

Les événements Agile vus par le développeur

Table des matières

I. Introduction	3
II. Daily meeting	3
III. Sprint planning	3
IV. Backlog refinement	4
V. Sprint retrospective	4
VI. Sprint review	5
VII. Auto-évaluation	5
Solutions des exercices	6

I. Introduction

Qu'attend-on d'un développeur dans les événements Scrum ?

Dans ce cours, nous allons revenir sur les principaux événements Agile afin de comprendre la place et le rôle qu'occupe un développeur dans chacun d'entre eux.

II. Daily meeting

Le développeur et le Daily Meeting

Le développeur doit préparer le daily meeting même s'il peut être aidé par le Scrum master.

Il organise sa journée sur la base des décisions prises lors du précédent point en s'alignant sur la gestion de configuration, il développe les développements stabilisés afin que l'intégration continue profite de la nuit pour lancer les traitements longs. Et le matin avant le daily meeting, il doit prendre connaissance des résultats des traitements de nuit pour préparer la réunion du jour. Sur la base des résultats et de son avancement sur les derniers jours, il réajuste l'estimation de ce qu'il reste à faire et s'efforce de présenter ses difficultés de manière à optimiser l'efficacité de l'échange. Ainsi, il commence son travail avec une vision claire grâce aux feed-backs de ses collègues.

Le daily permet également aux développeurs de savoir sur quels sujets travaillent les autres développeurs de son équipe afin d'éviter que deux personnes travaillent en doublon sur le même sujet sans le savoir.

Attention **À savoir !**

L'idée selon laquelle le daily meeting comporterait uniquement des développeurs, un Scrum Master et un product owner est fautive car les tâches pour atteindre l'objectif donné peuvent nécessiter l'intervention d'un DevOps, d'un business analyst, de l'UX, etc. Ainsi, si d'autres profils doivent se coordonner avec les développeurs, ils doivent être présents.

III. Sprint planning

Le développeur contribue à la planification

Dans le cadre du sprint planning, les développeurs portent collectivement la responsabilité de la solution technique : tous ensemble, ils décident de comment elle doit évoluer et répondent aux besoins prioritaires du product owner en assumant cette responsabilité technique.

De manière pratique, après que le product owner a présenté une story, le développeur doit poser des questions pour qu'elle soit parfaitement claire et que sa réalisation puisse se faire sans difficulté. Les développeurs ont la responsabilité d'identifier collectivement une éventuelle difficulté. Cette identification peut amener à repousser la story au sprint suivant afin que la difficulté soit éliminée au démarrage de ce nouveau sprint.

Sans difficulté identifiée, le développeur prépare son estimation personnelle dans le cadre du planning poker, et les arguments sur lesquels il s'appuie pour la justifier.

Concernant son attitude, il doit faire preuve d'une écoute active quand ses collègues défendent leurs estimations. Il s'efforce aussi d'être structuré et concis quand il s'exprime.

Enfin, si une story ne respecte pas la définition of ready, le développeur s'accorde avec ses collègues pour décider ou non de la prendre malgré tout et, le cas échéant, de préciser les actions nécessaires à son achèvement.

À noter : le développeur n'est pas nécessairement très actif en sprint planning, surtout si le contenu du sprint backlog a été préparé en amont durant le backlog refinement. Le développeur doit se montrer attentif pendant cette réunion.

Formation, partage de connaissances, innovation, etc.

Lors du sprint planning, on doit aussi tenir compte des autres activités, comme celles qui contribuent à la montée en compétence ou sont nécessaires au développement de l'innovation. Il s'agit de décider du temps à y consacrer et de regarder les besoins sur le moyen terme pour les répartir sur les différents sprints. Le développeur doit aussi être vigilant à communiquer ses absences et/ou congés à venir.

Le sprint planning est basé sur la vélocité moyenne de l'équipe. Il peut être envisagé de ne pas planifier le Sprint Backlog sur la totalité de la vélocité afin de garder du temps disponible pour les actions expliquées avant.

IV. Backlog refinement

Préparer les user stories avant le sprint planning

Le backlog refinement vise à valider les stories en amont du sprint planning. Si le travail est bien fait, les difficultés sont éliminées pendant ces points et les stories arrivent en sprint planning en respectant la definition of ready.

Tout d'abord, à la première présentation d'une story, le product owner donne le premier feed-back technique sur son besoin Métier : le besoin peut être mis de côté, car ce feed-back nécessite de le retravailler, il peut sinon être adapté ou conservé en l'état. L'affinage peut se faire sur plusieurs backlog refinements, si nécessaire.

Ensuite, quand le besoin est cadré et validé, on cherche à valider la rédaction de la story : y a-t-il toutes les informations ? Est-ce que tout est clair ? Quelle estimation (à grosse maille) est associée au besoin ?

L'estimation est souvent faite avec l'atelier du poker planning, expliqué plus avant dans la formation. Le développeur doit être actif dans cet événement pour lever tous les doutes sur la story avant de la développer et ainsi pouvoir donner l'estimation qui lui semble la plus juste.

Il n'y a pas de moment défini pour les séances d'affinage des stories : on peut en faire plus ou moins dans un sprint dans le respect des objectifs du sprint en cours. Souvent, on réutilise le créneau du sprint planning sur les semaines suivant la première.

V. Sprint retrospective

La rétrospective : un temps pour s'améliorer

La rétrospective est le moment où on évalue le travail qui a été fait dans le sprint écoulé.

Là aussi, il est préférable que, la veille, le développeur démarre sa réflexion, revienne sur son sprint et sur les problèmes ou les difficultés rencontrées. La première analyse est souvent de manière individuelle, pour présenter quelque chose de plus clair en rétrospective.

Sur les différents événements Scrum, une préparation est fortement recommandée pour être pleinement engagé dans la progression de l'équipe.

Là aussi, l'écoute active est primordiale et il est conseillé de s'appuyer sur des exercices pour redynamiser les échanges. C'est une des responsabilités du Scrum master, mais le développeur peut aussi faire des propositions.

La dernière chose essentielle est de chercher à être constructif en faisant émerger des sujets permettant des améliorations. Cela suppose aussi de se sentir investi dans les actions décidées par l'équipe afin de progresser.

Même si la rétrospective est souvent animée par le Scrum master, celui-ci peut demander à un membre de l'équipe de développement de l'aider à animer une partie de la rétrospective : lire les feed-back de l'équipe, surveiller et rappeler le temps de la réunion (time keeper) ou même, demander à l'équipe de développement de le remplacer. Car, avant tout, ces réunions ont pour but d'aider l'équipe à s'améliorer.

VI. Sprint review

Le développeur en sprint review

En sprint review, les choses sont plus simples : les développeurs exposent le produit en ciblant les nouvelles fonctionnalités, chacun présentant ses développements. Après la présentation, le client pose des questions sur ces nouvelles fonctionnalités. À cette étape, les développeurs s'efforcent de clarifier ce qui est exprimé par le client, de comprendre ce qu'il pense vraiment (ce qu'il aime et n'aime pas) afin de préciser la compréhension du besoin. En Agile et en Scrum particulièrement, on parle de boucle d'apprentissage (feedback loop) pour qualifier ces moments d'échange dans le processus de conception et de développement.

VII. Auto-évaluation

Exercice 1 : Quiz

[solution n°1 p.7]

Question 1

Le développeur est obligé de préparer en avance les événements Scrum.

- ☐ Vrai
- ☐ Faux

Question 2

Qui porte la responsabilité de la solution technique ?

- ☐ Le product owner
- ☐ Le tech lead
- ☐ Le client
- ☐ L'ensemble des développeurs

Question 3

Le daily meeting permet au développeur de recevoir ses consignes du Scrum master pour la journée.

- ☐ Vrai
- ☐ Faux

Question 4

Lors du sprint planning, le développeur écoute les échanges entre le tech lead et le product owner.

- ☐ Vrai
- ☐ Faux

Question 5

Le backlog refinement permet de (plusieurs réponses possibles) :

- ☐ Confronter les visions métier et technique
- ☐ Fournir au product owner un premier retour sur l'impact technique des besoins métier
- ☐ D'attendre sereinement l'estimation en sprint planning
- ☐ Convaincre les développeurs de la nécessité de réaliser les besoins


Solutions des exercices

Exercice p. 5 Solution n°1**Question 1**

Le développeur est obligé de préparer en avance les événements Scrum.

☐ Vrai

☒ Faux

 C'est faux. Le développeur n'est pas obligé de préparer ces événements en avance, mais c'est fortement conseillé.

Question 2


Qui porte la responsabilité de la solution technique ?

☐ Le product owner

☐ Le tech lead

☐ Le client

☒ L'ensemble des développeurs


 S'il paraît préférable qu'une seule personne assume cette responsabilité, comme c'est le cas pour la partie métier, c'est l'ensemble des développeurs qui décident en matière de solution technique. Évidemment le tech lead, ainsi que les personnes confirmées, sont relativement plus écoutés.

Question 3

Le daily meeting permet au développeur de recevoir ses consignes du Scrum master pour la journée.

☐ Vrai

☒ Faux


 Le développeur ne reçoit pas de consigne, encore moins du Scrum master. C'est un échange avec les autres développeurs qui conduit à ajuster son organisation.

Question 4

Lors du sprint planning, le développeur écoute les échanges entre le tech lead et le product owner.

☐ Vrai

☒ Faux

 Le développeur a un rôle très actif dans cet événement.

Question 5

Le backlog refinement permet de (plusieurs réponses possibles) :

☒ Confronter les visions métier et technique

☒ Fournir au product owner un premier retour sur l'impact technique des besoins métier

☐ D'attendre sereinement l'estimation en sprint planning

☐ Convaincre les développeurs de la nécessité de réaliser les besoins

- Q C'est le premier retour de ce qu'implique techniquement le besoin. On fait une estimation grossière et il n'est pas nécessaire de convaincre les développeurs : c'est le product owner qui décide des besoins prioritaires et de la feuille de route.