

HEURISTICS EVALUATION TECHNIQUES IN AGILE SOFTWARE MODEL FOR THE DEVELOPMENT OF USABLE SOFTWARE

Saad Masood Butt

Faculty of Computer Systems & Softwares Engineering, University Malaysia Pahang
saadmasoodbutt668@yahoo.com

AzuraOnn

Department of Management and Human Resource, Universiti Tenaga Nasional, Malaysia
azura@uniten.edu.my

Mazlina Abdul Majid

Faculty of Computer Systems & Softwares Engineering, University Malaysia Pahang
mazlina@ump.edu.my

ABSTRACT— *Majority of agile projects currently involve interactive user interface designs, which can only be possible by following UCD in agile software model. But the integration of UCD is not clear in the current agile models. Both Agile models and UCD have iterative nature but agile models focus on coding and development of software; whereas, UCD focuses on user interface of the software. Similarly, both of them have testing features where the agile model involves automated tested code while UCD involves an expert or a user to test the user interface. In this paper, a new agile usability model is presented and tested in companies and are presented. Key results from these projects clearly shows: the proposed agile model incorporates usability evaluation methods, improve the relationship between usability experts to work with agile software experts; in addition, allows agile developers to incorporate the result from UCD into subsequent interactions.*

Keywords- agile software development; user centered design, user interface, usability evaluation

I. INTRODUCTION

Increasing significance of software in today's modern world has made the software development process changeable and inherits a multidisciplinary approach to achieve software quality. The ongoing major research using current agile methods aims to achieve usability of the software. Consistent gaining popularity of the agile method is due to the iterative and fast development in fewer spans of time. This agile feature of rapid development has given a big success in the domain of software models, but a big question arises on how to make software usable and increase its usability in less time.

With the passage of time, a new field arises in the field of computer science called Human Computer Interaction (HCI). One of the common topics in HCI is User Centred Design (UCD). UCD design website, mobile applications etc. are based on the user experience. Main goal is to design an interface with simplicity and efficiency to achieve the user's goal. Still, the question is how to incorporate the UCD in Agile model? Both agile and UCD have iterative nature. While agile focused on a working software exclusive on the coding and development of a software; whereas, UCD focused on interface designing and making interactive interface design for the users. Similarly, both agile and UCD have testing methods. Agile model follows automated code testing; while, UCD have usability experts or user for the testing of software. In this paper, some literature has shown the importance of UCD in agile model after which a new software model is introduced, explaining all its processes. Next section shows explains on how UCD can be integrated with an agile model. In the main section, some projects are presented with the results and in the end present the conclusions.

II. BACKGROUND

Conventional approaches in design and implementation of software models have their own draw back. Most of these methodologies have different challenges like: continuous change in software requirements and other schedule delays which overall slows down the end product in software development. Following agile methodologies along with the reasons on product iteration and reduction in team coordination cost result sometimes to unproductivity.

Usability in agile software has become an important perspective for software quality assurance. Awareness of actual end users in software development life cycle is mostly a critical aspect. Most of the time, word customer is used instead of end users. Customers are giving the requirements, taking part in software design and are also involved in testing of software from the first stage of software development till the end. So the real query behind is this: Are customers the actual end users or just representatives of end users in software development?

McBreen [3] elaborated the notion of software agile method as strategy of rapid development spending less time on analysis and design. Customer's feedback of the system should be flexible must undertake upon delivery of each milestone. This can be achieved by making the development process speedy and by following an incremental and iterative approach to help a customer in a resourceful way. Software interface designing is not supported due to incremental and iterative approaches of agile methodologies, which is obligatory for the development of usable and interactive software [4].

Kane [5] argued that the agile methods do not support the usability or explicitly there is disintegration of usability in the agile methods. Also, integration of usability in agile methods will boost overall product usability along with the satisfaction of the end user.

Fox et al [6] addressed that no agile methods can guarantee the usability of software or product. Usability techniques in development methodology are usable in the development of interactive and usable software design especially in case of agile software methodologies.

Still, many software organizations used agile methods in software development in a small frame of time. But in the end of the day, they have the same opinion that agile alone cannot guarantee the usability of the software. So Usability integration within the process is mandatory [7]. But usability specialists are not sure that the resultant software is developed with the authentic end-users contribution or not [8]. Extreme Programming (XP) that is frequently used in agile method is still a customers' worth onsite during the development process but uncertainty remains whether these are the actual users or representative of end users [9].

Ferreira et al [15] argued about introducing Big Design Up Front (BDUF) in agile development methodologies. But on real grounds, an agile method is different in concept of BDUF since more of the design is upfront. Therefore, it is complex to change later on. BDUF is a technique in agile software methodology in which data about design detail is generated before programming modules and testing. Companies using agile methods do not focus on this method and spend substantial amount of time on code testing in order to build up workable software.

In software development, every single module must be validated through testing. Similarly, in agile methods acceptance testing and unit testing are significant testing techniques. These techniques test every software module in diverse environment but none of these methodologies directly support usability testing and usability evaluation of software. In agile, skilled analysis and developing paper prototypes do not assure product usability. Agile specialist spends no time on extensive usability testing with the end users in order to get product usability [8].

Singh [16] proposed a new methodology called U-SCRUM which supports two kinds of product owners: product functionality and product usability. Unlike a conventional SCRUM models, which focuses in development of software that is workable in a small amount of time? Hence, following U-SCRUM tactics can ensure usability in all SCRUM projects.

Gigantic companies like IBM put emphasis on the concept of usability in software development. This quality attributes not only incassates the market value of that product but also makes it reusable [17]. Similarly, Microsoft recently published a report focusing on the benefits of usability in software development. The reason to achieve software usability is to help minimize the training cost on usable software and enhance the user acceptance of the software by the end users.

III. PROPOSED AGILE USABILITY MODEL

Agile Software Development method was developed in order to hasten the development process. The main focus relies in these: coding and development of functional requirements and in running software. Though the software is running and has all the functional requirements, failure is caused due to the interaction by the users with lack of usability. Revising all the software features, which are non-interactive cost a lot and reduce the quality of software.

In this paper, a new software development model is presented named as Agile Usability Software Model. This model contributes few things in the domain of agile models like UCD and Usability which is considered as a proper process in the whole model, UCD issues and Usability results will be fixed before the development of software starts. Now, only typical testing like (Acceptance testing or Unit testing) will be done as well as usability testing is considered as a part of other testing. In this way, agile software developers and usability experts can work together under one model and can develop interactive software. Proposed model have five processes: FlexREQ, Interactive User Interface Designing, Cognitive analysis of software interfaces, Development & Testing and Deployment.

FlexREQ Process

The first process in the proposed model is FlexREQ [18] as shown in Figure 1. In the traditional software model, eighty percent effort is done on the development of SRS (Software Requirement Speciation) document. Customers help makes this document, where the development starts. Sometimes, the customer is non-technical and do not understand the SRS document properly which will highly affect the final software since it is based on SRS which is made by the help of a customer understanding the SRS or not.

In the agile model, the role of SRS is eliminated and has adopted an iterative and incremental approach. The iterative approach helps in adding the missing functions in the next release which was missing in the first release; while incremental approach helps to release the software in a form of small modules later merging in one big modules for final deployment at the end.

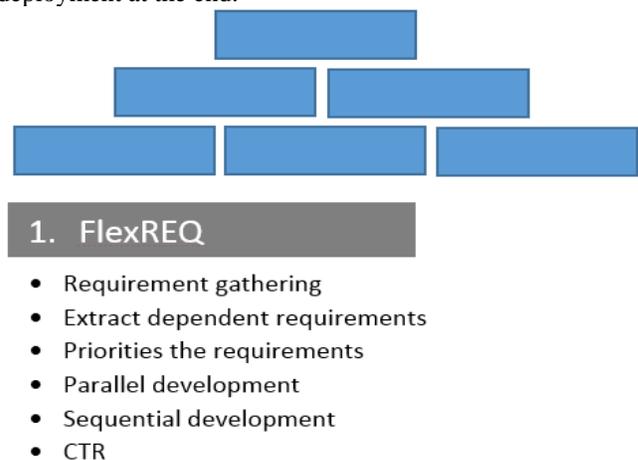


Figure 1: The FlexREQ [20]

FlexREQ is the automated requirement elicitation process that helps project team leader to manage the customer requirements and select the few requirements based on customer interest for the first release. Requirements selected for first release can undergo either parallel development or sequential development. For every release it calculates the CTR (Cost, Time and Resources) and records the process. This FlexREQ process helps to collect user requirements and manage them efficiently in a small amount of time.

Interactive User Interface Designing (IUID) Process

Software interactions mean high level of user interaction with the software. To develop interactive software, (like video games or mobile applications or operating system like Windows 8) a high level of Usability is considered and the users of usability experts will test it.

Using known agile models, result in a beneficial way of having faster software development in a minimal time. It has become a challenge for agile experts to focus on UCD and usability for the development of Interactive software. The second process in proposed model is the IUID (Interactive User Interface Designing) shown in Figure 2 that is designing of interactive interface mock-ups. IUID makes an objective to design the interface that is simple and interactive so that users can achieve its goal.

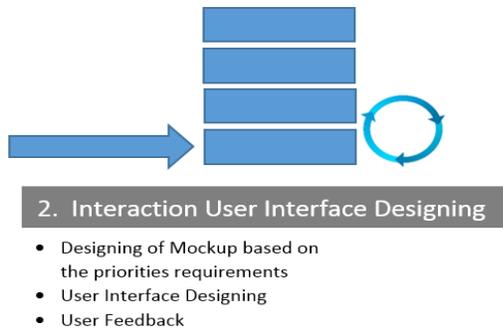


Figure 2: IUID Process

Cognitive analysis of software interfaces (CASI) Process

CASI as shown in Figure 3, is a usability evaluation process that evaluate every mock-up for the usability before the coding starts. The main objective of this process is to find Usability bug in the mock up and fix those bugs.

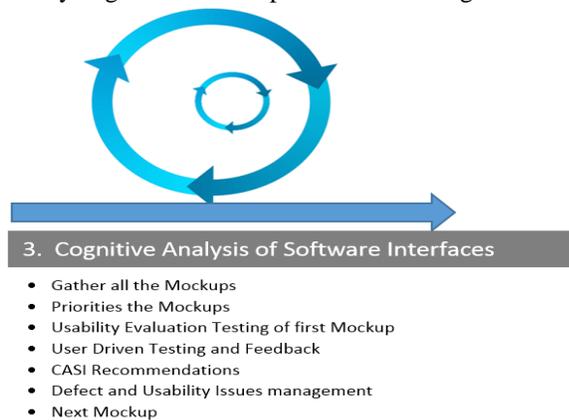


Figure 3: CASI Process

Mock-up having poor usability and bugs are send back to the previous process to fix those bugs. Successful mock-ups from CASI will now move for the coding phases.

Development & Testing Process

Shown in figure 5, successful mock-ups from CASI process will go for coding and testing. All mock-ups will be tested one by one after coding. This test involves Unit and Acceptance testing. The difference between these testing and CASI is that unit and acceptance testing will test the functional requirement of software whereas CASI will test the usability of the software.

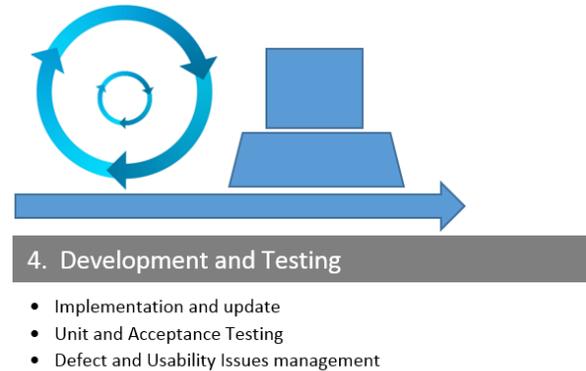


Figure 4: Development and Testing Process

Deployment Process

After all iterations, the software is now ready for deployment for the customer in the last step of overall process shown in Figure 5. The proposed model has this much capacity to accept new requirements or modify the existing requirements based on the customer request.

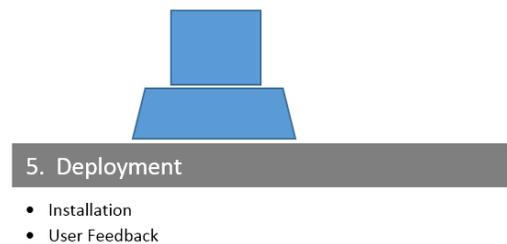


Figure 5: Deployment Process

IV. CONCLUSION

This paper shows some research on UI designers and Usability in agile software development. For this research, few projects have been conducted on the proposed agile model to show how UI, Usability Evaluation fit into the structure of agile iterations. These projects discussed took considerable time to design the interactive interface and maintain a good connection between Users, UI designers, Usability Experts and developers. The project aroused few results that emerged in combining usability to agile process like: Iteration improve UI design, Iteration support UE, CASI result changes in development and proposed model in

order to accommodate Software Developers and UI Designers under one model.

This point reflects a coherent picture of putting UI and Usability Evaluation in Agile processes for extensive advantages. Its iterative nature allows UI designer and Usability evaluation to identify bugs and fix the bugs in the next iteration, which are not expected but happens and advantages are seen. Moreover, the proposed model brought UI designers, UE experts, users and software developers under one model and hope that the practitioners work more closely to achieve their common goal.

REFERENCES

- [1] Koskela, "Software configuration management in agile methods," VTT publication 5 14, ESPOO, 2003, pp 1-54,
- [2] Lee, J. C., & McCrickard, D. S. (2007, August). Towards extreme (ly) usable software: Exploring tensions between usability and agile software development. In Agile Conference (AGILE), 2007 (pp. 59-71). IEEE.
- [3] D, Kane, "Finding a Place for discount usability engineering in agile development: Throwing down the gauntlet," Proc, Agile Development Conference (ADC'03), IEEE Press, 2003, ppAO-46.
- [4] D, Fox, I. Sillito and F. Maurer, "Agile methods and user-centered design: How These Two methodologies are being successfully integrated in industry," Proc, AGILE 2008 conference (Agile '08), IEEE Press, 2008, pp, 63-72.
- [5] D Sy, "Adapting usability investigations for agile user-centered design," Journal of Usability Studies, vol. 2, Issue 3. 2007, pp, 1 12-132.
- [6] Sohaib, O., & Khan, K. (2010, June). Integrating usability engineering and agile software development: A literature review. In Computer Design and Applications (ICCD), 2010 International Conference on (Vol. 2, pp. V2-32). IEEE.
- [7] M, Najafi and L. Toyoshiba, "Two case studies of user experience design and agile development," Proc, AGILE 2008 conference (Agile '08), IEEE Press, 2008, pp, 2 167-2177.
- [8] H. Sharp, H. Robinson, and I. Segal, "Integrating user centered design and software engineering: a role for extreme programming," [Online] Available: http://www.ics.heacademy.ac.uk/events/presentations1376_hcie_sharp2.pdf [Accessed: December 2013].
- [9] T. Memmel, F. Gundelsweiler and H. Reiterer, "Agile human-centered software engineering," Proc, HCI 2007, British computer society Press, 2007, pp, 167-175.
- [10] J. Kollmann, H. Sharp and A. Blandford, "The importance of identity and vision to user experience designers on agile projects," Proc, AGILE 2009 conference (Agile '2009), IEEE Press, 2009, pp, 1 1-18.
- [11] J. Patton, "Hitting the Target: Adding interaction design to agile
- [12] M Singh, "U-SCRUM: An agile methodology for promoting usability. " Proc, AGILE 2009 conference (Agile '09), IEEE Press, 2009, pp, 555-560.
- [13] M McNeill, "User centered design in agile application development," [online] Available: http://www.thoughtworks.com/pdfs/agile_and_UCD_MM.pdf [Accessed: December 2013].
- [14] P. Mcinerney and F. Maurer, "UCD in agile projects: dream team or odd couple?" Interactions, vol. 12, issue 6, ACM Press, 2007, pp, 19-23.
- [15] Parsons, D., Lal, R., Ryu, H., & Lange, M. (2007). Software development methodologies, agile development and usability engineering. ACIS 2007 Proceedings, 21.
- [16] Shackel, B. (2007). IBM makes usability as important as functionality. The Computer Journal, 29(5), 475-476.
- [17] Microsoft Corporation: "Usability in Software Design", [Online] Available <http://msdn.microsoft.com/en-us/library/ms997577.aspx>. [Accessed: December 2013].
- [18] Masood Butt, S., & Ahmad, W. F. W. (2012, June). Handling requirements using FlexREQ model. In Software Engineering and Service Science (ICSESS), 2012 IEEE 3rd International Conference on (pp. 661-664). IEEE.
- [19] Ferreira, J., Noble, J., & Biddle, R. (2007, August). Agile development iterations and UI design. In Agile Conference (AGILE), 2007 (pp. 50-58). IEEE.
- [20] Kavitha, C. R., & Thomas, S. M. (2011). Requirement gathering for small projects using agile methods. International Journal of Computer Applications (IJCA) Special Issue on "Computational Science-New Dimensions & Perspectives (3) Published by Foundation of Computer Science, 122-128.