

Méthode agile (source Wikipédia)

Introduction

Les **méthodes Agiles** sont des groupes de pratiques pouvant s'appliquer à divers types de projets, mais se limitant plutôt actuellement aux projets de développement en informatique ([conception de logiciel](#)). Les méthodes Agiles se veulent plus pragmatiques que les méthodes traditionnelles. Ces méthodes de développement visent à réduire le cycle de vie du logiciel (donc accélérer son développement) en développant une version minimale, puis en intégrant les fonctionnalités par un processus itératif basé sur une écoute client et des tests tout au long du cycle de développement.

L'origine des méthodes agiles est liée à l'instabilité de l'environnement technologique et au fait que le client est souvent dans l'incapacité de définir ses besoins de manière exhaustive dès le début du projet. Le terme « agile » fait ainsi référence à la capacité d'adaptation aux changements de contexte et aux modifications de spécifications intervenant pendant le processus de développement.

. Valeurs

Dans ce but, elles prônent 4 valeurs fondamentales (entre parenthèses, les citations du manifeste) :

- **L'équipe** (« Personnes et interaction plutôt que processus et outils ») : Dans l'optique agile, l'équipe est bien plus importante que les outils (structurants ou de contrôle) ou les procédures de fonctionnement. Il est préférable d'avoir une équipe soudée et qui communique, composée de développeurs (éventuellement à niveaux variables), plutôt qu'une équipe composée d'experts fonctionnant chacun de manière isolée. La communication est une notion fondamentale.
- **L'application** (« Logiciel fonctionnel plutôt que documentation complète ») : Il est vital que l'application fonctionne. Le reste, et notamment la documentation technique, est une aide précieuse mais non un but en soi. Une documentation précise est utile comme moyen de communication. La documentation représente une charge de travail importante, mais peut pourtant être néfaste si elle n'est pas à jour. Il est préférable de commenter abondamment le code lui-même, et surtout de transférer les compétences au sein de l'équipe (on en revient à l'importance de la communication).
- **La collaboration** (« Collaboration avec le client plutôt que négociation de contrat ») : Le client doit être impliqué dans le développement. On ne peut se contenter de négocier un contrat au début du projet, puis de négliger les demandes du client. Le client doit collaborer avec l'équipe et fournir un feed-back continu sur l'adaptation du logiciel à ses attentes.
- **L'acceptation du changement** (« Réagir au changement plutôt que suivre un plan ») : La planification initiale et la structure du logiciel doivent être flexibles afin de permettre l'évolution de la demande du client tout au long du projet. Les premières *releases* du logiciel vont souvent provoquer des demandes d'évolution.

Principes

Ces 4 valeurs se déclinent en 12 principes généraux communs à toutes les méthodes agiles :

- « Notre première priorité est de satisfaire le client en livrant tôt et régulièrement des logiciels utiles »
- « Le changement est bienvenu, même tardivement dans le développement. Les processus agiles exploitent le changement comme avantage compétitif pour le client »
- « Livrer fréquemment une application fonctionnelle, toutes les deux semaines à deux mois, avec une tendance pour la période la plus courte »
- « Les gens de l'art et les développeurs doivent collaborer quotidiennement au projet »
- « Bâissez le projet autour de personnes motivées. Donnez leur l'environnement et le soutien dont elles ont besoin, et croyez en leur capacité à faire le travail »
- « La méthode la plus efficace pour transmettre l'information est une conversation en face à face »
- « Un logiciel fonctionnel est la meilleure unité de mesure de la progression du projet »
- « Les processus agiles promeuvent un rythme de développement durable. Commanditaires, développeurs et utilisateurs devraient pouvoir maintenir le rythme indéfiniment »
- « Une attention continue à l'excellence technique et à la qualité de la conception améliore l'agilité »
- « La simplicité - l'art de maximiser la quantité de travail à ne pas faire - est essentielle »
- « Les meilleures architectures, spécifications et conceptions sont issues d'équipes qui s'auto-organisent »
- « À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis accorde et ajuste son comportement dans ce sens »

Projet :

Dans notre logiciel de e-Learning AREL, il manque quelque chose aux étudiants pour s'entraîner avant de venir à l'examen.

L'idéal est d'avoir pour chaque matière une batterie de questions probablement classées par ordre de difficultés. Pour chacune de ces questions on peut avoir une ou plusieurs réponses justes.

Le logiciel Arel peut proposer aux étudiant qui le souhaite un jeu de QCM pour une matière donnée, et fournir soit un corrigé directement ou lui signifié où il a réussi et où il a répondu faux, de cette manière l'étudiant peut re-tester ce même jeu de QCM ou en demander un autre.

Proposer une application ou un module qui s'intègre parfaitement à AREL pour répondre à ces besoins immédiats. Votre module ou application utilisera les mêmes ressources et les mêmes modules déjà existants dans AREL (module d'authentification, gestion des matières, ...)

Pour la première phase, on souhaite avoir une application minimaliste qui tournera au plus tard dans moins de 15 jours