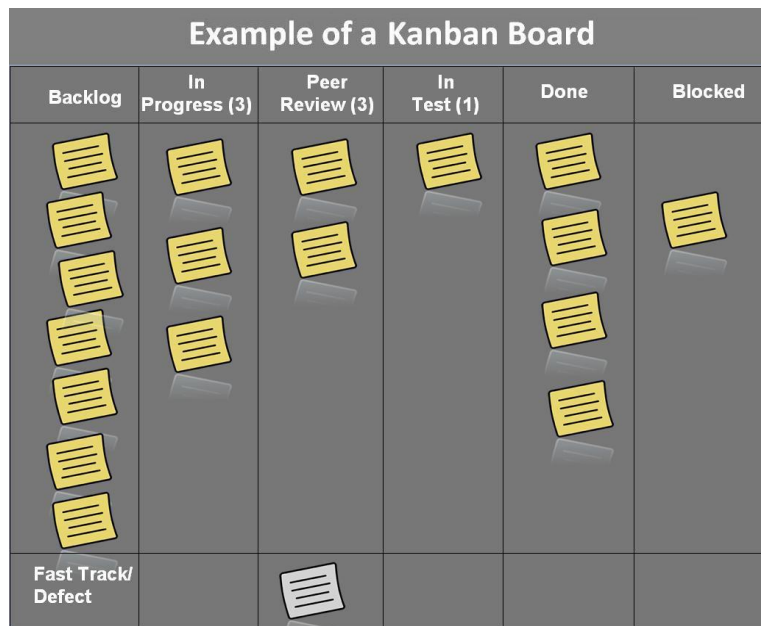
[Cette photo](#) par Auteur inconnu est soumise à la licence [CC BY-NC-ND](#)

Figure 2 Durée de vie d'un projet Scrum

### Kanban

Kanban est un autre Framework d'Agile, beaucoup plus visuel, en effet celui-ci est basé sur un tableau reprenant ce qu'on a pu voir comme le « Backlog », les tâches à réaliser (le « Sprint »), les tâches en cours, les tâches en *review*<sup>12</sup>, les tâches en test et enfin les tâches terminées. Des colonnes supplémentaires peuvent être utilisées dépendant du contexte.

<sup>12</sup> En revue ou en examen en français. Une tâche en *review* est une tâche examinée avant son « acceptation »



Cette photo par Auteur inconnu est soumise à la licence [CC BY-SA](#)

Figure 3 Durée de vie d'un projet Kanban

## Les différences

Malgré leurs ressemblances, Scrum et Kanban ont tout de même quelques différences tel que :

- La manière d'organiser
- Les rôles des membres
- Le rythme
- Le planning
- Les promesses au client
- Les réunions
- ...

Ces différences ne peuvent être mieux expliquées que par la figure suivante :



Figure 4 Différences entre Scrum et Kanban



## Mais encore ?

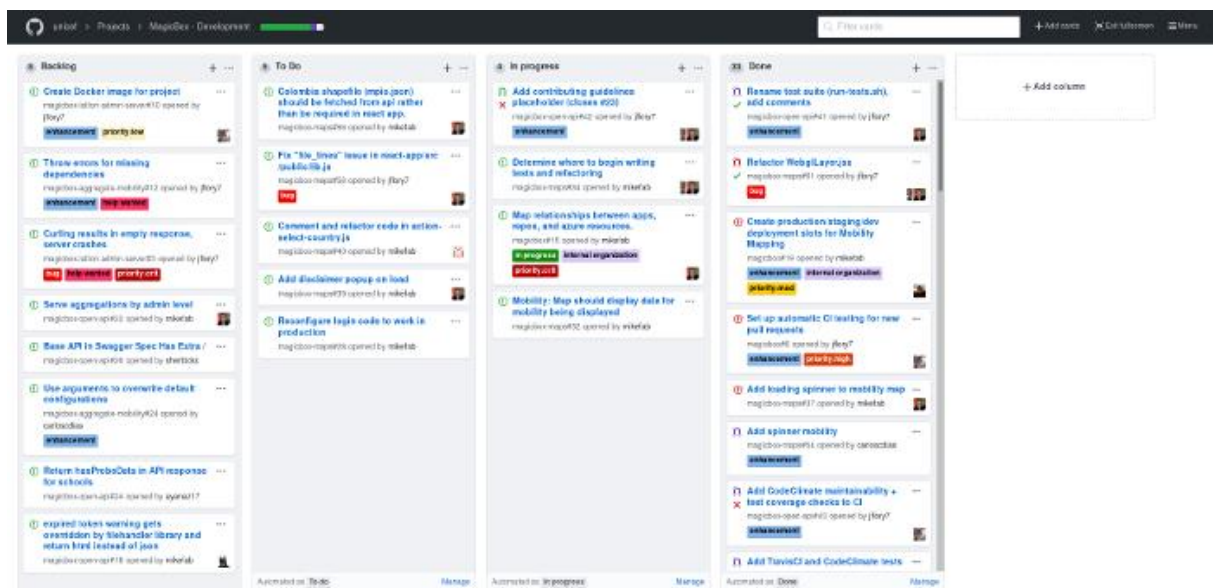
L'avantage d'Agile reste son utilisation très flexible, nous avons rapidement passé en revue les termes, les Framework et leurs fonctionnements qui peuvent paraître définit et immuable pourtant chaque entreprise peut adapter Agile à son environnement.

C'est notamment le cas en région flamande où un nouveau terme est apparu, Spurt.

Ce terme qui ne représente ni plus ni moins que 5 Sprint d'un délai total de 10 semaines.

## GitHub

Lors du développement d'un projet en utilisant GitHub, un mix des deux peut être utilisé, GitHub mettant à disposition un tableau modulable, c'est-à-dire un tableau dans lequel des colonnes peuvent être ajoutées, supprimées, modifiées.



[Cette photo](#) par Auteur inconnu est soumise à la licence [CC BY-SA](#)

Figure 5 Gestion de projet sur GitHub

GitHub étant orienté pour le développement informatique, il est possible d'assigner une tâche comme étant une « *Issue*<sup>13</sup> », lorsque qu'elle sera terminée, le membre fera un « *push* » de ses nouvelles fonctionnalités qui seront automatiquement déplacée dans une colonne en *review* par exemple.

<sup>13</sup> Un problème en français

## Figures

Figure 1 Durée de vie d'un projet Agile (Scrum).....	4
Figure 2 Durée de vie d'un projet Scrum .....	5
Figure 3 Durée de vie d'un projet Kanban .....	6
Figure 4 Différences entre Scrum et Kanban .....	7
Figure 5 Gestion de projet sur GitHub .....	8

## Références

- Adobe. (s.d.). *Waterfall Methodology*. Récupéré sur Workfront: <https://www.workfront.com/project-management/methodologies/waterfall>
- Atlassian. (s.d.). *Présentation d'Agile*. Récupéré sur Atlassian Agile Coach: <https://www.atlassian.com/fr/agile#:~:text=Agile%20est%20une%20approche%20it%C3%A9rative,travail%20par%20petits%20incr%C3%A9ments%20exploitables>.
- Atlassian. (s.d.). *Qu'est-ce que Kanban ?* Récupéré sur Atlassian Agile Coach: <https://www.atlassian.com/fr/agile/kanban>
- Atlassian. (s.d.). *Qu'est-ce que Scrum ?* Récupéré sur Atlassian Agile Coach: <https://www.atlassian.com/fr/agile/scrum>
- Atlassian. (s.d.). *Story points et estimation*. Récupéré sur Atlassian Agile Coach: <https://www.atlassian.com/fr/agile/project-management/estimation>
- Bit. (2021, 09 14). *What is agile methodology? How it actually works*. Récupéré sur Bit Business IT: <https://www.bit.com.au/guide/what-is-agile-how-it-actually-works-457412>
- Chappell, E. (2020, 09 10). *50 Agile Terms Everyone Must Know (Detailed Glossary)*. Récupéré sur ClickUp blog: <https://clickup.com/blog/agile-terms/>
- Grosjean, J. C. (s.d.). *Quand estimer les User Stories?* Récupéré sur Quality Street: <https://www.qualitystreet.fr/2019/11/29/quand-estimer-les-user-stories/>
- Paquet, J. (2020, 06 17). *Backlog traduction*. Récupéré sur MyAgilePartner: <https://blog.myagilepartner.fr/index.php/2020/06/17/backlog-traduction/>
- Penverne, A. (s.d.). *La différence entre SCRUM et KANBAN*. Récupéré sur Blog Gestion de Projet: <https://blog-gestion-de-projet.com/la-difference-entre-scrum-et-kanban/>
- Schwaber, K., & Sutherland, J. (2020, Novembre). *The Scrum Guide*. Récupéré sur Scrum guides: <https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-US.pdf>
- Simplilearn. (2022, 04 13). *What is Story Points in Agile And How to Estimate Them*. Récupéré sur Simplilearn: <https://www.simplilearn.com/story-points-in-agile-article#:~:text=Story%20points%20in%20Agile%20are,and%205%2C000%2C000%20if%20you%20want>.