

Quelques slides sur les Méthodes Agiles et Questionnement sur les modèles

Hervé Leblanc & Jean-Michel Bruel

IRIT-Université de Toulouse

Macao team

Mal aimées du génie logiciel ?

- Peu ou prou de communications scientifiques académiques.
- Littérature par des practitioners et des coachs.
- Une conférence IEEE dédié : XP comme....
- Autres événements :
 - Conférence Agile (Scrum Alliance), événement satellite de SPLASH (@OOPSLA)
 - Scrum Days (nouveaux ateliers)
 - Agile Tour (Evangélisation)

Méthodes Agiles

utilisant peu ou pas de modèles

- eXtreme Programming (Kent Beck)
 - Test Driven Development, Refactoring, Simple design, Pair Programming,...
 - Modèles : métaphore du système (un ensemble de stories expliquant le fonctionnement du système)
- Scrum (Ken Schwaber)
 - Management de projets, imprédictibilité et complexité du développement logiciel.
 - Modèles : Planning and Architecture/High level design dans la pre-game phase.

Méthodes Agiles

utilisant peu ou pas de modèles

- Famille de process Crystal (Alistair Cockburn)
 - De clear à rouge en fonction de la taille et de la criticité du projet.
 - Modèles : Du tableau blanc à des modèles contractuels.
- Feature driven development (Palmer and Felsing)
 - Conception et construction de features en //
 - Modèles : un chief architect, modèles objets du domaine, modèle d'architecture du système

Méthodes Agiles

utilisant peu ou pas de modèles

- Dynamic Systems Development Method (Stapleton)
 - RAD ou approche par prototypage
 - Modèles : Business Area Definition, Functional models (prototype code and analysis models)
- Adaptative Software Development (Highsmith)
 - Management des interactions et basé sur des composants
- Pragmatic Programming (Hunt and Thomas)
 - Bonnes pratiques du programmeur

Méthodes Agiles utilisant des modèles


- RUP & UP (Kruchten)
 - Utilisation intensive de modèles.
 - Pas considéré comme « Agile » par la communauté.
 - Allègements vers l'agilité : Basic Unified Process, OpenUP, Agile Unified Process,...
- Agile Modeling (Scott Ambler)
 - Activités de modélisation en utilisant les principes Agiles
 - « modeling tools are not as useful as usually thought »
 - « big up-front design is not required »

Caractérisation (1)

- Hypothèse des méthodes orientées process :
 - Les requirements d'un projet sont déterminés et figés avant le commencement des phases d'analyse, de conception et de développement.
- Hypothèse des méthodes agiles :
 - Adaptabilité au changement en continu.
 - Micro-fonctionnalité : client sur site, adéquation des tests d'acceptation et des tests unitaires.
 - Macro-fonctionnalité : priorités rediscutées entre 2 sprints ou itérations.
 - Nécessité d'un design simple et de tests automatisés pour assurer la non-régression du code.

Caractérisation (2)

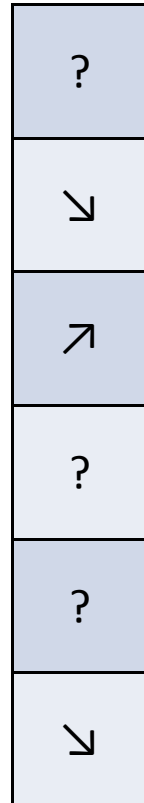
- Les quatre valeurs du manifeste agile :

- 
1. **Individuals and interactions** over processes and tools
 2. **Working software** over comprehensive documentation
 3. **Customer collaboration** over contract negotiation
 4. **Responding to change** over following au plan

Caractérisation (3)

- Les douze principes du manifeste agile :

1. Notre plus haute priorité est de satisfaire le client en livrant rapidement et régulièrement des fonctionnalités à grande valeur ajoutée.
2. Accueillez positivement les changements de besoins, même tard dans le projet. Les processus agiles exploitent le changement pour donner un avantage compétitif au client.
3. Livrez fréquemment un logiciel opérationnel avec des cycles de quelques semaines à quelques mois et une préférence pour les plus courts.
4. Les utilisateurs ou leurs représentants et les développeurs doivent travailler ensemble quotidiennement tout au long du projet.
5. 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.
6. 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.



Caractérisation (4)

- Les douze principes du manifeste agile :
 7. Un logiciel opérationnel est la principale mesure d'avancement.
 8. 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.
 9. Une attention continue à l'excellence technique et à une bonne conception renforce l'agilité.
 10. La simplicité – c'est-à-dire l'art de minimiser la quantité de travail inutile – est essentielle.
 11. Les meilleures architectures, spécifications et conceptions émergent d'équipes autoorganisées.
 12. À intervalles réguliers, l'équipe réfléchit aux moyens de devenir plus efficace, puis règle et modifie son comportement en conséquence.

