

Best Practices for Integrating User-Centered Design and Agile Software Development

Tiago Silva da Silva

PUCRS – Pontifícia
Universidade Católica do Rio
Grande do Sul
Av. Ipiranga, 6681, Prédio
32, Sala 507
Porto Alegre, RS, Brasil
tiago.silva@pucrs.br

Milene Silveira

PUCRS – Pontifícia
Universidade Católica do Rio
Grande do Sul
Av. Ipiranga, 6681, Prédio
32, Sala 507
Porto Alegre, RS, Brasil
milene.silveira@pucrs.br

Frank Maurer

UofC – University of Calgary
2500 University Dr. NW
Calgary, AB, Canada
frank.maurer@ucalgary.ca

ABSTRACT

Agile Methods have a distinct culture that at a first glance seems to conflict with User-Centered Design. However, the use of agile methods can result in improved usability of software systems. This tutorial aims to present and discuss the best practices regarding the integration of User-Centered Design and Agile Methods. We believe that the lessons learned from this tutorial can be really useful and applicable in both industry and academic contexts.

Keywords

User-Centered Design, Agile Methods, Integration.

INTRODUCTION

In general, according to [1], software systems are designed by software engineers who most of the times are responsible for the design of the User Interface (UI) as well as the software design to implement this UI.

Even if the designers and developers are competent users of development technologies, the UI developed by them are not attractive and appropriate to their target audience.

In general software engineers are not motivated and/or do not have the specific background necessary to analyze a system from the perspective of an interaction designer.

As in Human-Computer Interaction (HCI) area, Software Engineering (SE) focuses on products with high quality. However, the quality concepts appointed by HCI emphasize the user experience using the software, while Software Engineering emphasizes quality criteria focused on structural and functional aspects of the software, such as modularity, maintainability etc.

In practice, these two areas – Software Engineering and Human-Computer Interaction – remain away from each other, and methods and techniques from one area are not well known in the other one. This is the case of Agile

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.

10th Brazilian Symposium on Human Factors in Computer Systems & 5th Latin American Conference on Human-Computer Interaction. IHC+CLIH 2011 October 25-28, 2011, Porto de Galinhas, PE, Brazil. Copyright 2011 SBC. ISSN 2178-7697

Methods and User-Centered Design.

According to [2], due to its popularity and reports of good results, Agile Methods are being quickly adopted by the industry. However, we rarely find initiatives of inclusion of usability issues into Agile Methods [3].

Agile Methods have a distinct culture that at first glance seems to conflict with User-Centered Design (UCD) [4]. However, according to these same authors, the use of agile methods can result in improved usability. Moreover, the authors did not find any interaction designers who preferred traditional approaches than agile processes.

One of the problems of integrating these two methodologies is that traditionally they use different approaches on how resources are allocated in a project [5]. Agile methods strive to deliver small sets of software features to customers as quickly as possible in short iterations. On the other hand, UCD spend a considerable effort on research and analysis before development begins.

For example, UCD associated with non-agile teams has led to a combination of results [6]. In non-agile projects, the UCD group has written UI (User Interface) specifications that are Word documents ranging from 5 to 200 pages of description and images. It can take months to complete a UI specification, besides the need for meetings to review and provide answers to questions about it.

While the two methodologies have different approaches regarding requirements gathering and upfront design, they also have similarities. The main is that both approaches are user and customer focused. As the name suggests, UCD focuses on developing software with the user in mind. Agile involve a local representative of the client to shorten the feedback loop between the development team and the customer.

Considering the importance of HCI, the distance between the Software Engineering and HCI areas, the tendency to the use Agile Methods, and the increasing interest on the integration of this two areas, as presented in the Figure 1, this tutorial aims to present some of the best practices identified regarding the integration of these different areas, focusing on design and usability evaluation in the field of

HCI, and Agile Methods in the field of Software Engineering.

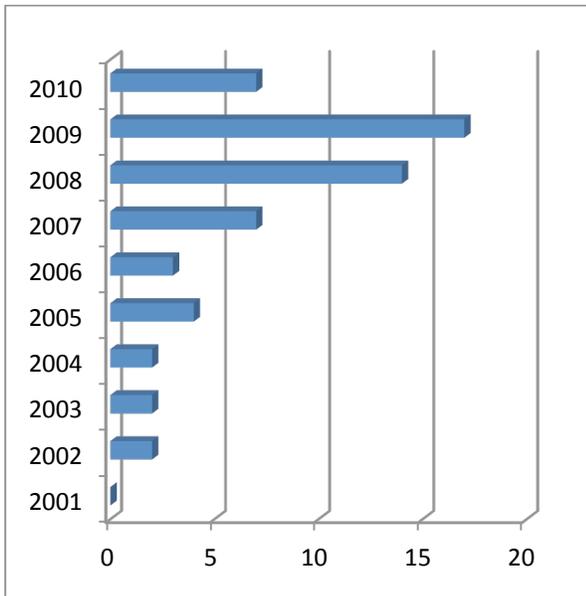


Figure 1 – Papers published by year regarding Agile Methods and UCD [7]

Objectives

The goal of this tutorial is to provide to the audience an overview of Agile Methods, practices used to integrate Agile Methods and Interaction Design as well as some practical experiences regarding this integration in the industry context. Most of the practices presented in this tutorial were collected in a Systematic Literature Review [7] and were implemented in an industry context.

Audience

This tutorial is suitable for undergraduate and graduate students, professors, researchers and professionals of Computer Science interested in Interaction Design and/or Agile Methods and how these two fields work together.

Due to the popularity and growing concern on Agile Methods and its adherence by the software industry, added to the introduction of new concepts of interaction between human beings and computers and the need of software systems really usable, we think this theme is totally relevant to the audience of IHC'11.

Material

Previous reading of some papers and/or book chapters will be recommended by the authors. This material will be available at the Tutorial website.

TOPICS

This tutorial will be offered in 4 hours with a 30 minutes break and it is organized as follows:

The first topic aims to provide an introduction and overview of Agile Methods as well to present common agile methods and some of the agile principles.

The second one aims to present some of the practices identified in a systematic literature review on the integration of agile methods and user-centered design.

The third topic aims to share with the audience some of the practical experiences faced by the authors both in academic and industry context.

These three topics will be presented as follows:

Topic 1 – Agile Methods (70 min)

- Introduction to Agile Methods
- Common Agile Methods
- Agile Principles

Topic 2 – Agile User-Centered Design (70 min)

- Best practices for integrating Human-Computer Interaction with Agile
- Structure of a User-Centered Agile Process

Topic 3 – Practical Experiences (70 min)

- Pros and cons of practical application of Agile User-Centered Design
- Practical activity
- Discussion

BIOGRAPHIES

Tiago Silva da Silva graduated in System Analysis at UCPel (Universidade Católica de Pelotas) in 2005 and obtained a Master degree in Computer Science at PUCRS (Pontifícia Universidade Católica do Rio Grande do Sul) in 2008. He is a PhD student at PUCRS and currently he is at University of Calgary as a Visiting Student, performing part of his PhD research with a scholarship from Capes (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior). His research interests include user-centered design, model-based usability evaluation, agile software development and the integration of agile methods and interaction design.

Milene Silveira obtained a PhD degree in Informatics at PUC-Rio (Pontifícia Universidade Católica do Rio de Janeiro) in 2002. Currently she is an Associate Professor at PUCRS (Pontifícia Universidade Católica do Rio Grande do Sul), where she has been a professor since 1994. Her research interests include Human-Computer Interaction, Informatics in Education, and her main research topic is the development of online help systems. Milene is a member of CEIHC (Comissão Especial em Interação Humano-Computador), acting as its current Coordinator.

Frank Maurer is a professor at the University of Calgary and leads the Agile Software Engineering group. His research interests include agile software methodologies, engineering digital table applications, executable acceptance test driven development, integrating agile methods with interaction design approaches, framework & API usability, tools for agile teams, and experience management. Dr. Maurer is a member of the Agile Alliance and a Certified Scrum Master. He is an Associate Editor of IEEE Software and recently served as Program Co-Chair

for XP 2009 as well as Agile 2007, two leading conferences on agile methods.

REFERENCES

- 1 Sommerville, Ian. *Software Engineering*. Addison-Wesley, 2007.
- 2 Hussain, Zahid, Slany, Wolfgang, and Holzinger, Andreas. Current State of Agile User-Centered Design: A Survey. (Linz 2009), Springer-Verlag.
- 3 Nodder, Chris and Nielsen, Jakob. *Agile Usability: Best Practices for User Experience on Agile Development Projects*. NN/g - Nielsen Norman Group, Fremont, 2009.
- 4 McInerney, Paul and Maurer, Frank. UCD in Agile Projects: Dream Team or Odd Couple? *Interactions* (2005), 19-23.
- 5 Fox, David, Sillito, Jonathan, and Maurer, Frank. Agile Methods and User-Centered Design: How These Two Methodologies are Being Successfully Integrated in Industry. In *Proceedings of the Agile 2008* (Washington 2008), IEEE Computer Society, 63-72.
- 6 Williams, Heather and Ferguson, Andrew. The UCD Perspective: Before and After Agile. In *Proceedings of the AGILE 2007* (Washington D.C. 2007), IEEE Computer Society, 285-290.
- 7 da Silva, Tiago Silva, Martin, Angela, Maurer, Frank, and Silveira, Milene. User-Centered Design and Agile Methods: A Systematic Review. (Salt Lake City 2011), IEEE Computer Society.