

# **M3301 : Méthodologie de la production d'applications - Introduction & Teasing**



*Isabelle BLASQUEZ*  
*@iblasquez*

*Septembre 2016*

# Bonne rentrée !



Images : <http://petit-pikachu.skyrock.com/1969658399-pikachu-sur-sa-boue.html>  
<http://diylol.com/meme-generator/chemistry-cat/memes/are-you-ready-for-school-children-i-know-i-am>



**Isabelle BLASQUEZ**



[@iblasquez](https://twitter.com/iblasquez)

**Enseignement : Informatique – Université de Limoges**

**Recherche : Développement logiciel agile**



ICSTUG #IUTAgile



**#Software  
Craftsmanship**

**USEOMIX** LIMOUSIN

Retrouvez ce support sur : <https://github.com/iblasquez/iut-m3301-agile>



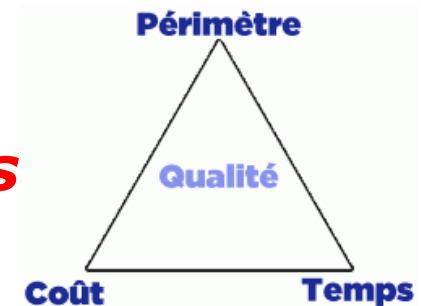
# Introduction

## **Projet Informatique :**

Ensemble d'actions mises en œuvre, afin de produire les résultats et fournitures définies en réponse aux objectifs clairement définis

- dans des **délais fixés** (date début et date de fin)
- mobilisant des **ressources humaines et matérielles**
- possédant un **coût prévisionnel et des gains espérés**

*L'objectif d'un processus de développement est de produire des logiciels de **qualité qui répondent aux besoins des utilisateurs dans des temps et des côts prévisibles...***



Le **triangle de projet**  
(ou **triangle de qualité**)

# Qu'est-ce que la qualité logicielle ?

La **qualité logicielle** est une **appréciation globale d'un logiciel**, basée sur de nombreux indicateurs <sup>1</sup>.

La norme **ISO/CEI 9126** définit une **classification** qui décrit la **qualité logicielle** au travers de **6 grands indicateurs** (caractéristiques)

**Capacité fonctionnelle**

**Facilité d'utilisation**

**Fiabilité**

**Rendement/Efficacité**

**Maintenabilité**

**Portabilité**

<sup>1</sup> A.April & C.Laporte, Assurance qualité logicielle 1: concepts de base, Lavoisier, 2011

# La norme ISO 9126 : 6 caractéristiques – 27 sous-caractéristiques un langage commun autour des attributs de la qualité logicielle

## Capacité fonctionnelle

Pertinence, Exactitude, Interopérabilité, Sécurité, Conformité

## Facilité d'utilisation

Maturité (faible fréquence d'apparition des incidents)  
Tolérance aux pannes  
Facilité de récupération

## Fiabilité

Facilité de compréhension  
Facilité d'apprentissage  
Facilité d'exploitation

## Rendement/Efficacité

Comportement temporel (temps de réponse, taux de transactions)  
Utilisation des ressources (mémoire, processeur, disque et réseau)

## Maintenabilité

Facilité d'analyse (identification dans le logiciel de l'origine d'un défaut constaté)  
Facilité de modification  
Stabilité  
Testabilité

## Portabilité

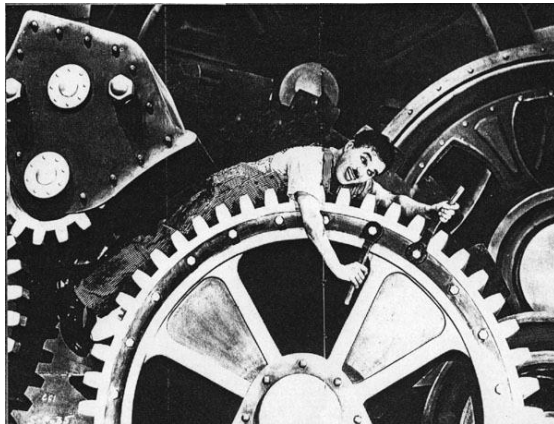
Facilité d'adaptation (à des changements de spécifications ou d'environnements opérationnels)  
Facilité d'installation  
Coexistence  
Interchangeabilité

# Qualité Interne et Qualité externe

**La qualité interne** est liée aux **artefacts de développement** et **la qualité externe** est mesurée par la **satisfaction des exigences client** <sup>1</sup>

## Les métriques internes

*can be applied to a non-executable software product  
(such as a specification or source code)  
**during designing and coding.*** <sup>2</sup>



***product right ?***

***Construire le produit correctement ?***

## Les métriques externes

*use measures of a software product derived from  
**measures of the behaviour of the system**  
of which it is a part, by testing, operating and  
observing the executable software or system.* <sup>2</sup>



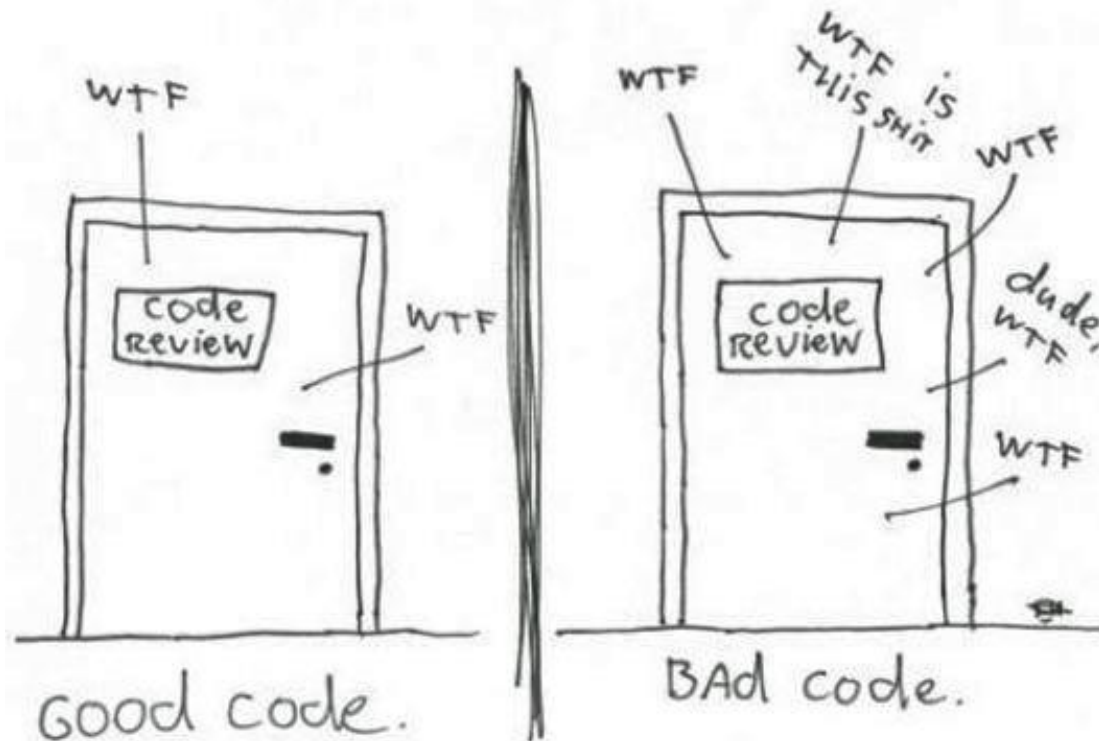
***right product ?***

***Construire le produit correct ?***



# Comment mesurer la qualité ...

The ONLY VALID MEASUREMENT  
OF CODE QUALITY: WTFs/MINUTE



(c) 2008 Focus Shift/OSNews/Thom Holwerda - <http://www.osnews.com/comics>



# ... ou plutôt comment assurer une certaine qualité ...

## Tests

*M2103 - M2104*  
*M3301 - M3105*

## Métrique

*M2104*

## Méthodologie

*M3301*

## Standards de Codage

*M2103 - M2104*

## Refactoring

*M2104 - M3105*

## Normes

*M3301*

## Revue de code

*M2104*

Revue de conception  
Preuve de programme  
Vérification de modèle

...

# Quelques normes et autres initiatives autour de la qualité logicielle

Initiative	Problématiques abordées	Acteurs concernés
ISO/CEI 9126	Description des caractéristiques, modèle qualité du produit final	Acquéreur, fournisseur de logiciel
ISO/CEI 14598	Démarche méthodologique pour l'évaluation d'un produit logiciel au regard des caractéristiques de la norme ISO/CEI 9126	Acquéreur, fournisseur de logiciel
ISO/IEC 25041	Démarche méthodologique pour l'évaluation d'un produit logiciel intégrant ISO/CEI 9126 et ISO/CEI 14598 <b>SQuaRE</b> (Systems and software Quality Requirements and Evaluation)	Acquéreur, fournisseur de logiciel, évaluateurs indépendants
ISO/IEC 12207	Déclinaison des activités du cycle de vie en processus	Équipes IT
ISO/IEC 15504	Évaluation des processus et de la maturité de l'organisation	Équipes IT
CMMi	Évaluation des processus et de la maturité de l'organisation	Entreprise

**CMMi**  
**(Capacity Maturity Model Integrated)**  
initiative plus globale  
puisque'elle ne vise pas uniquement  
la qualité logicielle d'un point de vue  
produit, mais s'intéresse au niveau de  
maturité de l'organisation qui élabore le  
logiciel et propose 5 niveaux de maturité.

**IEEE 1059**-1993 : Guide for Software Verification and Validation Plan  
**IEEE 829**-2008 : Standard for Software and System Test Documentation  
**IEEE 1028**-2008 : Standard for Software Reviews and Audits

...

**ITIL (Information Technology Infrastructure Library)** pour (Bibliothèque pour l'infrastructure des technologies de l'information) est un **ensemble d'ouvrages recensant les bonnes pratiques** du **management du système d'information**.  
*Comment organiser un SI ? Comment améliorer un SI ? Réduire les risques. Augmenter la Qualité de service, ...*  
*En savoir plus : <http://www.itilfrance.com/>*

# Et bien sûr dans tout projet, il y des risques qu'on voudrait éviter et auxquels il faut faire face et s'adapter rapidement pour garantir une certaine qualité ...

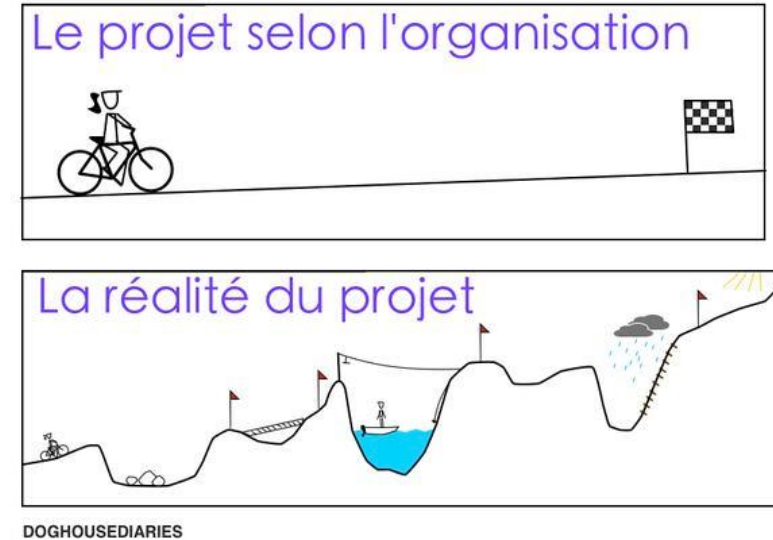
✓ **Un risque** peut être défini comme la probabilité qu'un événement, un danger, une menace ou une situation arrive, et que les conséquences indésirables qui en découlent constituent un problème potentiel.

✓ 2 types de risques :

→ **risques liés au produit** : à la **qualité** du produit

→ **risques liés au projet** qui menacent la capacité de ce dernier à atteindre ses objectifs:

- les **facteurs organisationnels** (coûts et délais à respecter, ressources humaines à gérer)
- les **problème techniques**



# Mais alors que va-t-on trouver dans ce module ?



# Dans ce module, nous nous intéresserons principalement au processus de développement autour du **Produit**



Sab En Bzh  
@SabEnBzh

Peu importe les ressources dont tu disposes, si tu ne sais pas les utiliser ça ne sera jamais suffisant.

**Pour le côté entreprise,  
ressources, management**

**voir le module M3204 :  
Gestion des systèmes  
d'information**



Extrait : <https://twitter.com/SabEnBzh/status/414291164703891456/photo/1>

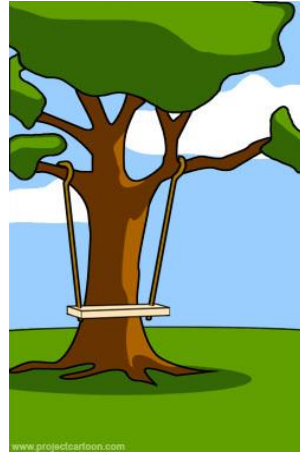
Isabelle BLASQUEZ - 2016



# Et parce qu'on voudrait éviter qu'un projet se déroule de la sorte...



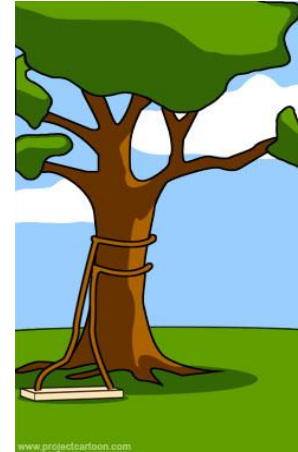
Ce que  
le client  
a expliqué



Ce que  
le chef de projet  
a compris



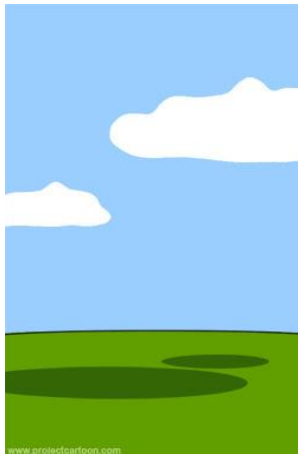
Ce que  
l'analyste  
a conçu



Ce que  
le développeur  
a fait



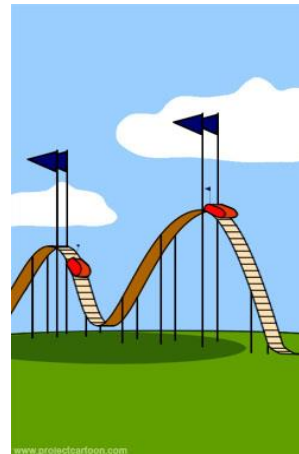
Comment  
les commerciaux  
ont décrit le produit



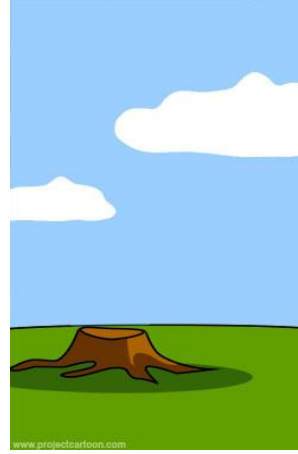
Comment  
le projet  
a été documenté



Ce que  
la production  
a installé  
chez le client



Ce que  
le client  
a été facturé



Comment  
le support technique  
est effectué



Ce dont  
le client  
avait réellement  
besoin

Réalisé à partir de : <http://www.projectcartoon.com/>

**... et qu'on préférerait**

**que de vraies interactions  
entre les acteurs du projet existent**

**et qu'une vraie valeur métier soit apportée**

**pour livrer un produit de qualité ...**



# Parce que la vision du métier de développeur est bien trop souvent erronée ...



Antoine Vernois @avernois ·

et toujours cette vision développeur = ouvrier. Assimilation d'un processus créatif a un travail à la chaîne :(



Antoine Vernois  
@avernois

Reminder: the job of a developer is not to write code but create software. Writing code is just a part of it.

14:13 - 25 juil. 2016



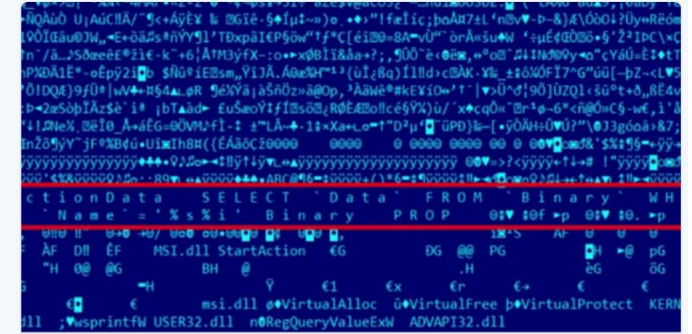
Gaspard  
@Gaspard\_PO

Apprendre à écrire ne me transforme pas en écrivain, apprendre à coder n'est pas suffisant pour être développeur.



Arnaud LEMAIRE @Lilobase · 19 août

1) Par où commencer avec cet article de @lemondefr ?



Faire du « code » pour travailler tout de suite

Maîtriser le développement informatique permet de booster son CV. Les entreprises apprécient les compétences opérationnelles des candidats is...

lemonde.fr



Arnaud LEMAIRE @Lilobase · 19 août

2) Déjà cette croyance que nous manquons de dev... Nous manquons de dev qualifié, et étrange ça ne prend pas 2 mois de former ces profils...



Arnaud LEMAIRE @Lilobase · 19 août

3) Ensuite le dev ne se limite pas à écrire du code mais à bien répondre à un besoin métier et ça aussi ça met plus de 2 mois à s'apprendre

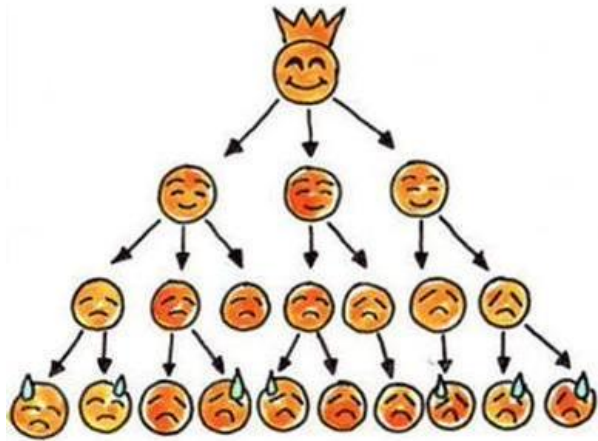
etc ...

... et que les **développeurs qualifiés**  
devraient plutôt être considérés comme  
de vrais **artisans** (du logiciel)  
qui imaginent, **créent** et **façonnent proprement**  
leur produit

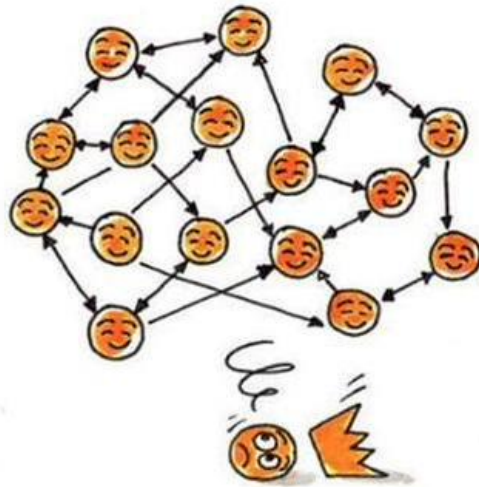


# Parce qu'un nouvel état d'esprit apparaît ...

*Depuis toujours, c'est ainsi...*



*Mais un jour peut-être !*



**Responsabilité Collective**  
**Intelligence Collective**

**Co-création**  
**Co-décision**  
**Collaboration**

...et que le **savoir-être**  
(soft skills)  
devient aussi important  
que le savoir-faire

...et que la **créativité**  
a une place de plus en plus  
prépondérante

Les compétences qu'il convient d'avoir  
aujourd'hui... et demain



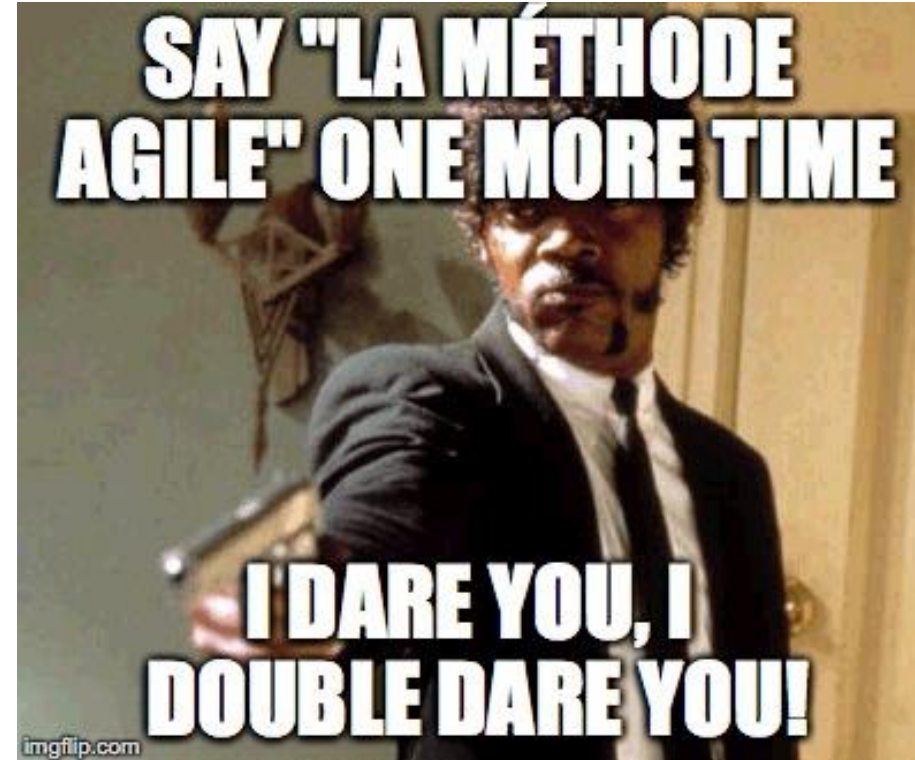
Extrait : <https://twitter.com/babgi/status/761871613272915968>



# Ce module se focalisera sur le **développement agile de logiciels** et la culture d'entreprise qu'il induit ...



Extrait : [www.agilemanifesto.org](http://www.agilemanifesto.org)

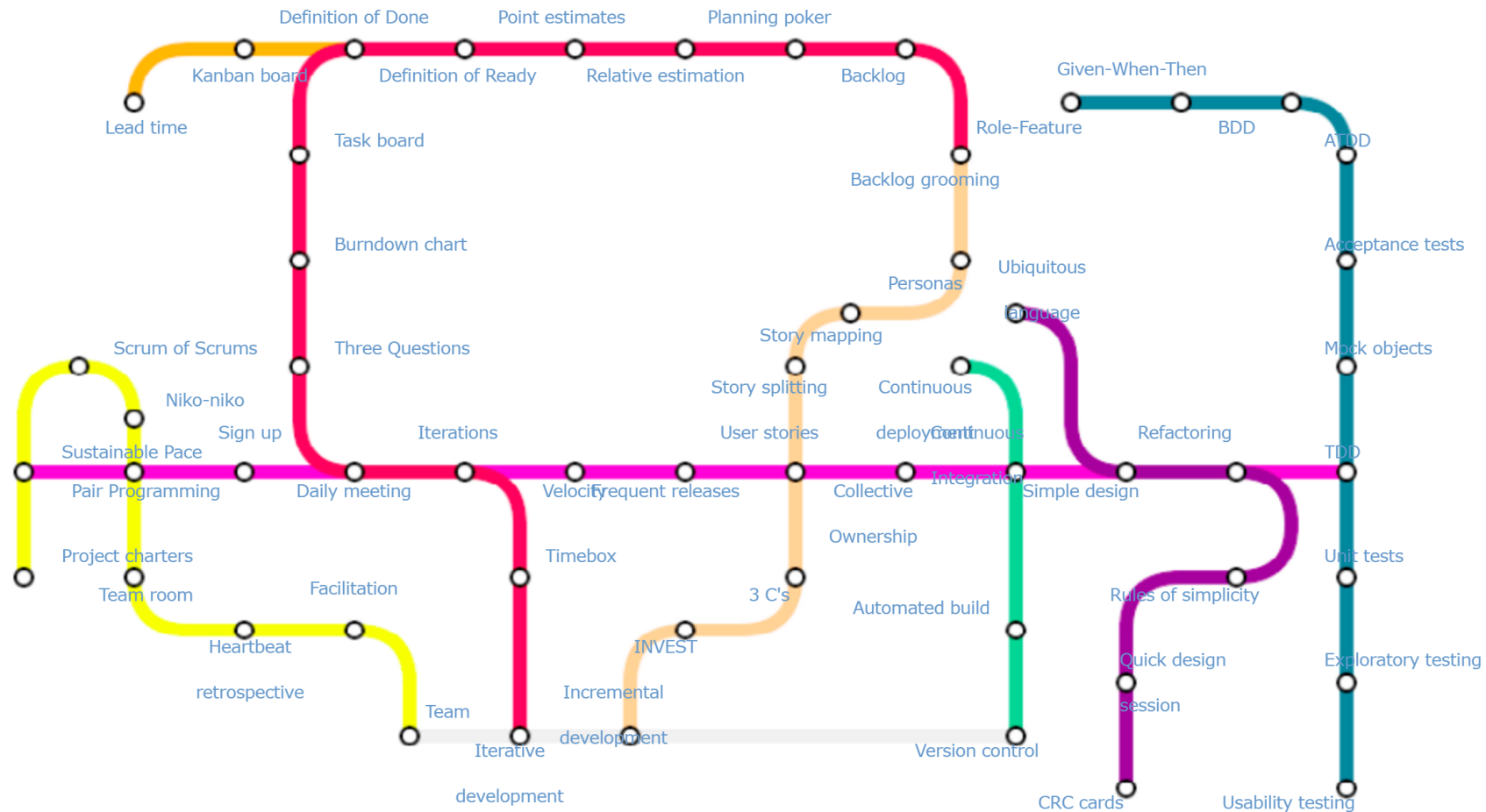


Extrait : <http://blog.soat.fr/2016/02/la-methode-agile-nexiste-pas>

Voir aussi : [https://medium.com/@cyril\\_lakech/l-agile-n-est-plus-sexy-c-est-la-norme-le-nouveau-challenge-des-agilistes-survivre-76cfc467fe6f#.kjbmrqsu](https://medium.com/@cyril_lakech/l-agile-n-est-plus-sexy-c-est-la-norme-le-nouveau-challenge-des-agilistes-survivre-76cfc467fe6f#.kjbmrqsu)

# Ecosystème des pratiques agiles ...

Carte interactive disponible sur <https://www.agilealliance.org/agile101/subway-map-to-agile-practices>



Lines represent practices from the various Agile "tribes" or areas of concern:

Extreme Programming  
Teams  
Lean

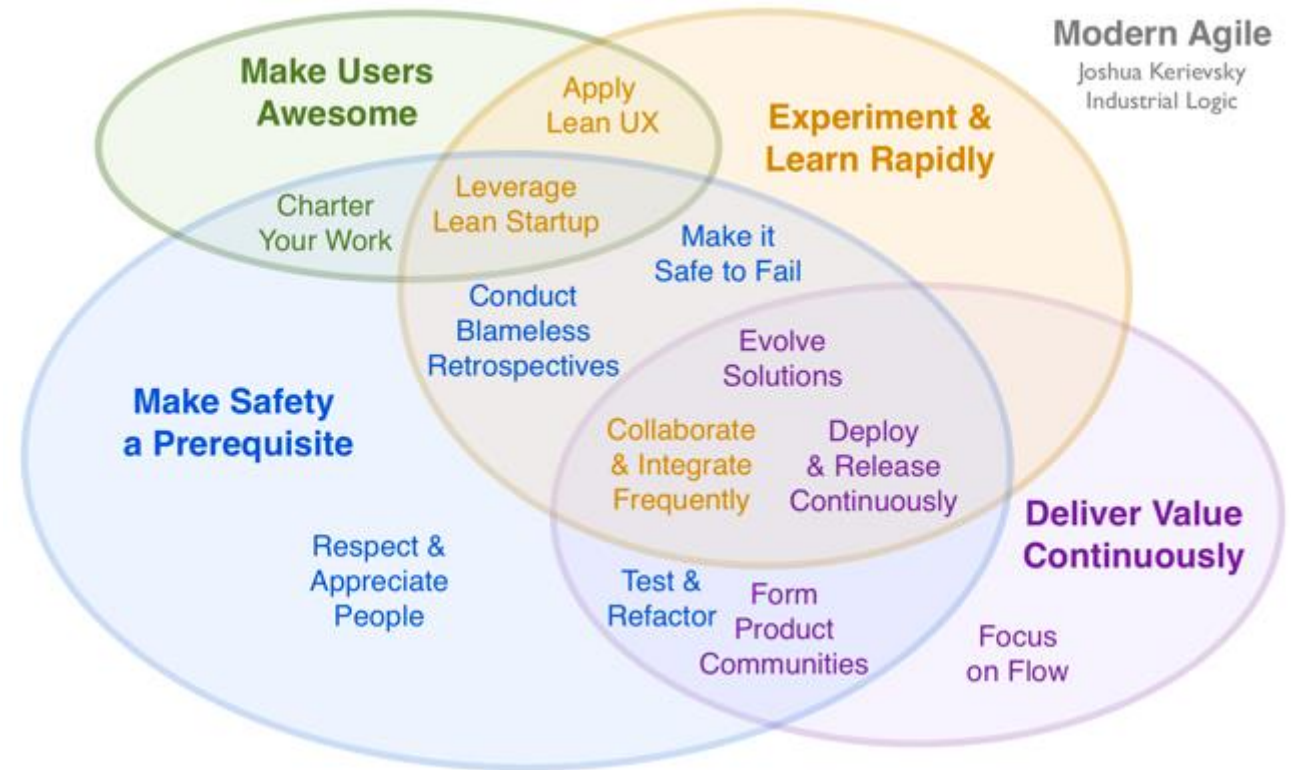
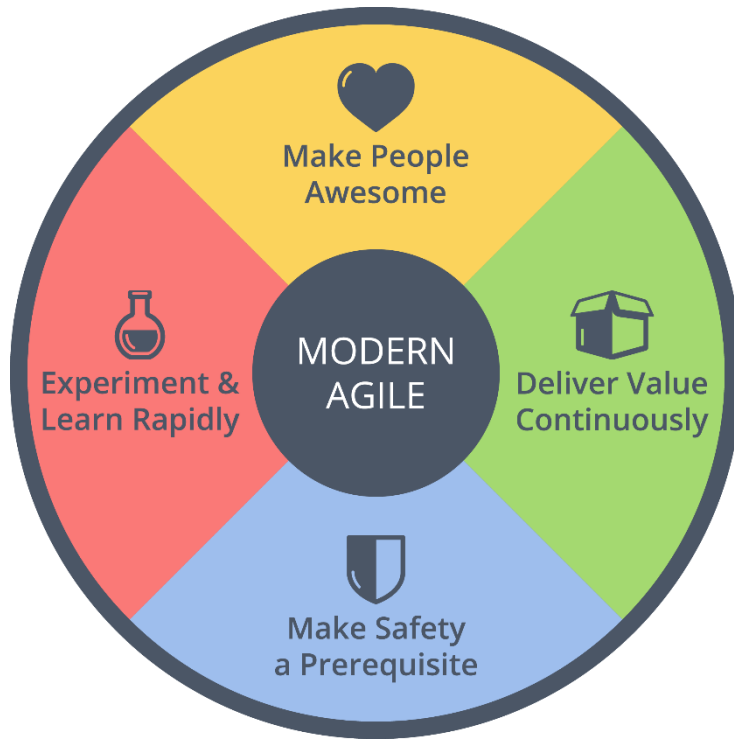
Scrum  
Product management  
Devops

Design  
Testing  
Fundamentals



# **Annexes**

# Vers une révision du manifeste agile ?



Modern Agile  
Joshua Kerievsky  
Industrial Logic

Extraits : <https://www.infoq.com/news/2016/08/agile2016-modern-agile>  
<https://www.industriallogic.com/blog/modern-agile/>

**A lire : Survey on the Need for an Agile Manifesto 2.0 :**  
<https://www.infoq.com/news/2016/08/survey-agile-manifesto>

# A méditer ...



Gilles Roustan

Artisan développeur Web



» Blog

» A propos de moi

## Artisan développeur

12/06/2016

### Avant-propos

Cet article est une retranscription de la conférence que j'ai donnée lors de l'agile tour de Montpellier le 13 octobre 2015. Pour mieux suivre, je vous conseille de regarder en parallèle les [slides de la conférence](#).

### A propos de moi

Je m'appelle Gilles et je suis artisan développeur.

Dans cet article, je vais vous parler de moi, de mon métier et de ma famille.

Plus particulièrement de l'évolution de la perception de mon métier et comment le métier de mon père et de mon grand-père m'ont aidé à devenir un développeur plus heureux.

### La mode du software craftsmanship

Il n'y a pas si longtemps, j'ai changé de poste et quand j'ai fait mon CV, j'ai mis que j'étais **artisan développeur**.

Il y a quelques années, j'ai vu débarquer cette *mode* lors de [conférences](#), dans des articles sur Software Craftsmanship.

J'ai trouvé ça sympa comme titre : artisan développeur ! Un contraste entre du moderne et du traditionnel. Et moi comme un mouton, j'ai trouvé ça cool et je me suis dépêché de modifier mon profil twitter.

# Et aussi ...

## How Can You Say You're A Software Crafts(wo)man?

Or what makes you a Software Crafts(wo)man

**T**his is a question that came up yesterday during our latest meetup with our local [French Riviera Software Craftsmanship Community](#) (note: a very nice round table session [on the beach](#) of Juan Les Pins ☺).

This post is my answer to this question.

### What Software Craftsmanship is not

**It's not an elitist club** of people who think to hold the truth of how to write the perfect code. This is an important aspect because it is related to how much inclusive or exclusive a group of people is (and also to how much nice people are). I've been following the international Software Craftsmanship movement since some time now and I met some of them during [SoCraTes UK](#) and I learned how much important are things like kindness, mentorship and empathy for a Software Crafts(wo)man. Software development is not only techniques and programming: it's also relationship and communication with your peers.

**It's not a certification** of one's capabilities to write good software (assuming it does exist really). A person yesterday was doubtful whether he could call himself a *software craftsman* or not because he is not working in TDD or applying all the time the SOLID principles.

These are technicalities, very important ones, but they are just tools that a person who cares about what he does can use or not in his daily job.

**It's not in conflict with the Agile movement.** Software Craftsmanship is focused a lot more on technical practices and writing quality software, but the goal is always the same: deliver values for the customer. The only difference is

Article complet disponible sur : <https://medium.com/alebaffa-blog/how-can-you-say-youre-a-software-crafts-wo-man-48ebc055ba9d#.byqyxzdjx>

Isabelle BLASQUEZ - 2016

# Bingo !

<b>Tests</b>	<b>Interactions</b>	<b>Right product</b>	<b>Responsabilité Collective</b>	<b>Créativité</b>
<b>Valeur métier</b>	<b>Intelligence Collective</b>	<b>Facilité d'utilisation</b>	<b>Risque</b>	<b>Collaboration</b>
<b>Savoir faire</b>	<b>Savoir être</b>	<b>Qualité</b>	<b>Maintenabilité</b>	<b>Refactoring</b>
<b>Product right</b>	<b>Logiciel opérationnel</b>	<b>Individus</b>	<b>Adaptation au changement</b>	<b>Capacité Fonctionnelle</b>
<b>Revue de code</b>	<b>Co-création</b>	<b>Besoins métier</b>	<b>Produit</b>	<b>Artisan logiciel</b>

Le premier joueur qui termine une ligne de 5 cases doit crier "BINGO" !  
Le premier joueur qui complète sa grille doit crier "SUPER BINGO" !