

Développement agile
Hafedh Mili
Copyright 2012 - 2016

Développement agile

- Un ensemble de pratiques de développement logiciel qui mettent l'emphase sur:
 - Le pragmatisme (vs dogmatise)
 - La réactivité aux changements
 - L'implication constante du client
 - Le feedback fréquent à tous les niveaux

Agile Manifesto

- Des experts en méthodes de développement dites "agiles" ont concocté un *manifeste* que l'on peut consulter ici: www.agilemanifesto.org
 - **Individuals and interactions** over processes and tools
 - **Working software** over comprehensive documentation
 - **Customer collaboration** over contract negotiation
 - **Responding to change** over following a plan

Manifeste agile

- Valeurs fondamentales:
 - Les individus et leurs interactions plus que les processus et les outils.
 - Des logiciels opérationnels plus qu'une documentation exhaustive
 - La collaboration avec les clients plus que la négociation contractuelle.
 - L'adaptation au changement plus que le suivi d'un plan.

Principes agiles

(verbatim de http://fr.wikipedia.org/wiki/Manifeste_agile)

- "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 opérationnel avec des cycles de quelques semaines à quelques mois et 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 ils 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.
- Un logiciel opérationnel est la principale mesure d'avancement.
- 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 une bonne conception renforce l'Agilité.
- 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 autoorganisées.
- À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis règle et modifie son comportement en conséquence."

Méthodes agiles

(verbatim de http://fr.wikipedia.org/wiki/Manifeste_agile)

- "Le développement Agile, appelé aussi développement adaptatif, se caractérise donc par un style de conduite de projet itératif incrémental, centré sur l'autonomie des ressources humaines impliquées dans la spécification, la production et la validation d'une application intégrée et testée en continu"
- Exemples:
 - SCRUM
 - Extreme Programming (XP)
 - Crystal Clear
 - etc.

Méthode SCRUM

- Principes
- Processus
- Livrables

Principes de SCRUM

- SCRUM n'est pas un *processus de développement*
 - C'est un *cadre (framework)* pour le développement de *processus agiles* de développement
- Principes: contrôle du processus de développement
 - **La transparence**: les aspects du processus qui affectent le résultat sont visibles/observables (et conformes à la réalité)
 - **L'inspection**: les différents aspects du processus doivent être inspectés régulièrement pour que les écarts soient détectés rapidement
 - **L'adaptation**: si l'inspection révèle que le processus s'écarte de ce qui est attendu, le processus doit être adapté

Principes de SCRUM

- SCRUM comporte quatre activités d'inspection/adaptation:
 - *Sprint planning meeting*: réunion de planification d'un incrément (sprint)
 - *Daily SCRUM*: mêlée quotidienne (15 minutes, tout le monde debout, etc., voir plus loin)
 - *Sprint review meeting*: réunion de revue de l'incrément qui vient de se terminer (ce qui a été fait, ce qui reste à faire) pour mieux planifier le prochain
 - *Sprint retrospective*: réunion de revue de l'incrément qui vient de se terminer pour adapter le *processus-même* de développement

Contenu de SCRUM

- Équipes SCRUM
 - Rôles et caractéristiques
- Évènements (milestones?)
- Artéfacts (livrables)
- Règles

Équipe SCRUM

- Client (product owner)
- Équipe de développement (development team)
- SCRUM Master (gestionnaire de projet)

Équipe SCRUM: *product owner*

- Personne responsable pour maximiser la valeur du produit et du travail réalisé par l'équipe de développement.
- Responsable de la gestion du *product backlog*
 - *Bien expliquer ce qui est demandé*
 - *Priorise le développement*
 - *S'assurer que tout le monde (développement et métier) le comprennent*
- Cela doit être une personne et non un comité

Équipe SCRUM: développement

- Les personnes qui ... développent
- Les équipes doivent être structurées de sorte à être auto-gérées
 - Personne ne leur dicte *comment* livrer la fonctionnalité souhaitée ou comment se répartir le travail entre eux
 - Une équipe doit contenir toutes les compétences nécessaires au développement de l'incrément (analyse métier, architecture, utilisabilité, programmation, BD,...)
 - Pas de titres/postes à part "développeur"
 - Solidairement responsables
 - Pas de sous-équipes spécialisées

Équipe SCRUM: développement

- Taille idéale:
 - 7 personnes + ou - 2.
 - La composition de l'équipe demeure la même durant un incrément (*sprint*)

Équipe SCRUM: SCRUM Master

- Gestionnaire de projet et dépositaire de la méthodologie
- Il rend des services à:
 - L'équipe de développement
 - Au client (product owner)
 - À l'organisation, au complet

Équipe SCRUM: SCRUM Master

- Services au client
 - Trouver une façon de bien gérer le product backlog
 - Communiquer de façon claire la vision, buts, et les items du product backlog aux développeurs
 - Former l'équipe à créer des items de product backlog clairs et concis
 - Comprendre la planification long-terme d'un produit
 - Comprendre et pratiquer les méthodes agiles
 - Organiser/faciliter/modérer les divers "événements" de SCRUM

Équipe SCRUM: SCRUM Master

- Services aux développeurs
 - Coaching dans l'auto-gestion et la 'cross-functionality' (collaboration entre spécialités)
 - Enseigner et mener l'équipe de développement à la création de produits de grande valeur
 - Lever les obstacles à l'avancement de l'équipe de développement
 - Faciliter les événements SCRUM
 - Coaching de nouvelles équipes SCRUM

Équipe SCRUM: SCRUM Master

- Services à l'organisation
 - Enseigner et "vendre" l'adoption de SCRUM à l'organisation
 - Planifier la mise en place de SCRUM
 - Aider les employés et les parties prenantes à comprendre et adopter SCRUM
 - Mettre en oeuvre les changements qui amélioreront la productivité des équipes SCRUM
 - Travailler avec les autres SCRUM master de l'organisation pour y améliorer la pratique de SCRUM

Événements SCRUM

- Les événements et leurs fréquence assurent la transparence, l'inspection (et donc, l'adapation)
- Tous doivent être adoptés, avec la fréquence prescrite
- Les différentes activités sont limitées dans le temps (durée maximale), pour ne pas s'enfermer dans la gestionnité

Événements SCRUM: Sprint

- Le sprint représente l'incrément de développement, d'une durée maximale d'un mois, durant laquelle on doit livrer une fonctionnalité ayant une valeur pour le client
- Un sprint consiste en:
 - Planification
 - Développement
 - Revue
 - Rétrospective
- Les sprint se succèdent sans temps morts

Événements SCRUM: Sprint

- Durant un Sprint:
 - On n'accepte pas les changements pouvant affecter les objectifs (livrables) du sprint
 - L'équipe de développement reste constante
 - Les objectifs de qualité ne peuvent décroître
 - La portée du sprint peut-être clarifiée et re-négociée au besoin, au fur et à mesure que les choses se précisent
- La limitation à un mois réduit les risques d'échec, ou tout au moins, leur coût

Événements SCRUM: Sprint

- Annuler un sprint
 - Seul le client peut annuler un sprint en cours de route
 - En consultation avec le reste de l'équipe et les parties prenantes
 - Si les objectifs du sprint deviennent obsolètes
 - Changement de stratégie
 - Les fonctionnalités complétées durant ce sprint sont évaluées par le client pour un éventuel "release"
 - Celles qui ne le sont pas sont remises sur le *product backlog*

Événements SCRUM: Sprint planning meeting

- Deux objectifs, qui se partagent le temps de la réunion
 - Déterminer *ce qui doit être développé* durant ce sprint
 - Planifier le développement
- La durée de la réunion est de 1 jour par sprint de 1 mois
 - Et donc, 4 heures par sprint de 2 semaines

Événements SCRUM: Sprint planning meeting

- Déterminer le contenu du sprint
 - Inputs:
 - Product backlog
 - Le dernier incrément développement
 - Capacité de l'équipe
 - Historique de productivité de l'équipe
 - Seule l'équipe de développement peut déterminer ce qui peut-être développé durant le sprint.
 - Une fois les items déterminés, on les synthétise en un 'objectif', qui pourra guider les réajustements rendus nécessaires en cours de route.

Événements SCRUM: Sprint planning meeting

- Déterminer le comment du sprint (plan de projet)
 - Processus
 - Faire une conception de haut niveau des fonctionnalités choisies pour le sprint
 - Faire un plan de développement
 - => *sprint backlog*
 - Le travail durant les premiers jours du sprint doit être décomposé au niveau du jour (ce qui sera fait chaque jour)
 - Le *sprint backlog* est sujet à révision durant le sprint

Événements SCRUM: SCRUM meeting

- Une réunion quotidienne de 15 minutes dont l'objectif est de:
 - Déterminer ce qui a été accompli depuis la veille
 - Identifier les obstacles identifiés entre temps
 - Déterminer ce qui sera réalisé d'ici le lendemain
- Les développeurs peuvent se rencontrer (en sous-équipes) après cette réunion pour replanifier

Événements SCRUM: Sprint review

- Le client identifie ce qui a été complété et ce qui ne l' a pas été
- L'équipe de développement explique ce qui a bien fonctionné, les obstacles rencontrés, et les solutions adoptées
- L'équipe de développement fait une démo
- Le client discute le product backlog
- Client et développeurs discutent quoi faire au prochain sprint
- Une réunion de 4 heures / mois.

Événements SCRUM: Sprint retrospective

- L'équipe évalue ses performances durant le sprint qui vient de se terminer, et identifie des façons de s'améliorer
- Plus particulièrement
 - Évaluer les personnes, relations, processus et outils
 - Identifier et classer les choses qui se sont bien déroulées et identifier des améliorations potentielles
 - Créer un plan pour la mise en oeuvre/adoption de ces améliorations
- Une réunion de 3 heures / mois.

Livrables SCRUM

- Incrément! (le logiciel livré à la fin de chaque sprint)
- Product backlog
- Sprint backlog

Livrables SCRUM: product backlog

- Une liste priorisée de ce qui doit être fait sur produit
- La source unique d'exigences en ce qui concerne les changements à apporter au produit
- C'est un document évolutif: certaines fonctionnalités peuvent être bien comprises au début, et d'autres le seront moins
 - Elles seront précisées ou revues ou éliminées plus tard

Livrables SCRUM: product backlog

- Une liste comprend
 - Fonctionnalités
 - Features
 - Exigences non-fonctionnelles
 - Améliorations
 - Bug-fixes,
 - Etc.
- Chaque item est décrit par:
 - Description
 - Ordre de priorité
 - Estimation
 - Statut

Livrables SCRUM: product backlog

- Suivi d'avancement
 - Pour faire le suivi d'avancement *durant* un sprint, on peut mesurer, pour chaque item du sprint
 - Temps consommé (verre à moitié vide)
 - Temps restant à complétion (verre à moitié plein)
- Les deux sont utiles pour
 - Projections pour le sprint actuel
 - Futures projections

Livrables SCRUM: sprint backlog

- Contenu:
 - Identifie les items livrables pour le sprint actuel
 - Inclut un plan détaillé pour réaliser ces items là
- Le plan doit contenir assez de détail pour que l'on puisse percevoir l'avancement (ou pas) durant les daily scrums
- Si de nouvelles tâches apparaissent (s'avèrent nécessaires), il faut les ajouter
- Au fur et à mesure que l'on avance, on doit réviser l'estimation du temps restant pour compléter les items du sprint backlog
