

# Méthode centrée utilisateurs et développement agile : une perspective « gagnant-gagnant » au service des projets de R&D

*Dominique Deuff*

Orange Labs  
2, avenue Pierre Marzin  
22307 Lannion cedex, France  
dominique.deuff@orange-  
ftgroup.com

*Mathilde Cosquer*

Orange Labs  
2, avenue Pierre Marzin  
22307 Lannion cedex, France  
mathilde.cosquer@orange-  
ftgroup.com

*Béatrice Foucault*

Orange Labs  
2, avenue Pierre Marzin  
22307 Lannion cedex, France  
béatrice.foucault@orange-  
ftgroup.com

## RESUME

Les services sur lesquels nous travaillons sont de plus en plus complexes et exigent un fort niveau de simplicité. Dans les projets R&D, l'ergonome intervient souvent, au début lors de la phase de conception, et après les développements pour valider le service par un test utilisateurs. La méthode que nous avons mise en place est une solution de mutuelle intégration de l'ergonomie et des développements agiles. Elle est basée sur la réalisation de test-utilisateurs courts qui permettent d'impliquer les utilisateurs en phase de développement. L'ergonome réalise ces tests en cycles coordonnés avec ceux du développement agile. Les résultats obtenus à chaque fin de cycles proposent des ajustements ergonomiques au même titre que les validations techniques peuvent impliquer des ajustements fonctionnels.

Cette méthode a été appliquée et validée dans un projet. Suite à la description de la méthode, l'article présente les retours d'expérience des membres du projet.

**MOTS CLES :** méthodes agiles, conception centrée utilisateurs, ergonomie, tests utilisateurs.

## ABSTRACT

Services on which we are working are more and more complex and require a high level of simplicity. In R&D projects, an ergonomist often acts at the beginning of the project, during the design phase. Or he acts after the development phase by applying a user test to validate the

service. The method that we have set up is a solution to integrate user-centered design and the agile developments to one another. It is based on short user tests which enable users to intervene during the development phase. The ergonomist realises user tests in cycles that match agile development ones. Results gained at the end of a cycle suggest ergonomics adjustments for the same reasons that technical validations can induce functional adjustments.

This method was applied and validated on a project. Following the description of the method, this paper introduces project members experience feedbacks.

**CATEGORIES AND SUBJECT DESCRIPTORS:** D.2.10 Design : Methodologies & Representation. H.5.2 User Interfaces : User-centered design.

**GENERAL TERMS:** Design, Human factors

**KEYWORDS:** agile methods, user center conception, ergonomity, user tests.

## INTRODUCTION

Parmi les nouveaux services sur lesquels nous travaillons à Orange Labs, nombreux se basent sur des interfaces de plus en plus convergentes (convergence des services via une multiplicité de dispositifs) et par conséquent, sur des services de plus en plus complexes à concevoir. En parallèle, l'exigence en matière de services simples d'utilisation et proches des besoins des clients, est également un facteur que l'entreprise Orange souhaite constamment privilégier.

Dans les projets R&D que nous menons habituellement, l'ergonomie intervient généralement avant les développements et/ou en fin de projet par l'application de la méthode de Conception Centrée Utilisateurs (CCU). Cette

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

IHM'2010, September 20-23, 2010, Luxembourg, LU

Copyright © 2010 ACM ISBN 978-1-4503-0410-8/09/2010... \$10.00

méthode a pour fondement la participation fréquente et itérative de l'utilisateur réel tout au long d'un projet de conception [2].

Rendre simple d'utilisation un système nouveau et complexe nous a amenés à vouloir impliquer l'utilisateur de façon plus récurrente sur ces types de projets ; le but étant de mieux appréhender la compréhension des utilisateurs face à ces nouveaux services en cours de conception, pour finalement proposer un service qui tienne compte des connaissances des utilisateurs et de leurs comportements face à ces nouvelles fonctions et ces nouveaux modes d'interaction. Il semblait donc tout naturel que la CCU puisse se poursuivre durant les développements.

Parallèlement, depuis quelques années, les méthodes agiles (notamment SCRUM) se sont répandues dans les équipes des Orange Labs. Ce sont des modèles de développement itératif et incrémental qui visent à répondre au mieux aux besoins exprimés par les demandeurs, en offrant une grande réactivité par rapport à leurs demandes. Ces méthodes sont unifiées autour d'un manifeste [3] mettant en avant quatre valeurs fondamentales : l'importance des interactions et des personnes, la mise à disposition d'un produit toujours opérationnel, une collaboration étroite avec les demandeurs et l'acceptation d'une réactivité face au changement. Grâce à ces valeurs, les méthodes agiles permettent une rapidité dans le développement d'un service, et une souplesse pour appliquer au produit les adaptations qui apparaissent nécessaires à réaliser sur le service en cours de développement, tout en visant une satisfaction des demandeurs, représentants des utilisateurs.

Nous avons donc étudié les relations possibles entre la CCU et les méthodes agiles afin que l'intégration mutuelle de ces méthodes apporte une valeur ergonomique plus importante aux services développés.

Sur la base d'études et d'expériences issues de la littérature, cet article présente, dans une première partie, le cadre de travail permettant le rapprochement de la méthode CCU avec les méthodes agiles. Puis, dans une deuxième partie, nous décrivons notre méthode qui a été mise en pratique lors du développement du prototype d'un service Orange. La troisième partie introduit le retour de cette expérience.

#### **CADRE DE TRAVAIL INTEGRANT CCU ET METHODES AGILES**

Depuis la discussion entre K. Beck et A. Cooper [19] (cité par [8]), beaucoup de travaux dans ce domaine ont été menés. Certains concernent des retours d'expérience, d'autres des propositions d'intégration des deux méthodes, et d'autres encore présentent des évaluations de ces intégrations.

De nombreux articles ont validé le rapprochement possible de la CCU et des méthodes agiles et noté les fondements communs qui permettent leur intégration [6] [10] [18] [19] (cité par [8]) [24] [25]. Ainsi, [6], [8] et [12] soulignent un premier point convergent de ces deux méthodes : elles basent toutes deux la conception sur des cycles itératifs. Un second point commun est leur volonté de répondre aux besoins de l'utilisateur. À cette différence que les méthodes agiles répondent aux besoins du demandeur (représentant des utilisateurs). Elles n'impliquent pas directement l'utilisateur réel, et n'intègrent pas l'utilisabilité dans leurs évaluations. Aussi, si au sortir d'une méthode agile, un service est correctement développé, la méthode agile ne peut garantir pour autant que ce service soit utilisable par les utilisateurs finaux [1] [6] [10] [13] [14] [18] [21].

Intégrer la CCU et une méthode agile vise donc à impliquer l'utilisateur réel dans le processus de développement. Ceci n'est pas sans difficultés comme le notent [8], [11] et [17].

#### **Difficultés et limites d'intégration**

La CCU est un processus de conception qui implique l'utilisateur de façon itérative. À chaque itération, l'ergonome réalise trois étapes principales. L'une vise à prendre des données sur le terrain, à connaître les utilisateurs et leurs activités, pour en définir leurs besoins. La deuxième, à partir de ce travail préliminaire, offre le moyen de représenter les utilisateurs (sous forme de personas par exemple), de construire une vision globale du service à réaliser pour en concevoir les écrans avant les développements. La dernière consiste à évaluer la conception développée afin de détecter les gros problèmes d'usage.

Le fait qu'avant le développement, la CCU nécessite un temps de recherche et de compréhension du contexte d'utilisation pour concevoir des écrans d'interface est une des difficultés d'intégration. En effet, l'application de cet ensemble de tâches demande du temps dont la méthode agile ne permet pas de disposer. Lors de l'intégration à une méthode agile, la CCU se voit donc contrainte de répartir différemment les tâches qui constituent ces trois étapes.

La majorité des articles constatent que l'intégration des deux méthodes ne peut se faire sans une phase d'étude avant le commencement des développements afin de disposer d'une vision générale et de préparer globalement la conception. La priorisation des fonctions à développer et le découpage des travaux en cycles s'en trouvent ainsi facilités au début de développement. Néanmoins, afin de ne pas déroger aux méthodes agiles, cette phase est limitée en temps par rapport à ce que propose la CCU. Suivant les articles, la durée de cette phase peut varier, mais

il semble souvent établi qu'elle ne doive pas excéder la durée d'une itération de développement.

Cette phase étant courte, les ergonomes se limitent souvent à un travail minimal d'étude préparatoire (étape 1 de la CCU) afin d'esquisser les écrans des premières fonctions qui seront développées. Ceci a pour conséquence que bien des tâches de la CCU sont effectuées dans les itérations :

- recherches sur l'utilisateur, sur son activité, ou sur son contexte [12] [22] [24],
- préparation en amont des écrans qui seront développés dans les itérations futures [5] [7] [11] [12] [13] [20] [22] [23] [24]. L'ergonome est généralement en avance d'un ou deux cycles sur les développements. Certains projets précisent que ces écrans sont évalués sur « papier » par des utilisateurs avant d'être fournis aux développeurs [5] [6] [12] [21] [23] [24],
- évaluations de la version du service livré en fin d'itération précédente. Certains auteurs spécifient que ces évaluations sont faites avec la participation de vrais utilisateurs [4] [5] [6] [10] [11] [13] [21] [25]. Dans certains projets, le nombre de testeurs est réduit à moins de 4 utilisateurs par cycle [5] [11]. D'autres appliquent des tests « classiques » d'environ 12 testeurs, mais ne le font pas sur l'ensemble des cycles [10] [13] [25]. Dans tous les cas, les résultats sont fournis dans le cycle suivant celui de l'évaluation.

### Modèles existants

Divers modèles d'intégration ont été proposés dans la littérature. Il en ressort que pour l'ensemble de ces modèles, la majorité des tâches de la CCU (recherche d'information, conception et évaluation) sont réalisées dans les itérations de développement. Aussi, la différence entre les modèles se joue généralement au niveau de la phase amont aux développements, phase qui offre le moyen d'avoir une vision plus ou moins globale de la conception à réaliser.

Ainsi, le modèle de Sy [22], modèle largement applicable [12] et appliqué dans l'industrie, décrit un cycle 0, équivalent en durée aux autres cycles. Il permet aux ergonomes de faire les recherches générales préalables et de commencer la conception des écrans qui seront développés dans les premiers cycles.

Le modèle de Beyer et al. [5] met d'avantage l'accent sur la phase qui précède les développements. Elle est décrite de façon précise, et est estimée à environ 8 semaines de travail ce qui est plus important que la durée d'un cycle. Elle permet un meilleur approfondissement de la connaissance de l'utilisateur et de son contexte, mais n'insiste pas sur la tâche de conception.

Le modèle de Memmel et al. [18] renforce également le travail préliminaire au codage par l'introduction de deux phases :

- la phase « initial requirements up-front » identique aux modèles précédents. Elle vise à construire des scénarios d'usage focalisés sur les utilisateurs, les tâches et les interactions, en s'appuyant sur des données issues d'utilisateurs.
- la phase « initial conceptual » qui cherche à produire une maquette fonctionnelle dont les architectures technique et de l'information ont été conjointement pensées. Cette phase semble accentuer le travail préparatoire à la conception sans pour autant définir complètement les écrans de l'interface.

Le modèle de Ambler [4], indique également la nécessité d'avoir une étape préalable pour la construction d'une vision globale où le périmètre fonctionnel et les architectures technique et de l'information coïncident. Par contre, la conception des écrans se déroule dans la phase de développement. Les interfaces graphiques sont codées comme toute autre partie du système, lors des itérations et sans étude approfondie préalable, ni pré-tests.

Dans le modèle d'Adikari et al. [1], il n'y a pas de phase amont. L'analyse du contexte, de l'activité de l'utilisateur préalable à la conception et au codage se réalise à chaque itération, et non en tout début de projet.

Il nous semble que le fait d'avoir une phase réduite en amont et la réalisation de la majorité des tâches de la CCU durant les développements amène, à plusieurs constats :

- il n'existe pas de description des écrans avant les développements. Ceci rend difficile l'évaluation de l'ergonomie puisque le contexte global ne pourra être présenté aux utilisateurs lors de tests. Cela peut entraîner des coûts de modifications plus importants lors des développements. En effet, la détection des erreurs d'utilisabilité tôt dans le processus, évite par la suite de forts changements d'orientation qui s'ils étaient appliqués durant les développements seraient très coûteux [16].
- au fur et à mesure des itérations, l'ergonome n'a pas le temps de prendre du recul sur le service. Et, [7] et [15] constatent au final que les ergonomes observés demandent à disposer tous les 3 cycles, d'un cycle dédié à la cohérence globale,
- un ergonome seul ne peut réaliser l'ensemble de ces tâches, car comme le précisent Williams et Ferguson [24], réaliser les tâches de conception et d'évaluation dans une même itération est difficile à gérer par une seule personne.

Comme nous l'avons vu, la plupart des papiers privilégient l'adaptation de la CCU aux méthodes agiles. Ainsi, les modèles que nous venons de décrire proposent une phase réduite préalable au développement pour comprendre le contexte d'utilisation, et essayent d'intégrer le gros de la conception et des évaluations des écrans dans les cycles itératifs.

Pourtant, [4] [6] [8] [15] et [18] mettent l'accent sur l'importance d'une intégration partagée, donc d'une adaptation des deux parties. Nous avons ainsi voulu conserver dans les deux méthodes les éléments qui nous semblaient importants. C'est pourquoi, nous présentons dans ce qui suit une méthode dans laquelle nous avons pris le parti de donner une place importante à la conception avant les itérations de développements pour conserver les principaux bénéfices de la conception de type CCU, et d'adapter la CCU aux méthodes agiles durant la phase de développement pour conserver le rythme d'une production itérative rapide.

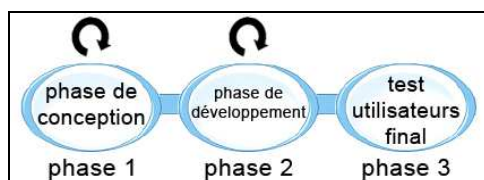
#### **NOTRE METHODE : INTEGRATION MUTUELLE DE LA CCU ET DES DEVELOPPEMENTS AGILES**

Pour présenter notre proposition, nous faisons l'hypothèse que les itérations font 3 semaines.

##### **Préserver une phase de conception en amont**

De manière générale, les projets sur lesquels nous travaillons, sont découpés en trois phases, la conception, le développement et un test utilisateurs final (Figure 1). L'ergonome, dans ces projets, intervient généralement sur la première et la dernière phase, mais peu durant la période de développement. Ce qui limite les itérations et la prise en compte du retour des utilisateurs car la phase de conception est parfois courte et les utilisateurs n'interviennent pas en phase de développement.

La méthode que nous proposons tient compte de ce découpage. Elle ne vise pas à supprimer la phase de conception, et recommande de conserver le test utilisateur de fin de projet. Elle a pour objectif de permettre à l'ergonome de poursuivre l'application de la CCU à travers des tests utilisateurs effectués régulièrement dans la phase de développement pour impliquer les utilisateurs réels tout au long du projet.



**Figure 1 :** découpage des projets en 3 phases

Le poids d'une conception initiale de type CCU se trouve ainsi maintenu en dehors des développements contraire-

ment à ce que nous avons pu noter dans l'existant. Ce choix est fondamental dans le cadre de la CCU.

Il en résulte, qu'au démarrage du processus de développement le projet dispose d'écrans minima qui permettent à l'ergonome d'avoir une vision globale du service. Ces écrans peuvent être plus ou moins aboutis en fonction du type de projet, mais ils doivent permettre un premier test utilisateurs cohérent dans son contexte et dans sa logique d'utilisation, dès la première version du service, même si seules quelques fonctions sont développées.

Ceci implique que plus il y a d'itérations en phase de conception, moins les tests utilisateurs de la phase de développement feront remonter d'importants problèmes d'utilisabilité. Les problèmes identifiés seront principalement focalisés sur les éléments difficilement observables en phase de conception, et étant minimes, ils seront plus facilement négociables.

Les deux parties suivantes décrivent comment nous préconisons d'intégrer les utilisateurs dans les itérations de développements agiles en conservant les avantages du rythme des itérations de l'agilité.

##### **Pré-requis en début de phase de développement**

Comme de nombreux papiers, tels que [2] [6] [10] [13] [15] [18] [20] [21] [25], nous recommandons que l'ergonome soit intégré à l'équipe agile pour partager les informations, et pour transformer les données issues de tests utilisateurs en recommandations de conception.

C'est-à-dire que l'ergonome est un membre de l'équipe agile au même titre que les développeurs. Par contre, il réalise des tâches autres que du codage, conservant ainsi sa spécificité et ses compétences.

Kolmann et al. [15] indiquent que l'ergonome doit comprendre la philosophie des méthodes agiles pour s'intégrer. Dans l'idée de conserver un équilibre entre « technologie » et « utilisabilité », il nous semble qu'inversement, les développeurs doivent saisir l'implication de la réalisation régulière de tests utilisateurs. Il est donc important que chacun :

- comprenne ce qu'est la méthode agile : développements sur des cycles réguliers, livraison en fin de cycle, ajustement des plannings en fin de cycle, etc.
- comprenne ce qu'est un test utilisateurs : but de l'évaluation, évaluation faite avec de vrais utilisateurs, etc. car les contraintes de développements pour un test utilisateurs ne sont pas les mêmes que celles de développements destinés à une démonstration. Elles sont plus fortes. Il faut éviter qu'un code trop instable empêche la réalisation de tests utilisateurs [9].

- prenne en compte les contraintes des tests utilisateurs dans la priorisation des fonctionnalités. En effet, la répartition des fonctionnalités ne doit plus se faire seulement par priorité des fonctions utiles du point de vue du demandeur.
- s'implique dans les scénarios de tests pour comprendre les éléments nécessaires à leur mise en œuvre. Effectivement, lors des tests, les utilisateurs ont besoin d'une vision globale du service, même si seule une partie de l'application est interactive.
- soit prêt à s'ajuster aux aléas rencontrés lors des développements. L'importance de l'implication de tous les membres est un élément également relevé par [8].
- adhère à la méthode et soit à l'écoute réelle des retours utilisateurs.

### Phase de développements

Intégrer l'ergonome dans les cycles itératifs, c'est lui permettre, à chaque cycle, d'apporter sa contribution au projet ; c'est permettre aux développeurs et à tous les

membres de l'équipe de continuer à être à l'écoute de l'utilisateur final.

L'ergonome contribue à apporter la voix de l'utilisateur et à faire en sorte que cette voix soit entendue lors des orientations prises pour l'amélioration du service. Pour cela, nous proposons que le travail de l'ergonome consiste, à chaque itération, à évaluer la version de l'application qui vient d'être livrée. Les évaluations se basent sur des tests utilisateurs réalisés sur 4 jours. Ils sont volontairement courts pour éviter un décalage entre les retours utilisateurs d'une version et leur intégration dans la version suivante.

Ainsi, comme le montre la Figure 2, la version 1 de l'application est évaluée par l'ergonome lorsque les développeurs s'attachent à développer la version 2 de l'application. Puis la version 2 est évaluée au début du 3e cycle de développement et ainsi de suite jusqu'à la fin du projet.

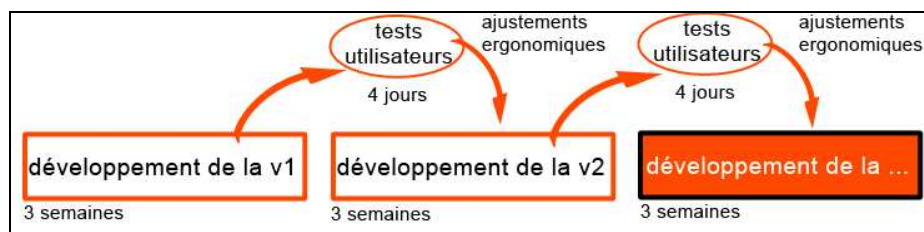


Figure 2 : des cycles itératifs qui mettent l'utilisateur au centre du développement

À chaque cycle, l'ergonome applique la méthode des tests utilisateurs bien connue des ergonomes à la différence que, dans notre proposition :

- le nombre d'utilisateurs par test est restreint à 4-5 personnes.
- parmi les utilisateurs, nous proposons qu'une partie (au moins 2) intervienne à chaque cycle pour évaluer le suivi des modifications effectuées dans le but d'éviter les régressions. Tandis que les autres ne participent qu'à un test d'un cycle afin de valider la simplicité d'acquisition de l'outil à chaque cycle. Cette proposition peut être une réponse à la question de la non-régression soulevée par Blonkvist [6], que Schwartz et al. [21] souhaitent réguler par l'usage de tests automatiques.

Durant sa tâche d'évaluation, l'ergonome réalise :

- un jour d'expertise pour appréhender l'application et pour identifier les problèmes qu'elle peut poser au regard de l'utilisabilité.
- deux jours de tests.

- un jour et demi d'analyse et d'écriture de recommandations.

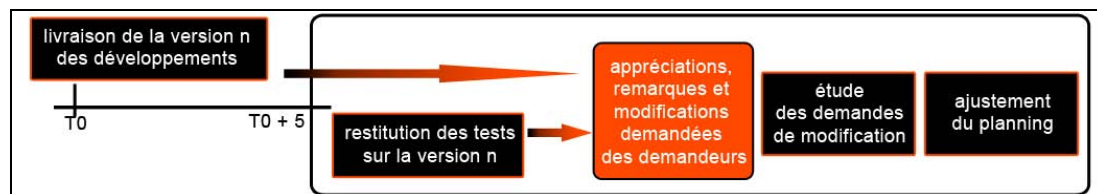
En méthode agile, à l'issue d'une itération, la livraison d'une version du service se déroule généralement sur une demi-journée. Durant cette étape, l'équipe agile présente l'état de l'application aux demandeurs. Les demandeurs formulent leurs appréciations, remarques et modifications qui peuvent avoir des implications sur le calendrier. Après analyse des demandes d'ajustement par l'équipe agile seule, l'ensemble de l'équipe projet discute pour ajuster le planning du travail restant à faire.

Dans notre proposition, au moment de la livraison, l'ergonome n'a pas encore évalué la version proposée. Il est donc nécessaire de décaler les discussions pour l'ajustement du planning, et de prévoir une seconde demi-journée de restitution qui introduit les retours utilisateurs et les recommandations ergonomiques issues des tests. Comme Detweiler [9], il nous semble important que cette restitution se fasse devant les développeurs et les demandeurs pour que l'ensemble de l'équipe possède les mêmes éléments de connaissance.

Ainsi, la Figure 3 représente les étapes à programmer entre la livraison d'une nouvelle version de l'application reçue à la fin d'une itération, et la réunion d'ajustement du planning avec les demandeurs :

- livraison de l'application par les développeurs à l'ergonome et aux demandeurs (T0),
- restitution des éléments venant des tests utilisateurs aux développeurs et aux demandeurs (T0 + 5 jours),

- étude des modifications demandées par les demandeurs entre les membres de l'équipe agile (ergonome et développeurs) pour évaluer les modifications à prendre en compte et les ajustements de planning (T0 + 5 jours),
- adaptation du planning entre l'ensemble des représentants du projet (T0 + 5 jours).



**Figure 3 :** succession des réunions pour la prise en compte des modifications à réaliser suite à la livraison des développeurs et à la restitution de l'ergonome.

Pour disposer d'un bon équilibre entre éléments techniques et utilisabilité, il est important de rester vigilant à ce que la restitution du travail d'ergonomie soit toujours prévue et soit considérée comme équivalente à la livraison que font les développeurs. Ceci afin de maintenir une égalité de pouvoir entre ergonome et développeurs [8], et éviter que les retours utilisateurs ne soient pas pris en compte.

Dans la seconde partie d'une itération, l'ergonome peut avoir diverses tâches comme la modification d'écrans, le soutien aux développeurs concernant l'interface et l'ajustement des scénarios des tests de l'itération suivante. Contrairement à ce que nous avons constaté dans l'existant, l'ergonome n'a pas ici comme tâche principale de créer des écrans durant les développements. L'axe majeur de son travail vise à l'amélioration de la conception définie en amont des développements, à partir des retours réguliers des utilisateurs.

De plus, les ajustements décidés suite aux tests utilisateurs réalisés dans une itération n (sur la version n-1 du service) sont injectés dans cette même itération n et non dans l'itération n+1 comme vu dans les modèles existants. Ce qui fait que la version n + 1 livrée contient les ajustements à la fois techniques et ergonomiques réalisés sur la version n.

### Phase 3 : test utilisateurs final

En fin de projet, nous préconisons, suite à la dernière itération, que le test utilisateurs « classique » final soit conservé afin d'effectuer une validation complète de l'application finale. Comme proposé par Memmel et al. [18], ce test final peut également permettre d'identifier les nouvelles fonctionnalités qui pourront être envisagées dans une version future de l'application.

Nous avons mis en œuvre cette méthode dans un projet Orange ayant pour objectif de refondre un service de communication en ligne. La partie suivante présente les retours des membres du projet par rapport à cette méthode.

### RETOURS D'APPLICATION DE LA METHODE A UN PROJET DE R&D

Le projet ayant appliqué la méthode comportait 3 mois de développements. Une phase de conception amont avait permis de disposer des écrans « fil de fer » du service à développer.

Durant la période de développement, nous avons réalisé 3 itérations de tests qualitatifs en rencontrant à chaque fois 4 utilisateurs. Deux de ces utilisateurs revenaient à chaque test. Très participatifs, ils étaient enthousiastes de voir que leurs remarques étaient prises en compte dans le développement du service. Nous avons ensuite conclu par un test final qui nous a permis de valider notre prototype auprès de 12 utilisateurs (dont les 2 testeurs récurrents).

Il est arrivé que les membres de l'équipe agile assistent aux tests utilisateurs par l'intermédiaire d'un retour vidéo présent dans la salle de tests ou via les enregistrements. Ceci leur a apporté un « feed back » direct et fidèle de l'utilisation de l'application et a renforcé leur compréhension des résultats des tests utilisateurs.

Cette méthode nous a permis d'affiner et d'enrichir le concept de service et les interfaces définis en début de projet. Pour exemple, il s'est avéré que les premiers tests utilisateurs ont mis en évidence un décalage entre les priorisations des utilisateurs, en termes de fonctionnalités cœur de service, et celles de l'équipe projet. Le service

ainsi généré est de haute qualité en respectant fidèlement l'évolution des besoins des clients.

Compte tenu des contraintes temporelles fortes, une communication fluide entre les membres de cette équipe a été une condition de réussite. Aussi noté par [7], [8], [17] et [24], l'ergonome ressort motivé de cette expérience par sa meilleure intégration dans l'équipe. Et, les développeurs ont mieux accepté les fortes modifications parfois demandées sur l'application, par leur compréhension des résultats des tests utilisateurs. L'équipe projet s'est sentie ainsi impliquée dans une même dynamique d'innovation centrée sur les utilisateurs, dans une véritable complémentarité des compétences.

Pour conclure, l'utilisation de cette méthode dans le cadre de notre projet de R&D nous a apporté de réels atouts : partage des connaissances, simplicité, efficacité, économie et feedbacks utilisateur réactifs. De plus, nos MOA (Maître d'Ouvrage) ont apprécié d'obtenir au plus tôt des remontées d'expérience utilisateur, ce qui donne davantage de poids au service livré.

## CONCLUSION ET PERSPECTIVES

Les études menées par Adikari et al. [1] ou Hussain et al. [14] soulignent l'intérêt d'un processus intégrant méthodes agiles et CCU. Elles concluent sur l'amélioration de la qualité de conception des applications ayant suivi ce processus. L'une des études précise que l'approche permet d'obtenir une application plus simple à acquérir, plus simple à utiliser et qui requiert moins de support technique. L'autre étude note une amélioration de l'utilisabilité de l'application et une meilleure satisfaction des utilisateurs finaux.

Nous avons proposé une nouvelle méthode d'intégration de la CCU à une méthode agile. Les éléments importants de cette méthode, pour une meilleure implication de l'utilisateur, sont :

- la conservation d'une phase de conception minimale avant les développements,
- l'intégration de l'ergonome dans l'équipe agile,
- l'application itérative de tests utilisateurs dans la phase de développement, dont les retours sont injectés dans la même itération que l'itération d'évaluation (pas de décalage),
- la gestion de la non régression par la participation d'utilisateurs nouveaux et récurrents,

Cette méthode est présentée sous forme d'un document qui décrit le processus générique de mise en œuvre afin de pouvoir être déployée sur divers projets de l'entreprise. Il est bien entendu que suivant la configuration des projets, des ressources associées aux projets et de la précision de l'objectif du projet, la situation réelle peut diverger de la théorie proposée dans le document.

La méthode a été appliquée et validée par un projet. L'expérience ayant été positive, nous souhaitons appliquer la méthode sur de nouveaux projets. Le but de cette deuxième étape serait de réaliser des observations et des entretiens dont une analyse permettrait d'affiner le processus.

Nous avons également l'idée d'introduire des questionnaires courts (SUS, Attrakdiff, NasaTLX ou les cartes « desirability toolkit ») en fin de tests utilisateurs, afin de faire un suivi de l'évolution de certains paramètres tels l'utilisabilité, l'attractivité, la satisfaction, l'esthétisme, etc. Ces outils nous semblent robustes et par conséquent pertinents à utiliser dans un contexte itératif. De plus, ils correspondent bien à l'esprit agile par leur facilité d'application en fin de test utilisateurs, et par le côté automatique de leur utilisation.

Nous prévoyons également d'utiliser le tableau de Sy [22] pour la gestion des remontées utilisateurs. Il nous semble que ce tableau permettra de suivre l'avancée des corrections et adaptations relatives à l'utilisabilité. Nous souhaitons l'adapter afin de disposer d'un historique des évolutions dans le cas où un second ergonome viendrait à participer au projet.

## REMERCIEMENTS

Les auteurs remercient l'ensemble des membres de l'équipe et le projet de recherche qui lui est associé pour avoir contribué à la validation de cette méthode.

## BIBLIOGRAPHIE

1. Adikari, S.; McDonald, C., Campbell, J. Little Design Up-Front : A Design Science Approach to Integrating Usability into Agile Requirements Engineering. In *proceedings of HCI 1*, Vol. 5610, Springer, 2009, pp. 549-558.
2. AFNOR ISO 13407. *Processus de conception centrée sur l'opérateur humain pour les systèmes interactifs*, 1999.
3. Agile Manifesto, 2001. Disponible en juillet 2010 à l'adresse <http://agilemanifesto.org/>
4. Ambler, S.W. Tailoring Usability into Agile Software Development Projects. E. Law et al. (eds.), *Maturing Usability*, chapter 4, Springer, London, 2008, pp. 75-95.
5. Beyer, H., Holtzblatt, K., Baker, L. An Agile Customer-Centered Method : Rapid Contextual Design. In *XP/Agile Universe 2004*, LNCS 3134, Springer Verlag, Heidelberg, 2004, pp. 50-59.



6. Blomkvist, S. Towards a Model for Bridging Agile Development and User-Centered Design. In *Human-Centered Software Engineering — Integrating Usability in the Software Development Lifecycle*, Human-Computer Interaction Series, vol. 8, Springer Netherlands, 2005, pp. 219-244.
7. Budwig, M., Jeong, S., Kelkar, K. When user experience met agile : a case study. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems CHI 2009*, ACM, New York, 2009, pp. 3075-3084.
8. Chamberlain, S., Sharp, H., Maiden, N. Towards a framework for integrating agile development and user-centred design. In: *Abrahamsson, P., Marchesi, M., Succi, G. (eds.) XP 2006*, LNCS, vol. 4044, Springer, Heidelberg , 2006, pp. 143-153.
9. Detweiler, M. Managing UCD within Agile Projects. In *ACM Interactions* (May-June, 2007), 2007, pp. 40-42.
10. DÜchting, M., Nebe, K. and Zimmermann, D. Incorporating User Centered Requirement Engineering into Agile Software Development. In *Proceedings of the International HCI 2007* (July 23--27, 2007, Beijing), LNCS 4550, vol. 1. Springer Verlag, Heidelberg, 2007, pp. 240-249.
11. Federoff, M. and Courage, C. Successful User Experience in an Agile Enterprise Environment. In *Proceedings of the Symposium on Human interface 2009 on Conference universal Access in Human-Computer interaction. Part I: Held As Part of HCI international 2009* (July 19 - 24, 2009, San Diego), vol. 5617. Springer-Verlag, Heidelberg, 2009, pp. 233-242.
12. Fox, D., Sillito, J., Maurer, F. Agile methods and User-Centered design: How these two methodologies are being successfully integrated in industry. In *proceedings of Agile 2008*, 2008, pp. 63-72.
13. Hussain, Z., Milchrahm, H., Shahzad, S., Slany, W., Tscheligi, M., Wolkerstorfer, P. Integration of extreme programming and user-centered design: Lessons learned. In *P. Abrahamsson, M. Marchesi, and F. Maurer, (eds.), XP 2009*, LNBIP, vol. 31, Springer, 2009, pp. 174-179.
14. Hussain, Z.; Slany, W.; Holzinger, A. Current state of agile user-centered design: A survey. In *proceedings of the 5th symposium of the workgroup human-computer interaction and usability engineering of the Austrian computer society* (November 9-10, 2009, Linz, Austria), Springer, Berlin, 2009, pp. 416-427
15. Kollmann, J., Sharp, H., Blandford, A. The importance of Identity and Vision to user experience designers on agile projects. In *proceedings of Agile 2009*, 2009, pp. 11-18.
16. Marcus, A. Return on investment for usable UI design. *User Experience*, Vol. 1, No. 3, 2002, pp. 25-31.
17. McInerney, P., Maurer, F. UCD in Agile Projects : Dream Team or Odd Couple ? In *Interactions* (November-December 2005), Vol. 12, No. 6, ACM Press, New York, 2005, pp. 19-23.
18. Memmel, T., Gundelsweiler, F., Reiterer, H. Agile human-centered software engineering. In *Proceedings of the 21st British CHI Group Annual Conference on HCI 2007: People and Computers XXI: HCI...but not as we know it*, (September 03-07, 2007, University of Lancaster), British Computer Society, Swinton, 2007, pp. 167-175.
19. Nelson. E. Extreme Programming vs. Interaction Design, 2002. Disponible en juillet 2010 à l'adresse <http://unthinkingly.com/extreme-programming-vs-interaction-design/>
20. Nielsen, J., Agile User Experience Projects. 2009. Disponible en juillet 2010 à l'adresse <http://www.useit.com/alertbox/agile-user-experience.html>
21. Schwartz L., Vergnol L., Gronier G., Vagner A., Altenburger T., Battisti S. Comment concilier agilité et conception centrée utilisateurs dans un projet de développement ? In *proceedings of IHM 2009, 21ème Conférence Francophone sur l'Interaction Homme-Machine*, (October 13-16, 2009, Grenoble), 2009, pp. 337-340.
22. Sy, D. Adapting Usability Investigations for Agile User-centered Design. In *Journal of Usability Studies*, Vol. 2, No. 3, 2007, pp. 112-132.
23. Ungar, J., White, J. Agile user centered design: enter the design studio - a case study, In *proceedings of CHI '08 extended abstracts on Human factors in computing systems* (April 05-10, 2008, Florence, Italy), 2008, pp. 2167-2177.
24. Williams, H., Ferguson A. The UCD Perspective: Before and After Agile. In *Proceeding of AGILE 2007* (August 13-17, 2007), IEEE Computer Society, Washington, 2007, pp. 285-290.
25. Wolkerstorfer, P., Tscheligi, M., Sefelin, R., Milchrahm, H., Hussain, Z., Lechner, M., Shahzad, S. Probing an agile usability process. In *proceedings of CHI 2008, extended abstracts on human factors in computing systems*,. ACM Press, New York, 2008, pp. 2151-2158.