

Par Clément Côté, 2C Solutions
IIBA Montréal, le 22 mars 2017

L'agilité au quotidien pour les analystes d'affaires

1

Clément Côté

- Consultant en analyse d'affaires avec plus de 25 ans d'expérience en ingénierie, en gestion de projets et en amélioration de processus
- Co-fondateur de PCMS Conseil, firme de consultants et développement informatique
- Président de 2C Solutions, firme promouvant les meilleures pratiques en analyse d'affaires
- Directeur du programme de certification universitaire en analyse d'affaire (CUAA) de l'université Laval

2

Objectifs de cette présentation

- Expliquer les particularités du rôle de l'analyste d'affaires dans un projet agile
- Décrire les attentes de l'organisation envers l'analyste, au-delà des rôles propres aux méthodes agiles
- Comprendre la mécanique propre aux processus de développement agile
- Expliquer les outils courants de l'agilité que les analystes d'affaires doivent utiliser

3

Qu'est-ce que l'agilité?

- C'est avant tout une manière de penser différente plutôt qu'une méthode particulière
- L'agilité est souvent associée au développement logiciel, mais elle ne s'y limite pas
 - Par contre, bien des organisations considèrent l'agilité seulement au moment de développer des logiciels
 - L'agilité est surtout associée à de diverses approches de développement itératives
- L'agilité fait ultimement référence à la capacité à répondre, à réagir et à s'adapter à un environnement fluide
- L'agilité met en évidence des valeurs comme la collaboration, la qualité et l'adaptabilité aux attentes évolutives des clients

4

Analyse d'affaire et agilité

- La plupart des approches agiles ne prévoient pas un rôle spécifique pour *l'analyste d'affaires*
- *L'analyse d'affaires* est vue comme une compétence de l'équipe plutôt qu'un individu
 - Les activités d'analyse d'affaires peuvent être...
 - effectuées par un analyste d'affaire membre de l'équipe
 - effectuées par un représentant du client ou le propriétaire du produit
 - distribuées à travers divers membres de l'équipe
- L'analyse d'affaire est essentielle pour:
 - Préserver les intérêts de l'organisation
 - Maximiser les interactions entre les parties prenantes pour que celles-ci adhèrent à une vision commune de la solution offrant la plus grande valeur à l'organisation

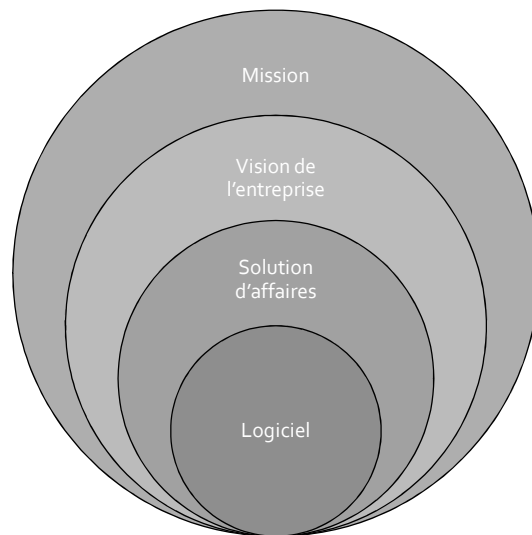
5

L'analyste d'affaires et l'agilité

- Les approches agiles ont pour objectif ultime de produire du logiciel
- L'analyste d'affaires a pour objectif ultime de permettre à l'organisation de *rencontrer sa vision*, de *combler ses lacunes* et de *résoudre ses problèmes d'affaires*
 - Le domaine d'intervention de l'analyste d'affaire est donc plus large que le seul logiciel
- L'analyste d'affaires n'est donc pas au service d'un processus de développement de logiciel, mais bien au service de l'organisation
 - Par contre, il doit adapter ses façons de faire aux processus utilisés

6

Le logiciel et l'entreprise



7

Autres responsabilités de l'analyste d'affaires

- Les analystes, dans le cadre de projets ou même des opérations courantes, se consacrent à une multitude d'activités sans lien direct avec le développement de logiciels
 - Résolution de problèmes d'affaires
 - Recherche des causes fondamentales et de solutions aux problèmes
 - Analyse et amélioration de processus d'affaires
 - Analyse de règles d'affaires
 - Étude des politiques de l'entreprise
 - Soutien à la prise de décisions
 - Facilitateurs de changements dans l'organisation

8

L'analyste agile

- **Adaptability.**
 - **Goal orientation.**
 - **Innovation.**
 - **Leadership.**
 - **Empathy.**
 - **Business orientation.**
 - **Anticipation.**
- L'analyste doit s'adapter aux méthodes utilisées

Toujours garder à l'œil les intérêts de l'organisation

Une solution à un problème d'affaires complexe n'est pas limitée à une seule chose

L'analyste doit interagir et communiquer

L'analyste est l'agent du changement dans l'organisation

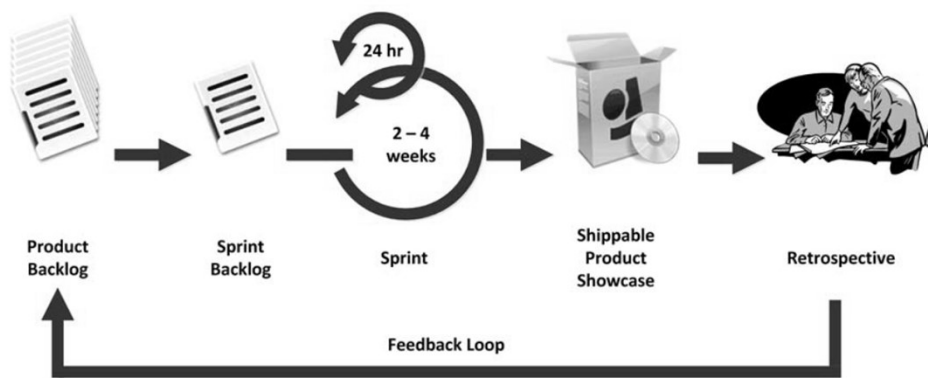
L'objectif est d'améliorer les affaires

Voir au-delà de quelques itérations

Adapté de « What is An AGILE Business Analyst? », par Steve Blais.
BA Times, 2014-01-21

9

Le cycle de vie agile, version SCRUM



Source: Agile Extension do the BABOK® Guide, Version 1.0
IIBA, Agile Alliance

10

Itérations et exigences

- Les approches agile prônent le développement d'une solution par *itérations* successives
 - Les itérations sont parfois nommées des sprints
- Chaque itération est centrée sur la réalisation d'une partie des exigences
- La portée du livrable final augmente à chaque itération en fonction des exigences réalisées

11

3 rôles définis dans le cadre SCRUM

- Propriétaire de produit / Product Owner
 - Le représentant des parties prenantes affaires
 - Responsable de la vision et de la direction du produit
 - Rédige et priorise les éléments du carnet de produit
- Maître de mêlée / Scrum Master
 - Responsable de l'adhésion à la méthode
 - Facilite la collaboration et la coordination
- Équipe de développement / Team
 - Responsable de développer et livrer le produit
 - Auto-organisée
- À éviter: Ajouter un analyste d'affaires comme interface entre le propriétaire de produit et le reste de l'équipe

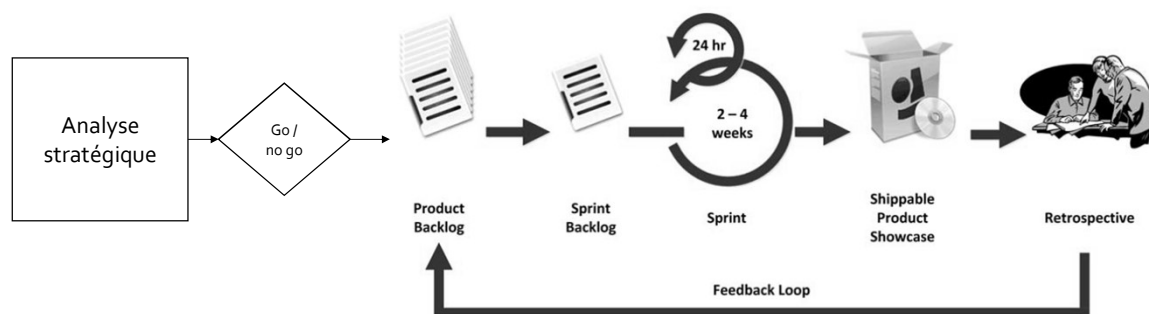
12

Le démarrage du projet

- Dans la plupart des entreprises, les décideurs doivent franchir certaines étapes pour formaliser l'existence du projet
- L'organisation veut s'assurer que le projet est aligné sur le plan stratégique
 - Les *critères de succès du projet* sont identifiés
- On élabore alors une *vision* pour le projet et la *portée de la solution* est développée
 - Un dossier de décision peut aussi être créé
- Perspective agile
 - Une version embryonnaire du carnet de produit peut être élaborée
 - Les règles et principes qui guideront la priorisation du travail devraient être établis en amont

13

L'analyse stratégique en contexte



Source: Agile Extension to the BABOK® Guide, Version 1.0
IIBA, Agile Alliance

14

Les artéfacts liés aux exigences

- Le carnet de produit
- Le carnet de sprint
- L'incrément de produit



15

Carnet de produit / Product backlog

- Le carnet de produit sert à définir la *portée du produit*
 - Il découle de décisions en amont concernant la direction du projet
 - Activités de l'analyse d'entreprise, telles que l'établissement d'objectifs et d'exigences d'affaires, énoncé de vision, portée de la solution
 - Il évolue au gré des revues du sprint
- Les items qui composent un carnet de produit sont appelés des PBI (Product Backlog Items)
 - Les PBI représentent, grosso modo, des *exigences des parties prenantes*
 - La forme la plus courante d'un PBI est un récit utilisateur

16

Priorités dans le carnet de produit

- Les PBI sont ordonnés dans le carnet de produit par priorité
 - Les priorités sont établies par le propriétaire de produit
 - En collaboration avec les autres parties prenantes et l'analyste d'affaires
 - Celles-ci découlent aussi de l'adéquation aux valeurs énoncées durant l'analyse stratégique (traçabilité en amont)
- Les priorités sont *évolutives et ajustées au gré des itérations*
- Différentes techniques de priorisation peuvent être utilisées et facilitées par l'analyste d'affaires
 - Voir BABOK 3.0 section 10.33

17

Contexte

- Les récits utilisateurs sont la forme la plus courante de PBI
 - Ils tirent leur origine des approches et méthodes agiles
 - Mais ils peuvent être utilisés dans d'autres types de projet
- Leur objectif est d'exprimer les attentes des utilisateurs dans leur langage
 - Dans un format *court et compact*
- Un « Epic » est un assemblage plus grossier, qui peut englober plusieurs récits utilisateurs

18

Récits utilisateurs

- À qui sont-ils destinés?
 - Aux parties prenantes, qui veulent une approche simple et surtout *conviviale* pour contribuer au carnet de produit
 - Aux utilisateurs, qui pourront articuler leurs *exigences détaillées* autour de ces récits
 - Au propriétaire de produit et au maître de mêlée, qui planifieront la priorité et la portée des itérations autour des récits afin d'assurer une *valeur ajoutée réelle* à chaque itération

Adapté de « User Stories - WHO Wants Them, WHAT Are They And WHY Use Them? »
par Ryan Hewitt, BA Times, 2013-03-23

19

Format

- Il existe plusieurs formats pour décrire des récits, mais tous se ressemblent
 - En tant que <qui>, je veux <quoi> afin de <pourquoi>
 - Afin de <obtenir un avantage> en tant que <qui>, je veux <quoi>
 - En tant que <qui>, <quand> et <où>, je veux <quoi> afin de <pourquoi>
 - En tant que <rôle>, je peux <quoi> afin de <obtenir un avantage>
- Donc, 3 éléments de base:
 - Le qui, donc un **rôle** (partie prenante impliquée)
 - Le quoi, donc une **action** (fonctionnalité)
 - Le pourquoi, donc un **avantage** (valeur ajoutée)

20

Exemples

- En tant que client de la banque, je veux compléter des transactions bancaires simples en mode libre-service
- En tant que cinéophile, je veux pouvoir acheter des billets sans devoir faire la file
- En tant qu'utilisateur d'une plateforme mobile, je veux pouvoir accéder à des résultats de sports à jour facilement
- En tant qu'employé, je veux soumettre ma feuille de temps dans les délais, sachant que celle-ci est exempte d'erreurs

21

Besoins et exigences

- Le récit utilisateur est parfois qualifié d'exigence intermédiaire
- L'accent est mis sur la « fonctionnalité »
 - Donc, les exigences *fonctionnelles*
 - La partie <quoi>
 - Et les utilisateurs
 - La partie <qui> ou <rôle>
- On peut aussi inclure une perspective sur le *non-fonctionnel*
 - Souvent exprimée avec des adjectifs ou des adverbes
 - C'est la partie du <comment>
 - « simple », « facilement », « exempte d'erreurs »
 - Les exigences non-fonctionnelles devront être développées plus tard

22

Critères INVEST pour un récit

- Inspiré de Bill Wake
 - Indépendant
 - **N**égociable et **N**égocié
 - « **V**aluable »
 - **E**stimable ou au moins **E**valuable
 - Suffisamment petit (« **S**mall ») ...**S**ommaire...**S**uccinct...**S**velte...
 - Testable

23

Récits utilisateurs vs Cas d'utilisation

Récits utilisateurs	Cas d'utilisation
Brève description d'une fonctionnalité	Séquence d'actions, plus ou moins détaillée
Format plus ouvert	Format plus rigide
Rapide et simple	Plus long et complexe
Les critères d'acceptations <i>utilisateurs</i> sont <i>implicites</i>	Les critères d'acceptations <i>fonctionnels</i> sont <i>explicites</i>
Correspond normalement à une seule itération de réalisation	Peut être réalisé en une itération ou plus
Réalisés par le propriétaire du produit en mode collaboratif	Réalisés par des analystes d'affaires en mode plus autonome
Limités au scénario le plus courant	Couvrent explicitement les scénarios secondaires

24

Avantages / inconvénients des récits

Par rapport aux méthodes dites « traditionnelles »

- AVANTAGES

- Très rapides et simples. Peu d'efforts requis
- Sujets à interprétation (aussi un inconvénient)
- Favorisent la collaboration utilisateur-développeur
- Permettent de décomposer un projet en « morceaux » plus simples
- Les itérations de réalisation basées sur les récits ajoutent toutes de la valeur
- Demandent peu de formation pour la rédaction

- INCONVÉNIENTS

- Difficiles à maintenir sur de grands projets
- Souvent perçus comme vagues et incomplets
- Sujets à interprétation (aussi un avantage)
- Difficiles à transformer en cas d'essai
- Les exigences non-fonctionnelles peuvent être omises
- Éventuellement, les détails devront être discutés

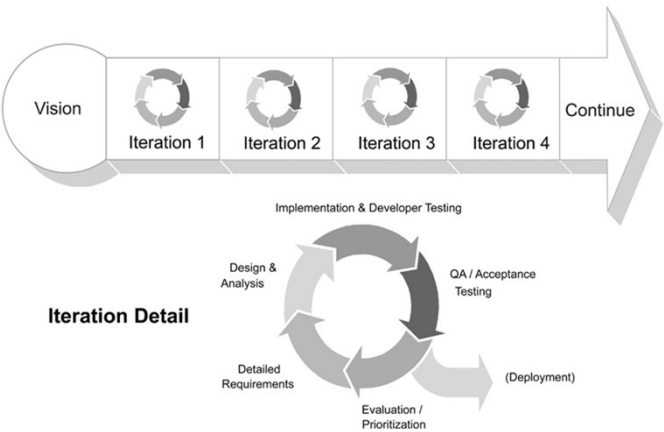
25

Cas d'utilisation vs récits, quelques citations

- « A story is a promise to have a conversation and a use case is the record of the conversation. If you think you need one. »
Jim Standley
- « A Use Case is a way of describing requirements. A User Story is a way of prioritizing work. »
Tim Wright

26

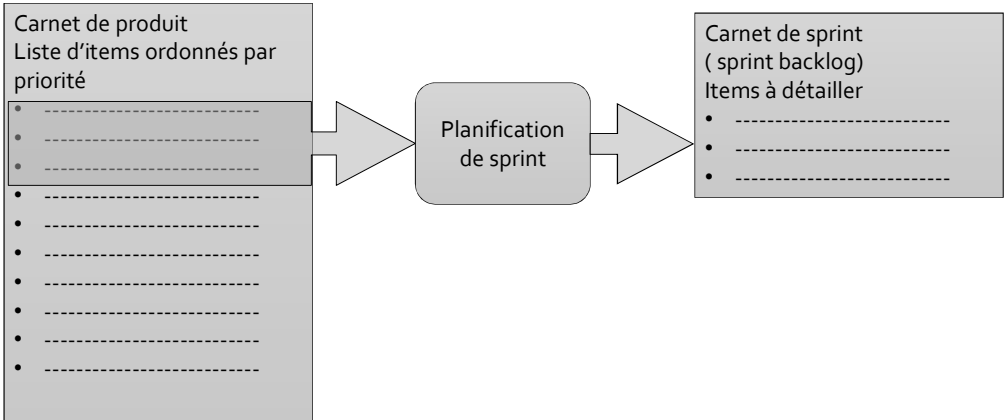
Sprints / itérations



Source: <http://scrumreferencecard.com/scrum-reference-card/>

27

Planification de sprint



28

Carnet de sprint / Sprint backlog

- Le carnet de sprint contient les items qui seront élaborés en exigences au *niveau adéquat de détail* au cours du sprint
 - Les principes de base pour des exigences agiles:
 - « Just in time »
 - On consacrera temps et effort à développer des exigences seulement qu'au moment où elles seront nécessaires
 - « Just enough »
 - On adaptera l'exhaustivité et le niveau de détail aux besoins du travail de développement
 - L'objectif étant d'éviter le gaspillage d'efforts
 - Les techniques d'élicitation « classiques » sont alors utilisées pour produire des exigences au bon niveau de détails
 - Les techniques collaboratives sont toutefois préférées
 - Un éventail complet de parties prenantes doit être impliqué

29

Élaboration des exigences

- À considérer pendant l'élaboration des exigences
 - Le niveau de détail requis des exigences (page précédente)
 - La traçabilité en amont vers les objectifs de l'organisation (valeur ajoutée)
 - L'inclusion d'exemples peut aider à rendre certains concepts plus concrets
 - Les exigences implicites dans les récits utilisateurs qui devront être explicitées
 - Particulièrement pour les exigences non-fonctionnelles
 - « Rapidement », « robuste », « aisément », « conformément à... », etc.
 - Les critères d'acceptation nécessaires aux essais

30

Exemple de récit développé

Récit: « En tant que gérant du magasin, je veux ajouter à l'inventaire les items qui sont retournés, afin de maintenir à jour mon inventaire »

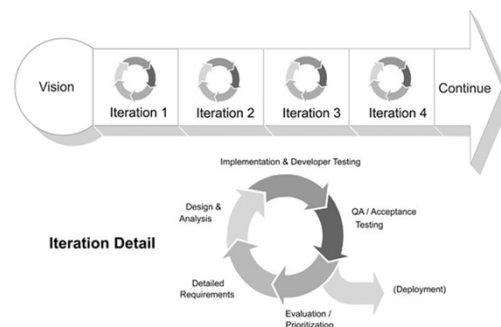
- Scénario 1: Les items remboursés devraient être retournés en inventaire
- En supposant que:
 - Un client a acheté un pull noir
 - J'ai déjà 3 pulls noirs en inventaire
 - Le client retourne le pull pour remboursement
- Donc
 - Je devrais avoir 4 pulls en inventaire
- Scénario 2: Les items remplacés devraient être retournés en inventaire
- En supposant que:
 - Une cliente a acheté une jupe bleue
 - J'ai 2 jupes bleues en inventaire
 - J'ai 3 jupes noires en inventaire
 - La cliente échange la jupe bleue pour une noire
- Donc
 - Je devrais avoir 3 jupes bleues en inventaire
 - Je devrais avoir 2 jupes noires en inventaire

Inspiré de la méthode « Behavior Driven Development »

31

Sprints / itérations

- Les exigences détaillées permettront à l'équipe de procéder aux étapes suivantes de l'itération
- Les « détails requis » sont en fonction de ces étapes subséquentes
- Voir le DoD page suivante



Source: <http://scrumreferencecard.com/scrum-reference-card/>

32

Incrément de produit / Product Increment

- C'est la somme des éléments du carnet de produit qui sont finis
- Cette liste est incrémentée à la fin de chaque sprint
- Les items ajoutés doivent rencontrer les critères propres à l'item
 - Définition de fini ou Définition de terminé / « Definition of Done » ou DoD
 - Le DoD renseigne habituellement sur:
 - L'approbation des items du backlog
 - Les cas de tests devant être écrits
 - L'état de complétion du code
 - Les tests devant être réussis
 - L'état de la documentation
 - Une (autre) DoD peut aussi s'appliquer aux exigences avant de procéder aux autres activités du sprint

33

Autres rencontres typiques

- Mêlée quotidienne / Daily scrum
 - Rencontre quotidienne pour discuter des priorités du travail en cours pendant une itération
- Réunion de planification d'un sprint / Sprint planning meeting
 - Rencontre en début de sprint pour discuter de la portée de l'itération
- Revue de sprint / Sprint review meeting
 - Rencontre en fin de sprint pour discuter des items qui ont effectivement été réalisés en cours de sprint
 - Le carnet de produit est mis à jour (ajouts, retraits, tri) au cours de ces revues
- Rétrospective du sprint / Sprint retrospective meeting
 - Rencontre en fin de sprint pour discuter des occasions d'amélioration continue (similaire aux leçons apprises)

34

À retenir

- Plusieurs cadres agile, Scrum étant le plus courant
- Différents artefacts pour documenter et gérer les exigences
 - Carnet de produit, carnet de sprint, récits utilisateurs, incrément de produit
- Le récit utilisateur est une technique courante pour développer les exigences dans un projet
 - L'accent est mis sur le qui, le quoi et le pourquoi
 - Les récits devront éventuellement être développés

35

Conclusion

- Les projets agiles ont besoin d'analyse d'affaires, car la réussite d'une solution sera toujours dépendante de bonnes exigences
 - Par contre, la définition de « bonnes » exigences est plus fluide
 - Les techniques pour les documenter doivent être adaptées
- Sur les projets agiles, les analystes d'affaires sont appelés à devenir plus généralistes
 - Par exemple, moins de besoins de maîtriser des techniques de modélisation, d'analyse ou d'élicitation avancées.
- Les organisations auront toujours besoin d'analystes d'affaires dans un rôle « traditionnelle » pour les activités autre que le développement logiciel

36

Questions?

Merci de votre participation

Clément Côté

<https://www.linkedin.com/in/clementcote/>

37

Références

- A Guide to the Business Analysis Body of Knowledge (BABOK), v 3.0, International Institute of Business Analysis, 2015
- Agile Extension to the BABOK® Guide, v 1.0, International Institute of Business Analysis, Agile Alliance
- Agile Project Management with Scrum, Ken Schwaber. Microsoft Press, 2004
- Agile Software Development with Scrum, Ken Schwaber, Mike Beedle. Prentice Hall, 2002
- « Three Ways Business Analysts Fit into the Agile Methodology », Tony Higgs, 2014-10-26. <http://www.blueprintsys.com/blog/three-ways-business-analysts-fit-into-the-agile-methodology/>
- « Role of Business Analysis in Agile », Ben Linders, 2016-03-10. <https://www.infoq.com/news/2016/03/business-analysis-agile>
- « What Is An AGILE Business Analyst », Steve Blais, 2014-01-21. <https://www.batimes.com/steve-blais/what-is-an-agile-business-analyst.html>
- « Business Analysts in Scrum », Roman Pitchler, 2010-03-09. <http://www.romanpitchler.com/blog/business-analysts-in-scrum/>
- « User Stories - WHO Wants Them, WHAT Are They And WHY Use Them? », Ryan Hewitt. 2013-03-26. <https://www.batimes.com/articles/user-stories-who-wants-them-what-are-they-and-why-use-them.html>
- « Scrum Reference Card », Michael James. <http://scrumreferencecard.com/scrum-reference-card/>

38