

# Agile

Thomas Clavier  
tclavier@azae.net

# Qui suis-je ?

Thomas Clavier <tclavier@azae.net>

Consultant indépendant, enseignant chercheur, entrepreneur,  
passionné d'agilité et de logiciel libre.

# C'est quoi ?

Être agile et pas faire de l'Agile.

C'est avant tout un état d'esprit partagé par l'ensemble des participants à un projet.

# Être agile

"Les firmes qui survivent dans le long terme ne sont pas celle qui sont les plus fortes ou les plus intelligentes, mais celles qui s'adaptent le mieux aux changements d'environnement"

Hiroshi Okuda, Toyota

# Pourquoi l'agilité ?

- Méthode en V
- Effet tunnel
- Contrat
- Perdant / perdant !

# Le changement en entreprise

- On avance en dessous !

# Qui est là ?

- Tours de tables, expérience et attentes de chacun.
- Arbre à mots.

# Coût, Qualité, Délai



# Règle des 80/20

# Le manifeste agile

**Individuals and interactions** over processes and tools  
**Working software** over comprehensive documentation  
**Customer collaboration** over contract negotiation  
**Responding to change** over following a plan

# L'équipe

« **Personnes et interaction plutôt que processus et outils** »

Dans l'optique agile, l'équipe est bien plus importante que les moyens matériels ou les procédures. Il est préférable d'avoir une équipe soudée et qui communique composée de développeurs moyens plutôt qu'une équipe composée d'individualistes, même brillants. 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 secondaire, même si une documentation succincte et 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.

# Les 12 principes

- « 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 ».

# Les 12 principes

- « 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 soutenable. Commanditaires, développeurs et utilisateurs devraient pouvoir maintenir le rythme indéfiniment ».



# Les 12 principes

- « 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 intervalle régulier, l'équipe réfléchit aux moyens de devenir plus efficace, puis accorde et ajuste son comportement dans ce sens »

# Valeurs et principes

Des cycles court, un produit en production à chaque fin de cycle et des producteurs de valeurs qui s'améliorent continuellement.

# Une question de culture

- L'individualisme des cultures judaïc : dieu dicte, dieu reviendra pour le jugement dernier !
- Le collectif des cultures orientales : pas de dieux, juste des hommes

# Toyota

- 1950, Toyota
- Kaizen : amélioration continue, « Mieux qu'hier, moins bien que demain. »
- 5S : propre et ordonné, mais surtout organisé selon l'usage
- Kanban : les fiches sur les chaînes de montage

# D'autres exemple

- 1983 : Yamaha et MBK
- Canon et son Canon Products System
- 1980 : Sony, Fujitsu
- 2001 : La poste japonaise, Ito-Yokado avec l'aide de Toyota

# Favi : 30 ans d'agilité

- adaptations de toutes les méthodes japonaise aux contexte local
- des fiches
- L'homme est bon => pas de sanction mais de l'action
- Quoi ? Qui ? = Pourquoi ?
- 1 jour, 20 ans

# Les grands du web

- 2 pizzas team
- devops chez amazon
- du nombre de relations à l'organisation des sociétés
- créer des groupes sociaux culturel
- privilégier le "time to market", donc le client !
- générique implique complexité donc délais + coût

# Les méthodes

- Lean
- Scrum
- kanban
- devops
- Extreme Programming



# Principe

- Travailler ensemble : collectif
- Concentré sur un sous ensemble de tâches
- Simplicité, efficacité, qualité

# Kanban

カンバン

- Un tableau
- des fiches
- de la bonne humeur

# 10 étapes

- Aucune tâche technique
- De petites tâches priorisées
- Fini
- Bac rouge
- Ajouter des colonnes
- Cumulative flow diagram
- Limiter l'encours
- Plusieurs lignes
- Ligne rouge
- Retrospectives

# Scrum

- un backlog produit
- un backlog d'itération
- des rendez-vous
- un produit

# Les rendez-vous Scrum

- Planning Poker
- Découpage en tâches
- Mêlée quotidienne
- Démonstration
- Rétrospective

# Le tableau scrum



# Scrum



# 3 petits jeux

- Communication et collaboration