



HAL
open science

Early Incremental User Testing Design Approach Validation for Satellite Command Center's Application

Jonathan Tolle

► **To cite this version:**

Jonathan Tolle. Early Incremental User Testing Design Approach Validation for Satellite Command Center's Application. Cristian Bogdan; Kati Kuusinen; Marta Kristín Lárusdóttir; Philippe Palanque; Marco Winckler. 7th International Conference on Human-Centred Software Engineering (HCSE), Sep 2018, Sophia Antipolis, France. Springer International Publishing, Lecture Notes in Computer Science, LNCS-11262, pp.311-318, 2019, Human-Centered Software Engineering. 10.1007/978-3-030-05909-5_19 . hal-02270723

HAL Id: hal-02270723

<https://inria.hal.science/hal-02270723v1>

Submitted on 26 Aug 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Early incremental user testing design approach validation for satellite command center's application

Jonathan Tolle

Thales Alenia Space, Cannes 06156, France
jonathan.tolle@thalesaleniaspace.com

Abstract. Thales Alenia Space has recently widely deployed user centered design process in their software product conception. This standard breaks today technology and data centric approach by integrating end-users all along the iterative design stages : context and usage understanding, end-user need specification, quick mockups and end-user validation. This paper is a return on experience. It describes and investigates a dirty UCD methodology relevance based on prototyping and user testing only (skipping user research first activities). This process is made to fit project which needs front-end requirements at day one.

Keywords: User testing, Incremental software design, User centered design, Return on experience.

1 Introduction

User centered design is a well-known software conception standard in ergonomics and human factors fields (ISO 9241-210) [1]. To guarantee system's utility and usability, it implies four stages where end-user is integrated within many ergonomics methods such as focus group [2], participatory design [3], interviews, job task-model based analysis [4], persona specification [5] and so on. In some constrained industrial context, deploying every practices cannot be possible. Thales Alenia Space believes that every process can be tailored and scaled to fit every project. For our new satellite command center (SCC) procedure executor named PRISM, we try a dirty UCD approach based on incremental mockups and user testing only. As user requirements were taken from our previous product version, this process enables earlier front-end development than classical ergonomics methodology.

This testimonial explains first how we integrated UCD activities within agile development process. Then we briefly present our study case PRISM and its incremental mockups. Next, we inspect our process relevance from usability metrics and user feedbacks. Does this approach lead to solution convergence ? Does it bring high quality interaction and high user satisfaction ? Does it allow continuous interaction improvement ? From these problematic, the paper concludes on a preliminary analysis of this dirty UCD approach (ongoing work). Please note this article doesn't list any SCC

user needs. It only aims to analyze the process relevance from ergonomics point of view.

2 Process overview

Industry often works with subcontractors while developing products. When interaction designers are not involved in project planning phase, it may result to a development-first approach without much consideration for ergonomics activities [10]. It can be hard to break this convenient design culture as subcontractor is employing software engineers in tight timeframe. In this context, we try a new incremental process to deal with early front-end development requirements.

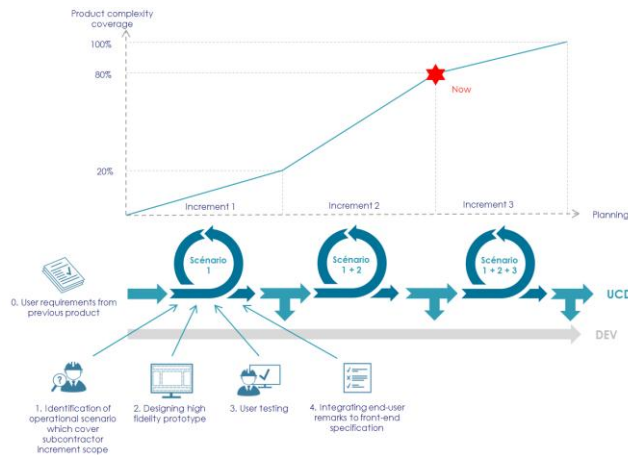


Fig. 1. ‘Early incremental user testing design approach’ activities

To ensure usable and useful software, we empirically plan 3 user test increments with a minimum of 5 participants per end-user test [6]. These tests are based on the walkthrough ergonomics methodology [7] by using fictive operational scenarios. Scenario’s increments are built from simple and non-risky interactions (increment 1) to complex widget and data visualization interactions (increment 2 and 3). Scenario coverage (e.g. “1+2+3”) between increments allows us to compare if previous front-end modifications bring better ergonomics satisfaction through usability metrics. At the end of each increment, user observations are discussed with the project manager before integrating them to subcontractor’s agile development: considering usability gain over development planning, feasibility and effort.

3 Study case : PRISM

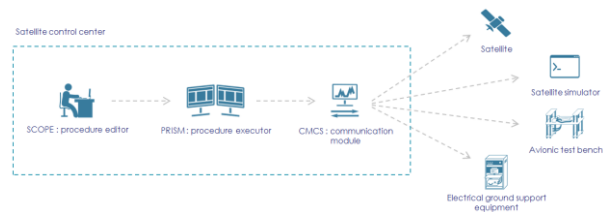


Fig. 2. PRISM simplified eco-system

PRISM is a web-based application inside the SCC eco-system. It basically executes procedure (list of command and automatic checks edited from SCOPE) to the real satellite or to a test platform such as avionic test bench, simulator, and other electrical ground system through the communication module (CMCS).

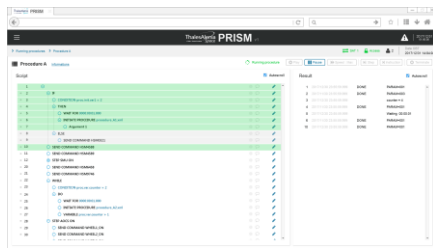


Fig. 3. Increment 1: main screen which displays direct feedback of the procedure execution

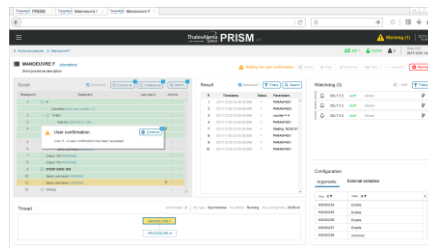


Fig. 4. Increment 2: main screen which displays direct feedback of the procedure execution and provides complementary services to end-user

Thales Alenia Space has specified a graphical charter for every web applications and lot of reusable widgets have already been designed for other solutions (such as button, table, list, interactive menu, and so on). Therefore we decided to directly work with high fidelity mockups rather than having a first low fidelity prototype's iteration. Prototypes were made using AXURE software. This tool creates interactive interfaces that can simulate expected scenario's solution behavior and be used in our user testing sessions.

4 Process inspection

4.1 User testing protocol

To investigate process relevance, it is important to set up a testing protocol. It allows us to compare data between increments. At the moment, two end-user test iterations have been performed (the third and last iteration will be done before product release). 10 distinct end-users participate to this analysis : 5 at first iteration and 7 at the second

one (2 persons from increment 1 were involved as well in increment 2). Participants are all PRISM final end-users and come from different departments with different level of expertise. Each individual test session lasts 1 hour and a half and follows the same walkthrough methodology [7] :

1. Welcome participant, Informed consent form signature and present user centered approach
2. Scenario reading and mockup limitation explanations (example : few auto scroll interactions could not be simulated)
3. Participant interaction with the prototype performing scenario goals without any help from the ergonomist. If the user seems stuck after many tries, the ergonomist gives the solution but then the interaction is tagged as failed (0% or 100% in Fig.7 and Fig.8).
4. System usability scale (SUS) survey to measure global software satisfaction
5. Task satisfaction scale form and user remark debriefing
6. Open questions and participant acknowledgments

4.2 User satisfaction

System usability scale.

At the end of each test, participants fill SUS form. SUS gives us the global satisfaction of the prototype. Rather than “Think aloud protocol”, participants were not allowed to give any observation before this step. This protocol aims to avoid any ergonomist’s explanation bias from participant satisfaction opinion.

#	Question	SUS score					Mean of SUS scores
		U1	U2	U3	U4	U5	
1	Je pense que je vais utiliser ce produit fréquemment	2	4	4	4	4	25 Worst imaginable 38 Poor 52 Ok 66 Average 72 Good 85 Excellent 92 Best imaginable
2	Je trouve ce produit inutilement complexe	4	3	4	2	4	
3	Je pense que ce produit est facile à utiliser	3	4	2	2	4	
4	Je pense qu'il y aurait besoin de l'aide d'un technicien pour être capable d'utiliser ce produit	4	4	3	3	4	
5	J'ai trouvé que les différentes fonctions de ce produit ont été bien intégrées	1	4	4	2	4	
6	Je pense qu'il y a trop d'incohérence dans ce produit	4	4	4	4	4	
7	J'imagine que la plupart des gens serait capable d'apprendre à utiliser ce produit très rapidement	3	4	4	3	4	
8	J'ai trouvé ce produit très lourd à utiliser	4	3	4	3	4	
9	Je me sentais très en confiance en utilisant ce produit	2	3	4	2	3	
10	J'ai besoin d'apprendre beaucoup de choses avant de pouvoir utiliser ce produit	3	3	4	3	4	
		Total :	72	90	82,5	79	97,5
		Sum :	85				

Fig. 5. French translated SUS result from the 5 participants of increment 1.

#	Question	SUS score						Mean of SUS scores
		U1	U2	U3	U4	U5	U7	
1	Je pense que je vais utiliser ce produit fréquemment	3	4	4	3	0	4	25 Worst imaginable 38 Poor 52 Ok 66 Average 72 Good 85 Excellent 92 Best imaginable
2	Je trouve ce produit inutilement complexe	4	3	3	3	4	3	
3	Je pense que ce produit est facile à utiliser	1	3	3	1	3	3	
4	Je pense qu'il y aurait besoin de l'aide d'un technicien pour être capable d'utiliser ce produit	1	3	1	3	1	2	
5	J'ai trouvé que les différentes fonctions de ce produit ont été bien intégrées	1	2	3	2	2	1	
6	Je pense qu'il y a trop d'incohérence dans ce produit	2	4	4	3	1	2	
7	J'imagine que la plupart des gens serait capable d'apprendre à utiliser ce produit très rapidement	3	4	3	3	3	4	
8	J'ai trouvé ce produit très lourd à utiliser	3	4	4	3	2	3	
9	Je me sentais très en confiance en utilisant ce produit	2	2	3	3	1	4	
10	J'ai besoin d'apprendre beaucoup de choses avant de pouvoir utiliser ce produit	1	3	3	3	2	4	
		Total :	47,5	86	72,5	60	75	72,5
		Sum :	69,6					

Fig. 6. French translated SUS result from the 7 participants of increment 2. Note that one participant results have been removed from average score due to inconsistent answers.

From increment 1 and 2, we can observe a big satisfaction deterioration : losing 17 points on SUS average score. The main empirical hypothesis raised from this drop is the higher complexity and the higher number of widgets provided between increments

(see Fig.3 and Fig.4). For the next iteration, our SUS goal is to reach a score higher to 72/100 by moving secondary widgets out of first end-user's eye exploration (and placing them into visible / hidden panel).

Task satisfaction scale.

SUS doesn't extract which technical task lower the satisfaction. So we decided to add a dirty homemade survey (rating from 1 to 5). This survey aims to prioritize which part of the software front-end need to be improved for the next iteration. A proper survey wasn't investigated as it would have taken too much time in the test protocol.

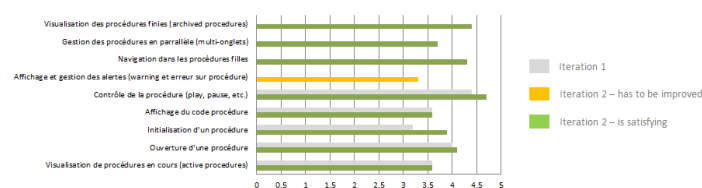


Fig. 7. Bar chart high-level functionality satisfaction result from both iterations (1 & 2). From an empiric decision, every score below 3,4/5 is marked “has to be improved”

From increment 1 and 2, five equivalent tasks were covered (e.g. “ouverture d’une procédure”) and four tasks were totally new. We can notice that modifications made between increments did bring better satisfaction for three of them (while the others stayed stable). By contrast with SUS score, it gives us a good feedback for our dirty UCD approach relevance. This chart will also be used to compare this analysis to iteration 3 and check task satisfaction progress.

4.3 Interaction quality

Interaction quality is inspected through two usability metrics :

- Effectiveness : does your participant succeeded or not to complete the interaction ? (column “goal reached”)
- Efficiency : does your participant make errors before completing his goal ? And how long does he take to finish it ? (column “error rate” and “time”)

Scenario part	Interaction	Objectives	Result		
			Goal reached	Error rate	Time (minutes)
Ouverture et initialisation procédure A	1.01	Clic du “open procedure”	100%	0,2	6,7
	1.02	Ouverture de la procédure A	100%	1,4	
	1.03	Configuration variables avant instanciation procédure	100%	0	
	1.04	Validation configuration et instanciation	100%	0	
	1.05	Expand du step pour trouver l'instruction	80%	1,4	
Manipulation et lancement de la procédure A	1.06	Mise en place du point d'arrêt sur l'instruction	80%	3,8	
	1.07	Création de l'annotation	100%	2	
	1.08	Validation de l'annotation	100%	0	
	1.09	Lancement de la procédure en automatique	100%	0	
	1.10	Déroulement de la procédure “instruction par instruction”	100%	1,2	
	1.11	Relance de la procédure en automatique	100%	0	
	1.12	Visualisation de la procédure terminée	100%	0	

Fig. 8. Efficiency and effectiveness report from iteration 1 (5 participants)

Scénario part	Ref	Interaction	Result		Time (minute)
			Goal reached	Error rate	
Ouverture et initialisation manœuvre F	1.01	Clic "New procedure"	100%	1,7	11,7
	1.02	Ouverture "SAT 1"	100%	0,1	
	1.03	Ouverture "OOC Contingency"	100%	0,3	
	1.04	Sélection "Manœuvre F" dans liste	100%	0,3	
	1.05	Clic sur "Select"	100%	0,7	
	1.06	Choix du mode "Interactive"	100%	0	
	1.07	Clic "Open"	100%	0	
	1.08	Initialisation "Arguments" clic sur "Submit"	100%	0	
	1.09	Initialisation "Variable externes" clic sur "Submit"	100%	0,4	
Switch et ouverture manœuvre I dans nouvel onglet	2.01	Clic sur "Active procedures"	71%	3,8	11,7
	2.02	Clic "New procedure"	100%	0,6	
	2.03	Clic sur "Nouvel onglet"	86%	0,7	
	2.04	Ouverture "SAT 1"	100%	0	
	2.05	Ouverture "OOC Contingency"	100%	0	
	2.06	Sélection "Manœuvre I" dans liste	100%	0	
	2.07	Clic sur "Select"	100%	0	
	2.08	Choix du mode "Interactive"	100%	0	
	2.09	Clic "Open"	100%	0	
Ouverture de manœuvre F dans un nouvel onglet et lancement des deux manœuvres en parallèle	2.10	Initialisation "Arguments" clic sur "Submit"	100%	0	11,7
	2.11	Initialisation "Variable externes" clic sur "Submit"	100%	0	
	3.01	Switch onglet "PRISM"	86%	0,6	
	3.02	Clic sur droit (simulé) sur "Control" Manœuvre F	100%	2,3	
	3.03	Clic sur "Nouvel onglet"	100%	0	
	3.04	Clic sur "Play" Manœuvre F	100%	0	
	3.05	Switch onglet "Manœuvre I"	100%	0	
	3.06	Clic sur "Play" Manœuvre I	100%	0	
	Gestion des warnings et des alertes. Affichage des procédures archivées	4.01	Clic sur "Warning" dans la header barre	86%	
4.02		Clic sur "Go to"	100%	0	
4.03		Clic sur "Continue" dans la popup de warning	100%	0	
4.04		Switch sur la procédure Manœuvre F2 par le composant Thread	100%	1,7	
4.05		Clic sur "Continue" dans la popup d'erreur	100%	0	
4.06		Switch back sur la Manœuvre F dans le composant Thread	71%	1,4	
4.07		Switch onglet "Manœuvre I"	100%	0,1	
4.08		Clic sur "Link" de la popup d'archivage	100%	0	

Fig. 9. Efficiency and effectiveness report from iteration 2 (7 participants)

The tables above list every interaction needed to complete the scenario at increment 1 and 2. The table colorization shows which operation need to be improved or modify in priority for the next increment. For instance, between mockup 1 and 2, a big front-end modification has been designed to open and initialize a procedure : passing from native browser “file explorer” to “custom procedure explorer” (interactions represented by #1.02 to #1.04 of Fig.7 and #1.02 to #1.09 of Fig.8). Surprisingly even if task satisfaction grew significantly (Fig.9), the newer interface didn’t lower interaction error rate: 1,6 at iteration 1 and 3,5 at iteration.

On other hand, time measures didn’t bring any comparative added-value on interaction quality. Overall time evaluation must be changed to task-oriented time evaluation. For the next iteration, time should be sampled by task (column “scenario part”) and task should be kept consistent on next iterations. Also by using keystroke-level ergonomics method [9], we could analyze and compare the interaction performance to what an expert would have done.

4.4 Solution convergence

Solution convergence is analyzed from the user remark number at each end of increment (see activity 4 Fig.1).

#	Widget	Besoins / Remarques	Rapport	Besoin émis par	Quand	Niveau de satisfaction perçu	Complexité perçue	Status	Commentaire
98	GLOBAL	Garder une interface claire, fluide et non surchargée. Pas de fenêtre dans tout les sens. Vu-à-vis des autres SCC concurrents l'interface est très moderne.	4	AIT + VCF	Test utilisateur - itération2	High	Medium	Pris en compte	
99	procedure-thread	Manque de visibilité sur le déroulement des procédures mères en parallèle sur le satellite ainsi que leur état (mieux les mettre en valeur)	4	OPS + AIT + VCF + CDSL	Test utilisateur - itération2	High	Low	Pris en compte	
100	procedure-alerts	Les erreurs ne sont pas assez visibles. Faire cliquer le label d'erreur pour attirer le regard de l'opérateur (autour qu'au centre de commande satellite, un opérateur est face à une multitude d'écran)	3	AIT + VCF + CDSL	Test utilisateur - itération2	High	Low	Spécification	Avec robert car panie DEV TAS
101	procedure-thread	Ajouter une arborescence / visualisation pour voir toutes les procédures en cours sur le satellite à chaque procédure ouverte (très intéressant pour les clients aussi)	5	OPS + AIT + VCF + CDSL	Test utilisateur - itération2	High	Medium	Pris en compte	Iteration 3
107	moniteur-ensemble	Quitté et basculer automatiquement sur les procédures qui se déroulent sur le satellite. Si il y a des modifications en parallèle	2	OPS + AIT	Test utilisateur -	High	Medium	A discuter	Impact sur comment on fonctionne le...

Fig. 10. User remark list sample. Each remark is tagged with the number of people who ask for it, from which department the participant comes from, when it has been identify (increment number), what is the empirical perceived ergonomics gain if integrated and what is the perceived complexity to implement it.

At the first increment, we report 97 user remarks on the tested prototype. At the second iteration, we count only 51 observations (approximately -50%) for a prototype which cover more complexity and tasks. This reduction shows a first solution convergence to the final user requirements. This preliminary analysis will be corroborated with the next increment results.

4.5 Continuous improvement

Another important point measured in our process is the continuous improvement of PRISM interactions at each increments.

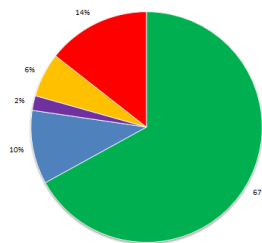


Fig. 11. Remark traceability chart iteration 1

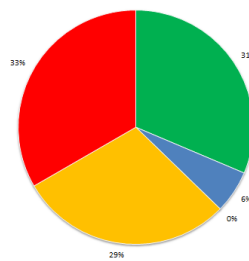


Fig. 12. Remark traceability chart iteration 2

From remark listing (Fig.10), every user feedback is labeled by his integration status (according to interaction designer and project manager discussion) : Accepted (green) meaning “sent directly to development requirements” - To be specified (blue) meaning “related to other software system specifications” - Delayed (purple) meaning “to be investigated at next iteration” - To be discussed (orange) meaning “to be discussed with the chef architect for feasibility investigation” - Rejected (red) meaning “remark is not relevant or in contradiction with previous taken decision”. From the two increment statements, we can deduce that the closer the project gets to the deadline, the fewer remarks are integrated into development (-54% accepted, +58% rejected and +80% to be discussed).

5 Conclusion

This paper presents the ongoing work of an early incremental user testing design approach. This dirty user centered design process allows front-end development at day one (as the first increment was prototyped and tested within 10 days). To assess this

methodology relevance, we base our preliminary analysis on utility, usability and traceability metrics from two incremental user testing iterations.

As more complex features are integrated at each iteration, this approach doesn't seem relevant to keep SUS score consistently high. Also from interaction efficiency and effectiveness metrics, no significant improvements were noticed. These results are mainly due to the incremental approach where designer has to focus on next increment requirements rather than improving previous ones. From user remark traceability statement, this process doesn't as well give the opportunity to make major modifications as the development team gets closer to the release deadline.

However, homemade task satisfaction progress and user remark number analysis give a good feedback on the convergence of user needs. Furthermore, by integrating user in prototype testing loops, we identify one "game changer" idea which will be developed and test at iteration 3.

If development-first approach is requested and mandatory, this incremental user testing design approach can be applied as a dirty UCD process. Despite the lack of user research activities, our return on experience shows solution convergence and great feedback from participants but low capabilities on continuous improvement (interaction usability and remark integration) and some risks of user satisfaction drops.

6 References

1. Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems, ISO 9241-210, F.: <https://www.iso.org/standard/52075.html> (2010).
2. Caplan, S. (1990). Using focus group methodology for ergonomic design. *Ergonomics*, 33(5), 527-533.
3. Muller, M. J., & Kuhn, S. (1993). Participatory design. *Communications of the ACM*, 36(6), 24-28.
4. Martinie, C., Palanque, P., Barboni, E., & Ragosta, M. (2011, October). Task-model based assessment of automation levels: application to space ground segments. In *Systems, Man, and Cybernetics (SMC), 2011 IEEE International Conference on* (pp. 3267-3273). IEEE.
5. Bornet, C., & Brangier, E. (2013). La méthode des personas: principes, intérêts et limites. *Bulletin de psychologie*, (2), 115-134.
6. Nielsen, J. (2000). Why you only need to test with 5 users.
7. Wharton, C. (1994). The cognitive walkthrough method: A practitioner's guide. *Usability inspection methods*.
8. Van Someren, M. W., Barnard, Y. F., & Sandberg, J. A. C. (1994). The think aloud method: a practical approach to modeling cognitive.
9. Card, S. K., Moran, T. P., & Newell, A. (1980). The keystroke-level model for user performance time with interactive systems. *Communications of the ACM*, 23(7), 396-410.
10. Holtzblatt, K., Beringer, J., & Baker, L. (2005, April). Rapid user centered design techniques: Challenges and solutions. In *CHI'05 Extended Abstracts on Human Factors in Computing Systems* (pp. 2037-2038). ACM.