

# Usability Evaluation Techniques for Agile Software Model

Saad Masood Butt<sup>1\*</sup>, Shahid Masood Butt<sup>2</sup>, Azura Onn<sup>3</sup>, Nadra Tabassam<sup>4</sup>, Mazlina Abdul Majid<sup>5</sup>

<sup>1</sup> Faculty of Computer Systems & Softwares Engineering, University Malaysia Pahang, Malaysia.

<sup>2</sup> Department of Management Sciences, Hamdard University, Islamabad Campus Islamabad, Pakistan.

<sup>3</sup> Department of Management and Human Resource, Universiti Tenaga Nasional, Malaysia.

<sup>4</sup> Comsats Institute of Information Technology, Wah Cantt, Pakistan.

<sup>5</sup> Faculty of Computer Systems & Softwares Engineering, University Malaysia Pahang, Malaysia.

\* Corresponding author. email: saadmasoodbutt668@yahoo.com

Manuscript submitted January 10, 2014; accepted August 12, 2014.

---

**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.

**Key words:** Agile software model, usability evaluation, user interface, user centered designed.

---

## 1. 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.

## **2. 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.

Conventional software design methodologies have its darker side and that is taking more time in developing and meeting the requirement in a given deadline. These drawbacks generate the need of new strategy like software agile development. Agile development overcomes the time barrier and software completion within a given framework along with frequent customer involvement and capability to handle requirement changes if any [1]. Agile methodologies focus on incremental and iterative approaches that not only supports overall optimization in development process but also support developers to generate product incremental results in small amount of time [2].

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].

Sharp *et al.* [10] intended to focus on customers and Extreme Programming (XP) collaboration and to emphasize on the effectiveness of UCD in XP. The foremost issue focused in the paper was the shortage of reliance between customers and developers. Involvement of the right user not only helps in getting the right

requirement but also enlarges the usability of overall software. Still, there is a difference between customers and real users in the development of software [11].

Memmel *et al.* [11] gave concepts of a new model called Cross-discipline User Interface design and Software Engineering (CRUISER) lifecycle along with HCI and SE. Involvement of the user in the designing stage produces usable results, which can overall optimize the development process. Agile methodology targets in delivering a piece of the working code in a small frame of time. The iterative approach can fabricate good quality code, but it will not assure consistent user interface as the focus is only in delivering small pieces of functionality after every interaction [12]. Agile methodologies produce software in workable form but cannot make sure the delivery of usable software. The XP delivers good quality software but it might fail to satisfy the customer. Hence, to generate usable interactive software, UCD is a must in agile methodologies [13].

Patton [13] focused on the ten facts that can help agile specialist to integrate UCD during the software development but unclear on how the design can reach the implementation stage. Lee *et al.* [14] proposed a new model called extreme Scenario-based Design (XSBD). This model emphasized on the concept of central design record (CDR) which helps to maintain the communication between: agile experts, stakeholders and usability specialists; also, by helping usability expert in designing issues and conducting usability testing.

Ferreia *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. McNeill [17] described that techniques of storyboarding is an iterative approach and a low reliability prototyping in agile that can perk up the software design instead of spending time on coding modules. Hence, enhanced versions of these products can develop a complete focus on end-users by adding UCD in agile approaches.

High budgets and long-time periods both are not necessary for the Usability testing's effectively. Techniques such as discount usability allow engineers to work in team by thinking distinctly. Other methodologies: scenario-based, walkthroughs, card sorting and heuristics make the process much cheaper, vibrant and easy. These techniques can be applied at an earlier phase in the life cycle and the implementation phase so that any major or minor usability issues can be notified. The working framework mentioned [7], discount usability model is implemented within agile iterative settings and becomes more effective. Agile development model implements an iterative approach and has a very stern time frame wherein daily scrum based meeting is held to update the team with the recent changes happening in the project scope. Researchers came up with possibility of integration of usability methods and agile model. Adopting agile approaches while focusing on usability-centred design leads to an amazing experience that will result in a timely delivery of a highly usable end product.

Many software companies follow usability in agile methods but UCD practices are still behind. As focused in [18] the association of usability approaches in agile methodology is a good combination. However, agile and usability specialists possess diverse expertise and knowledge. Hence, the significant issue faced is that agile development merely focused on workable software along with explicit ignorance of usability issues. Along with this, many agile methods do not engage end users, which is a serious drawback in the development phase. Usability approaches require real users, which allow software developers to put into practice usable software [19]. After the research, industry reports and work done in usability, it is clear that the assumption of having the best way to evaluate the software design is to understand the end user and their interaction with the system. Beforehand, software developers who were following usability in agile modelling were not producing accurate designs. After consideration, that the world is moving from computers to Web and to advancement of Mobile applications, it is a serious requirement of UCD to increase day by day in order to attain the quality in software i.e. usability. Hence, for the development of usable software, the agile developer needs to include interactive design and good usability methods in their development methodology [20].

Gigantic companies like IBM put emphasis on the concept of usability in software development. This quality attributes not only increases the market value of that product but also makes it reusable [21]. Similarly, Microsoft recently published a report [22] 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.

### **3. 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.

#### **3.1. FlexREQ Process**

The first process in the proposed model is FlexREQ [23] as shown in Fig. 1. In the traditional software model, eighty percent effort is done on the development of SRS (Software Requirement Specification) 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.

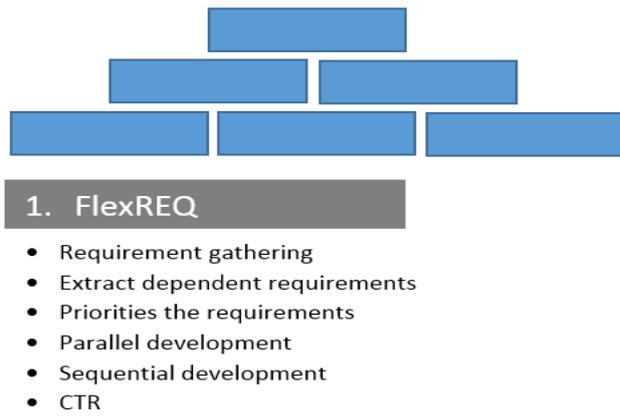


Fig. 1. The FlexREQ [23].

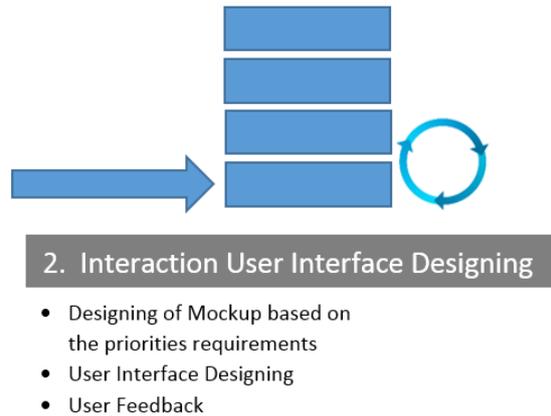


Fig. 2. IUID process.

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.

### 3.2. 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 Fig. 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.

### 3.3. Cognitive Analysis of Software Interfaces (CASI) Process

CASI [24] as shown in Fig. 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.

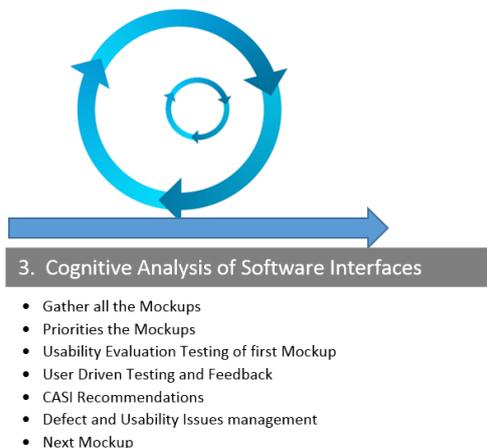


Fig. 3. CASI process [24].

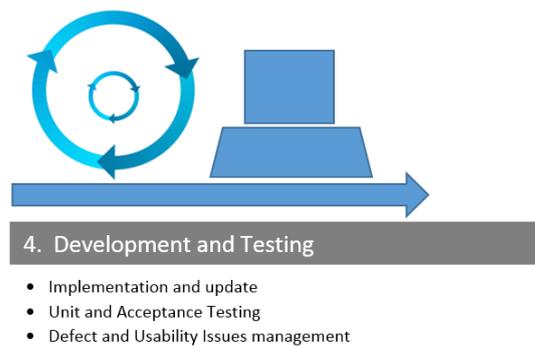


Fig. 4. Development and testing process.

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.

### 3.4. Development and Testing Process

Shown in Fig. 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.

### 3.5. Development Process

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

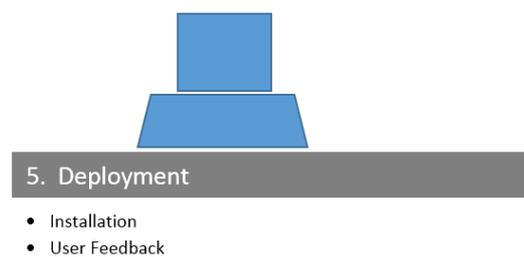


Fig. 5. Deployment process.

## 4. Project Profiles

The proposed model practically tested in various companies proves the divergence on different software projects. These are based on real projects and each project is unique according to its own way. For example, some projects need to develop new software while others were to release the new version of the software. Other projects were to improve the Usability of the existing software. This section highlights the actual practice of proposed agile usability software model.

### 4.1. Project 1

The first project, P1, develops and markets civil engineering software for new students enrolled in civil engineering discipline. P1 project was developed on the proposed model with a team of total five engineers, one domain expert and one UI designers, one usability expert, one coder and a tester. P1 involves a development of new software that helps civil engineering students to construct and present Town Plan in 3D view. The P1 begins with the collection of requirements from the domain experts like faculty member and students. During the first meeting, the requirements are prioritized based on user stories called as wish-list. These wish-list of requirements are recorded in the same order using FlexREQ. Every independent wish-list has its own time schedule. During pre-release planning meeting, it was recorded when the product will be released. Development of software consists of two or three iterations and each iteration lasts for five days.

Wish-list selected from FlexREQ are now ready for IUID. This involved domain experts to work with UI designers to design mock-ups using Balsamiq. This helps in designing simple and interactive mock-up for every wish-list. Missing features in mock-up will be added in the next iteration. All these mock-ups are now ready for Usability Evaluation. The expert domain mostly does this and results are then recorded. Poor mock-ups having usability bugs are send back to IUID to fix these bugs. Within the first iteration the successful mock-ups are coded and tested by the developer. Unit testing and acceptance testing are

performed together including the Usability expert, which is also tested from the perspective of Usability and UI. The final module that contains single wish-list will be deployed to the customer. Customer feedback will be noted or usability issues will be highlighted to fix in the next iteration. On the end of all iterations, all modules are integrated and the complete software will be deployed to the customer and it will run in the real environment. Feedback regarding changes or omission or usability issues were provide to the developers and usability expert. After these changes the software will be tested by UI expert and client.

## **4.2. Project 2**

The second project, P2, is developed software. Now, the client wants to release the next version of software because the previous have poor UI and have poor usability. The new version of software is developed using the proposed model. P2 involved the team consist of UI designer, Usability expert, Coder and onsite customer.

P2 involved the release of the next version of existing software. UI designer have a meeting with customer where the new requirements and changes in an existing requirement are noted. All this will be added in FlexREQ to get the simple wish-list of requirements. So UI designers did not perform any change in overall design of the software but they just need to redesign the existing UI design with new features or changes in the existing features. New mock-ups are developed and now ready for usability evaluation. The onsite customer does this before these mock-ups are sent to the development team. Usability issues will be identified by CASI and the report will be sent back to UI designer to fix the design issues. After fixing the design issue the development is now ready to code the requirements in the wish-list. These changes are small but the development team take minimum time to develop. Finally, they will be sent for acceptance testing. During the testing the customer walked through the new UI, usability is evaluated and how the new software is working.

## **4.3. Project 3**

The third project P3 is already developed but the users of the software had difficulty in understanding the various functions and had undertaken task difficulty. The project is then evaluated for its Usability using CASI. It was discovered that the software was developed by hiring developers from freelancer website like (ODESK). P3 consist of UI designer, Usability Expert, Developer and User. In P3, the interface of software is selected and loaded in CASI for the evaluation of Usability; where those having poor usability are send to UI designers. UI designer discussed with users in order to improve the design according to user needs. Final mock-up will be re-evaluated for the usability and later send to the developer for final development. Some interfaces, which the user wants to revise, took considerable amount of time because other functions are dependent on that interface. At that point UI designer had meetings with developers to redesign the interface without affecting other functions.

## **5. Results**

In this section, some results are presented based on the project. The results also highlight the importance of usability in agile model where both usability experts and agile experts can work together under one model.

### **5.1. Iteration Improve UI Process**

The first result observed from the projects is that iteration helps user interface designer to improve the design. This iterative nature is very cost effective that helps to cover the interface issues before the development of the software.

### **5.2. Iteration Support Usability Evaluation**

Usability evaluation means how easy for the user to work with the real software to achieve its goal or to what extent the software is easy to use. Usability evaluation is a well-established discipline of HCI. The iterative nature of agile software model seems very helpful to evaluate the usability of software at a very early stage. This helps in fixation of usability bugs during the design stage and gets proper feedback from the user before the coding starts. Moreover, usability evaluation seems fit with the agile model as it seems with the acceptance testing.

### 5.3. CASI Changes the Development Result

Testing in agile model helps to identify the bugs, which is then improved in the next iteration. This is the reason why iteration process is universally accepted for software development [25]. Similarly, the CASI process helps to identify usability bugs in the user interface which is fixed in the next iteration of software release. Having iterations in agile model is seen as an opportunity for usability evaluation to identify usability bugs or can accommodate any new feature; so that the next iteration is seen as an opportunity to fix the usability bugs or accommodate those changes.

### 5.4. Proposed Model Accommodates Software Developers and UI Designers under One Model

Final result identified in the proposed model and project is a better relation between UI designers and software developers. It is obvious from the proposed model that development of interactive software can accommodate both UI designers and agile software developer under one model. Table 1 gives a comparison of the proposed model and agile model process.

Table 1. Proposed Model vs Agile Models

	<b>Proposed Model</b>	<b>Agile Models</b>
Requirement Gathering	FlexREQ Technique	Manual way of writing User Stories [25]
Designing	IUID and Users	Software Developers
Usability	CASI	Not Defined[6]-[8]
Development	UI Designer, Usability Expert, User and Coder	Coder
Testing	CASI, Acceptance and Unit Testing	Acceptance Test [25]
Deployment	Get User Feedback	Not Defined
Aim	Develop Software up to User Expectation and focus Usability	Working Software [12]

## 6. 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] What is agile software development. Retrieved May 14, 2009, from <http://www.agilealliance.org/show/2>.
- [2] Keramati, H., & Hosseinabadi, S. H. M. (2008). Integrating software development activities with agile methodologies. *Proceedings of 6th ACS/IEEE International Conference on Computer System and Application (AICC SA '08)* (pp. 749-754).
- [3] Koskela. (2003). *Software Configuration Management in Agile Methods*. VTT publication.
- [4] Lee, J. C., & McCrickard, D. S. (2007, August). Towards extreme (ly) usable software: Exploring tensions between usability and agile software development. *Proceedings of Agile Conference (AGILE)* (pp. 59-71). IEEE.
- [5] Kane, D. (2003). Finding a place for discount usability engineering in agile development: Throwing down the gauntlet. *Proceedings of Agile Development Conference (ADC'03)* (pp. AO-46). IEEE Press.
- [6] Fox, D., Sillito, I., & Maurer, F. (2008). Agile methods and user-centered design: How these two methodologies are being successfully integrated in industry. *Proceedings of AGILE 2008 Conference (Agile '08)* (pp. 63-72). IEEE Press.
- [7] Sy, D. (2007). Adapting usability investigations for agile user-centered design. *Journal of Usability Studies*, 2(3), 112-132.
- [8] Sohaib, O., & Khan, K. (2010, June). Integrating usability engineering and agile software development: A literature review. *Proceedings of 2010 International Conference on Computer Design and Applications (ICDDA)*, 2, V2-32. IEEE.
- [9] Najafi, M., & Toyoshiba, L. (2008). Two case studies of user experience design and agile development. *Proceedings of AGILE 2008 Conference (Agile '08)* (pp. 2167-2177). IEEE Press.
- [10] Sharp, H., Robinson, H., & Segal, I. Integrating user centered design and software engineering: A role for extreme programming. Retrieved from [http://www.ics.heacademy.ac.uk/events/presentations1376\\_hcie\\_sharp2.pdf](http://www.ics.heacademy.ac.uk/events/presentations1376_hcie_sharp2.pdf).
- [11] Memmel, T., Gundelsweiler, F., & Reiterer, H. (2007). Agile human-centered software engineering. *Proceedings of HCI 2007* (pp. 167-175). British Computer Society Press.
- [12] Kollmann, J., Sharp, H., & Blandford, A. (2009). The importance of identity and vision to user experience designers on agile projects. *Proceedings of AGILE 2009 Conference (Agile '2009)* (pp. 11- 18). IEEE Press.
- [13] Patton, J. (2002). Hitting the Target: Adding interaction design to agile software development. *Proceedings of Conference on Object Oriented Programming Systems Languages and Application (OOPSLA 2002)* (pp. 1-7). ACM Press.
- [14] Lee, J. C., Scott, D., & McCrickard, K. T. S. (2009). Examining the foundations of agile usability with eXtreme scenario based design. *Proceedings of AGILE 2009 Conference (Agile '2009)* (pp. 3-10). IEEE Press.
- [15] Ferreira, I. N., & Biddle, R. (2007). Up-front interaction design in agile development. *Proceedings of XP* (pp. 9-16). Berlin Heidelberg: Springer-Verlag.
- [16] Singh, M. (2009). U-SCRUM: An agile methodology for promoting usability. *Proceedings of AGILE 2009*

*Conference (Agile '09)* (pp. 555-560). IEEE Press.

- [17] McNeill, M. User centered design in agile application development. Retrieved from [http://www.thoughtworks.com/pdfs/agile\\_and\\_UCD MM, pdf](http://www.thoughtworks.com/pdfs/agile_and_UCD MM, pdf).
- [18] Mcinerney, P., & Maurer, F. (2007). UCD in agile projects: dream team or odd couple? *Interactions*, 12(6), 19-23. ACM Press.
- [19] Parsons, D., Lal, R., Ryu, H., & Lange, M. (2007). Software development methodologies, agile development and usability engineering. *Proceedings of ACIS 2007*, vol. 21.
- [20] Agile Development Projects and Usability. Retrieved from <http://www.nngroup.com/articles/agile-development-and-usability/>.
- [21] Shackel, B. (2007). IBM makes usability as important as functionality. *The Computer Journal*, 29(5), 475-476.
- [22] Microsoft Corporation: Usability in Software Design. Retrieved from <http://msdn.microsoft.com/en-us/library/ms997577.aspx>.
- [23] Masood, B. S., & Ahmad, W. F. W. (2012, June). Handling requirements using FlexREQ model. *Proceedings of 2012 IEEE 3rd International Conference on Software Engineering and Service Science (ICSESS)* (pp. 661-664). IEEE.
- [24] Butt, S. M., & Ahmad, W. F. W. (2012). Analysis and evaluation of behavior in software interfaces using an expert system. *International Journal*, 5.
- [25] Ferreira, J., Noble, J., & Biddle, R. (2007, August). Agile development iterations and UI design. *Proceedings of Agile Conference (AGILE)* (pp. 50-58).



**Saad Masood Butt** is a full PhD research student and formally working as a research assistant as UMP. His research interest includes software engineering, human computer interaction, network usability, and agile development. He is an active member of IAENG.

He received his BS (software engineering) degree from Bahria University Islamabad, Pakistan in 2008. he completed his MS (software engineering) degree in 2010 from Bahria University Islamabad, Islamabad Pakistan. He is a recognized engineer of Pakistan approved by Higher Education Commission and Pakistan Engineering Council (PEC). He has got more than 7 years' experience and was associated with various organizations in Pakistan. Currently, he is pursuing his PhD degree in the field of software engineering.